



ENTRUST

Entrust PKI Hub 1.0

Installation and Administration Guide

Document issue: 1.0

Issue date: December 3, 2024

© 2024, Entrust. All rights reserved

Entrust and the hexagon design are trademarks, registered trademarks and/or service marks of Entrust Corporation in Canada and the United States and in other countries. All Entrust product names and logos are trademarks, registered trademarks and/or service marks of Entrust Corporation. All other company and product names and logos are trademarks, registered trademarks and/or service marks of their respective owners in certain countries.

This information is subject to change as Entrust reserves the right to, without notice, make changes to its products as progress in engineering or manufacturing methods or circumstances may warrant. The material provided in this document is for information purposes only. It is not intended to be advice. You should not act or abstain from acting based upon such information without first consulting a professional. ENTRUST DOES NOT WARRANT THE QUALITY, ACCURACY OR COMPLETENESS OF THE INFORMATION CONTAINED IN THIS ARTICLE. SUCH INFORMATION IS PROVIDED "AS IS" WITHOUT ANY REPRESENTATIONS AND/OR WARRANTIES OF ANY KIND, WHETHER EXPRESS, IMPLIED, STATUTORY, BY USAGE OF TRADE, OR OTHERWISE, AND ENTRUST SPECIFICALLY DISCLAIMS ANY AND ALL REPRESENTATIONS, AND/OR WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT, OR FITNESS FOR A SPECIFIC PURPOSE.

Contents

1 About this guide	7
Revision information.....	7
Documentation feedback.....	7
2 Overview	8
Cluster controller	8
Management console.....	8
Logs and metrics console	8
Operating system.....	8
Other tools.....	8
3 Release notes	10
Installation known issues	10
CA Gateway known issues	10
Certificate Enrollment Gateway known issues.....	11
Certificate Hub known issues	12
Entrust Validation Authority known issues.....	16
Timestamping Authority known issues	16
4 Requirements.....	17
Machine requirements.....	17
Network requirements.....	19
Software requirements.....	22
HSM requirements	22
Solution-specific requirements.....	23
5 Starting up PKI Hub.....	25
Downloading the Entrust PKI Hub image	25
Verifying the downloaded files.....	26
Installing the Entrust PKI Hub image	26
Running clusterctl install.....	70
Replacing the default TLS certificate	71
Configuring the proxy	72

Changing the keyboard layout	73
Changing the operating system timezone	73
Configuring time synchronization	74
Manually starting starting the chrony service	74
Configuring an nShield HSM	74
6 Logging into the Management Console	77
7 Setting or updating the license	78
8 Starting up Entrust solutions	79
Starting up Certificate Authorities	79
Starting up CA Gateway	92
Starting up Certificate Enrollment Gateway	181
Starting up Certificate Hub	396
Starting up Timestamping Authority	423
Starting up Entrust Validation Authority	454
Starting up Entrust log-forwarder	511
9 Browsing logs with Grafana	514
Browsing and exporting logs with the Grafana Loki Dashboard	515
Browsing log file contents with Grafana	517
10 Administrating	519
Adding nodes	519
Administrating console users	519
Backing up and restoring the state	528
Checking the etcd database size	530
Checking the persistent volume disk usage	530
Defragmenting the etcd database	530
Managing the retention policies	531
Recovering from disaster	531
Restarting the nodes	532
Updating DNS resolution	532
11 Uninstalling	534

12 clusterctl reference	535
clusterctl backup create	535
clusterctl backup restore.....	536
clusterctl certificate	537
clusterctl help.....	538
clusterctl install.....	538
clusterctl license import	539
clusterctl node add	540
clusterctl node info	541
clusterctl node join-token	541
clusterctl proxy clear	541
clusterctl proxy info	542
clusterctl proxy set.....	542
clusterctl retention config logs	544
clusterctl retention config metrics.....	545
clusterctl retention info	546
clusterctl solution config export	546
clusterctl solution config import.....	547
clusterctl solution deploy.....	548
clusterctl solution info	549
clusterctl solution secret set	550
clusterctl solution upload.....	551
clusterctl uninstall	552
clusterctl upgrade.....	552
clusterctl version.....	553
clusterctl volume capacity	553
clusterctl volume info	554
13 CIS benchmarks	555
Linux CIS benchmarks	555
Password policy CIS benchmarks	557
Kubernetes CIS benchmarks	558

14 Troubleshooting and technical assistance	568
Entrust TrustedCare.....	568
Customer support	568
Professional services	570
Training.....	570
15 Third-party license acknowledgments	571
16 Licensing.....	572
17 Certificate profiles reference	574
Basic authority certificate profiles	574
External subordinate CA certificate profiles.....	576
Subscriber certificate profiles	584

1 About this guide

This guide describes installing and managing Entrust PKI Hub 1.0.

- [Revision information](#)
- [Documentation feedback](#)

Revision information

See the following table for the changes in each document issue.

Issue	Date	Section	Changes
1.0	Dec 2024	All	The first release of this document.

Documentation feedback

You can rate and provide feedback about product documentation by completing the online feedback form:

<https://go.entrust.com/documentation-feedback>

Any information you provide goes directly to the documentation team and is used to improve and correct the information in our guides.

2 Overview

Entrust PKI Hub is a versatile and robust virtual appliance that streamlines and simplifies deployment across various environments of the following Entrust solutions.

- Certificate Authorities
- CA Gateway
- Certificate Enrollment Gateway
- Certificate Hub
- Timestamping Authority
- Entrust Validation Authority

Entrust PKI Hub is tested with different virtualization platforms and cloud providers.

- VMware vSphere
- Microsoft Hyper-V
- Nutanix
- Amazon Web Services (AWS)
- Microsoft Azure

At a lower level, PKI Hub comprises the following components.


- [Cluster controller](#)
- [Management console](#)
- [Logs and metrics console](#)
- [Operating system](#)
- [Other tools](#)

Cluster controller

The `clusterctl` command-line tool for managing the cluster and the Entrust solutions.

Management console

The web portal for managing Entrust solutions.

 See the corresponding solution guide for examples of managing a specific solution.

Logs and metrics console

The web portal (powered by [Grafana](#)) to browse logs and metrics on Entrust PKI Hub and the deployed solutions.

Operating system

Entrust PKI Hub runs on a Linux operating system meeting the [CIS benchmarks](#).

Other tools

The platform wraps the following third-party tools.

Component	Description	Provider
Calico	The internal network fabric.	github.com/projectcalico/calico
CoreDNS	The internal DNS server.	coredns.io
Docker registry	The container images repository.	docker.com
etcd	The internal database for the cluster configuration.	etcd.io
Grafana	The web portal for browsing logs and metrics.	grafana.com
Istio	The network traffic manager.	istio.io
K3s cluster	The container orchestration system.	k3s.io
Loki	The log aggregation system.	grafana.com
Longhorn	The cluster file system.	longhorn.io
Metrics server	The metrics monitoring system.	kubernetes.io
Prometheus	The metrics aggregation system.	prometheus.io
Promtail	The agent that ships local logs to Loki.	grafana.com

3 Release notes

PKI Hub 1.0 is the first commercial release of PKI Hub. This release has the following known issues.

- [Installation known issues](#)
- [CA Gateway known issues](#)
- [Certificate Enrollment Gateway known issues](#)
- [Certificate Hub known issues](#)
- [Entrust Validation Authority known issues](#)
- [Timestamping Authority known issues](#)

Installation known issues

The PKI Hub installation has the following known issues.

- [EDM-14516](#)
- [EDM-14996](#)

EDM-14516

On ISO image installations with BIOS boot, the system only loads when the larger disk is first in the BIOS boot order.

Workaround: See [Configuring the BIOS boot on a PKI Hub ISO installation](#) for details.

EDM-14996

Upgrading Entrust Deployment Manager 2.0.x to PKI Hub 1.0 is not supported.

Workaround: Wait for the PKI Hub 1.1 release, which will support upgrading.

CA Gateway known issues

CA Gateway releases 3.0.3 has the following known issues.

- [ATEAM-16246](#)
- [ATEAM-16264](#)

ATEAM-16246

When configuring a CA Gateway client, the following mandatory parameters are mutually exclusive (that is, you must select one but not both).

- Tenant ID
- Integrator ID

However, the Management Console raises an error during validation if any of these values is unselected.


Workaround:

1. Delete the client settings.
2. Recreate the client settings using either the **Tenant ID** or **Integrator ID** parameter.

ATEAM-16264

For performance reasons, the PKIaaS CA Plugin will not honor the `subject.certificates` field in the following endpoint.

```
api/v1/certificate-authorities/{caId}/subjects/dn
```

 Future releases may restore this functionality.

Certificate Enrollment Gateway known issues

This section describes known issues and limitations for Entrust Certificate Enrollment Gateway. For other known issues with Certificate Enrollment Gateway, see the [Knowledge](#) section of Entrust TrustedCare. Reference numbers are for internal purposes only.

Configuration backup is only supported in single-node deployments (CSF-704)

Certificate Enrollment Gateway is deployed as a solution into Entrust Deployment Manager. Only single-node deployments of Entrust Deployment Manager support the `clusterctl backup config` command for exporting the cluster configuration. For information about the limitations and workaround for a multi-node backup and restore process, see the Entrust Deployment Manager documentation.

Unsupported ACMEv2 features (PKI-30901)

The Certificate Enrollment Gateway implementation of the ACME Server does not support the following [RFC 8555](#) features:

- EdDSA signature algorithm
- Rate limits
- `termsOfService` optional string
- Changes of Terms of Service
- External Account Binding
- Pre-authorization

Unsupported Intune-SCEP operations (PKI-28149, PKI-31351)

The Certificate Enrollment Gateway integration with the Intune-SCEP protocol does not support the following [draft-nourse-scep-23](#) operations:

- GetCRL
- GetNextCACert

CSRs sent from ACMEv2 clients cannot have an empty Subject DN if they will be sent to Entrust Certificate Services for processing (ECSPR-39482)

If an ACMEv2 client sends a CSR (certificate signing request) with an empty Subject DN, Certificate Enrollment Gateway will use the first Subject Alternative Name value in the CSR as the Subject DN. Certificate Enrollment Gateway will not alter the CSR, but will send the Subject DN value as a separate parameter to CA Gateway for

processing. Entrust Certificate Services requires that CSRs must have a Subject DN. Entrust Certificate Services will ignore the Subject DN parameter sent by Certificate Enrollment Gateway.

Workaround: You must generate the CSR externally from the ACMEv2 client using another tool, such as openssl. The ACMEv2 client can then use the externally-generated CSR.

CSRs sent from ACMEv2 clients cannot have an empty Subject DN if they will be sent to a Microsoft CA for processing (PKI-32853)

If an ACMEv2 client sends a CSR (certificate signing request) with an empty Subject DN, Certificate Enrollment Gateway will use the first Subject Alternative Name value in the CSR as the Subject DN. Certificate Enrollment Gateway will not alter the CSR, but will send the Subject DN value as a separate parameter to CA Gateway for processing. A Microsoft Certification Authority (CA) requires that CSRs must have a Subject DN. A Microsoft CA will ignore the Subject DN parameter sent by Certificate Enrollment Gateway.

Note: This issue does not occur when using Certificate Enrollment Gateway with CA Gateway 2.5.0 or later. When using CA Gateway 2.5.0 or later, ACMEv2 clients can send a CSR with an empty Subject DN intended for a Microsoft CA without issue.

Workaround: You must generate the CSR externally from the ACMEv2 client using another tool, such as OpenSSL. The ACMEv2 client can then use the externally-generated CSR.

Authentication error message is always logged when enrolling for a certificate with a Cisco LibEST client using basic authentication (CEG-3287)

When enrolling for a certificate with a Cisco LibEST client and the client is using basic authentication, Certificate Enrollment Gateway will always log an authentication error, even when the simpleenroll and serverkeygen operations are successful. For example:

```
[2024-09-16 13:16:49.711] [ERROR] [10] [EST] [] [a06bef31] [https-jsse-nio-1443-exec-1]
[com.entrust.ceg.commons.audit.AuditLogger=>process] [Request to EST
operation:simpleenroll failed.Reason:Access to EST operation:simpleenroll must be
authenticated ]
```

This error is expected with the LibEST client. Even when using basic authentication, the LibEST client does not provide the parameters for basic authentication on the first request. When the EST server does not obtain the basic authentication parameters on the first request, it issues header "WWW-Authenticate" to the LibEST client. When LibEST client receives the "WWW-Authenticate" header, the client will repeat the request and include the basic authentication parameters.

Certificate Hub known issues

Entrust Certificate Hub 4.0.3 has the following known issues..

- [ATEAM-1445](#)
- [ATEAM-15463](#)
- [ATEAM-15933](#)
- [ATEAM-16039](#)
- [ATEAM-16346](#)
- [ATEAM-16436](#)
- [ATEAM-16910](#)

- [ATEAM-16920](#)
- [ATEAM-16923](#)
- [ATEAM-16930](#)
- [ATEAM-16942](#)
- [ATEAM-16950](#)
- [ATEAM-16982](#)
- [ATEAM-16986](#)
- [ATEAM-16988](#)
- [ATEAM-16997](#)
- [ATEAM-17030](#)
- [ATEAM-17063](#)
- [ATEAM-17070](#)
- [ATEAM-17072](#)
- [ATEAM-17230](#)

ATEAM-1445

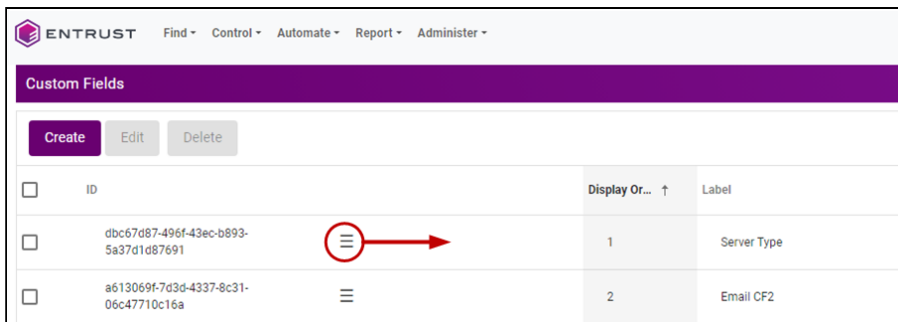
For certificates with **Revocation Reason: On Hold**, attempting to unhold the certificate may fail, or the **Unhold** option may be absent from the **Actions** dropdown.

ATEAM-15463

In the **Custom Fields** page of the web console, Certificate Hub administrators can change the display order of the custom fields. However:

- Reordering a custom field does not change the **Display Order** column value.
- Refreshing the **Custom Field** page reverts all changes.

Workaround: after dragging a custom field to a different position, move the ☰ drag icon within its row to make the changes persistent.



ATEAM-15933

The following endpoint filters do not display correct results on the downloaded report files.

- is empty
- is not empty

ATEAM-16039

Certificates without a name are not successfully synced from a **Source**.


ATEAM-16346

When generating a report containing a large number of certificates, the report generation may fail with an out-of-memory error.

Workaround: Increasing the heap size of the `acm-api` pod (for report preview) and scheduler pod (for scheduled reports) may fix this issue. A Java Heap Size parameter has been added to the deployment script to allow for manual adjustment of the allocated heap size.

ATEAM-16436

Certificate Hub does not record wildcard certificates successfully scanned by the Discovery Scanner.

 Wildcard certificates are certificates containing the wildcard asterisk in the issuer and subject.

ATEAM-16910

If a report does not have a **Schedule**, the **Manage Schedules** option is disabled from the **Actions** dropdown of the **Report Designer** page.

ATEAM-16920

Switching the language before logging in does not affect the language of the **Delete** and **Cancel** buttons in the **Confirm Delete** popup on the **Destinations** page.

ATEAM-16923

The **Owner** grid column is not populated when accessing the **Report Schedules** from the **Report Designer** grid.

Workaround: Access the **Report Schedules** grid from the navigation bar.

ATEAM-16930

The **Authorities** page does not display authorities missing an authority certificate.

ATEAM-16942

Switching language in the **Report Designer** page does not affect the labels of the grid columns.

ATEAM-16950

After the failed verification of a Destination, clicking **Verify** again does not trigger a new verification.

Workaround: Make any change in the create form – for example, change the **Description**.

ATEAM-16982

When issuing a certificate using the **Key Manager (KMIP)** destination, the **public key ID** is also referring to the **private key ID**

ATEAM-16986

When Certificate Hub is licensed with a FIND license, selecting the **Domains** widget on the **Dashboard** displays the following error.

```
Unable to show information: Forbidden. This request is not allowed.
```

ATEAM-16988

When Certificate Hub is licensed with a FIND license, the option to **Archive** certificates is missing in the **Actions** dropdown on the **Certificates** grid.

ATEAM-16997

When creating a new certificate, the downloaded chain only includes the certificate of the CA that issued the new certificate instead of including the entire chain.

ATEAM-17030

Verification fails for IIS destinations if the username includes a domain name – for example:

```
.\user
```

```
domain\user
```

ATEAM-17063

Certificate creation fails when adding an `otherName` field to the Subject Alternative Names.

ATEAM-17070

Administrator requests for certificate issuance with Apache, Nginx, or IIS destinations fail during the approval process if the administrator performing the approval has only the Approver role.

Workaround: Let a global administrator with the Authority role perform the approval.

ATEAM-17072

The user guide does not indicate that Python 3.9 or newer is a requirement for Apache and Nginx destinations.

ATEAM-17230

When installing Certificate Hub on Kubernetes with an external database, you can safely ignore the following error message:

Error from server (NotFound): configmaps "postgres-config" not found

Entrust Validation Authority known issues


Unable to render include or excerpt-include. Could not retrieve page.

Timestamping Authority known issues

Timestamping Authority 2.1.1 has the following known issue.

EDM-13275

When integrated with a Splunk server, Entrust Deployment Manager does not forward logs recording `tsactl` commands. However, these logs can be browsed using the Grafana portal.

 See the Entrust Deployment Manager guide for integrating a Splunk server or browsing logs in the Grafana portal.

4 Requirements

As explained in [Overview](#), you can run Entrust PKI Hub on virtualization platforms and cloud providers. See below for the requirements shared by all types of installation.

- [Machine requirements](#)
- [Network requirements](#)
- [Software requirements](#)
- [HSM requirements](#)
- [Solution-specific requirements](#)

Machine requirements

The machines running Entrust PKI Hub must meet the following requirements.

- [CPU requirements](#)
- [Disk requirements](#)
- [Memory requirements](#)
- [Recommended number of nodes](#)

CPU requirements

You need the following CPU cores to install Entrust PKI Hub.

Installation	Number of deployed solutions	Minimum number of cores
Single-node	1	4
Single-node	>1	8
Multi-node	Any	4 per node

Disk requirements

PKI Hub installations require the following disks.

- [Main disk](#)
- [Additional disk](#)

Main disk

You need a main disk with the following requirements.

Setting	Required value
Size	1 TiB or more
Storage type	SSD (Solid-state Drive)

Additional disk

The `etcd` daemon requires a dedicated disk with the following requirements.

Setting	Required value
Size	15 GiB or more
Storage type	SSD (Solid-state Drive)
fsync latency	As explained in the etcd documentation , the p99 percentile of the <code>wal_fsync_duration_seconds</code> duration should be less than 10ms to confirm the disk is reasonably fast for production workloads.
IOPS (input/output operations per second)	50 or more sequential write operations per second.

Memory requirements

You must meet the following RAM requirements to install Entrust PKI Hub.

Installation	Number of deployed solutions	Minimum RAM size
Single-node	1	8GB
Single-node	>1	16 GB
Multi-node	Any	8 GB per node

Recommended number of nodes

In case of disaster, an Entrust PKI Hub installation with N nodes is available if at least $(N/2)+1$ nodes are available. This minimum number of working nodes is referred to as "quorum".

When deploying Entrust PKI Hub in high availability, we recommend 3 or 5 nodes because:

- A 2-node installation fails when a single node fails.
- A 4-node installation fails when 2 nodes fail, just like a 3-node installation. Adding a node to an odd number of nodes does not increase the number of tolerated node failures.
- Entrust PKI Hub does not support more than 5 nodes.

See the table below for the supported node failures.

Cluster nodes	Nodes alive	Nodes broken	Quorum
3	1	2	✗
3	2	1	✓
5	1	4	✗
5	2	3	✗
5	3	2	✓
5	4	1	✓

Network requirements

Connect Entrust PKI Hub to a network with the following requirements.

- [DNS requirements](#)
- [IP address requirements](#)
- [Load balancing requirements](#)
- [Required open ports](#)
- [Reserved subnets](#)

DNS requirements

The selected DNS servers must be accessible without a proxy, as Entrust PKI Hub cannot access a DNS server through a proxy.

IP address requirements

Entrust PKI Hub only supports IPv4 and disables IPv6 by default. In multi-node deployments, all nodes:

- Must have a fixed hostname and IP address.
- Must be in the same subnet, cloud region, or virtual network.

After the installation, run the following command in each node to check the IP address and subnet mask.

```
nmcli
```

Verify that all nodes are in the same subnet.

- ✗ After running the `clusterctl install` or `clusterctl node add` commands you cannot change the IP address or hostname of a node.

Load balancing requirements

In multi-node installations, you must set up an external load balancer to operate as a single point of contact for Entrust solutions users and distribute the incoming traffic across all the cluster nodes. This prevents any node from becoming a single point of failure.

⚠ It is recommended to use sticky web connections because sessions with Entrust solutions can experience issues when the load balancer switches nodes.

Required open ports

The Entrust PKI Hub operation requires opening the following ports.

- [Required ports for incoming traffic](#)
- [Required ports for internode communication](#)

Required ports for incoming traffic

In all the installation nodes, open the following ports for incoming traffic to Entrust PKI Hub.

Target Port	Protocol	Source	Target service
22	TCP/SSH	The IP of the sysadmin Entrust PKI Hub administrator.	SSH
443	TCP/HTTPS	The IP of the Grafana and Management Console users.	Grafana and the Management Console.

Open also the following port for incoming traffic to the services of the deployed Entrust solutions.

Target Port	Protocol
80	TCP/HTTP

i Refer to the solution documentation for any product leveraging port 80.

Required ports for internode communication

In multi-node installations, the following ports allow traffic to internal services, such as:

- Monitoring node status
- Synchronizing data between nodes

You don't need to manually open these ports in the firewall of the host machines, as running the following commands will automatically opens them:

- The `clusterctl install` command executed when [Running clusterctl install](#).
- the `clusterctl backup restore` and `clusterctl node add` commands executed when [Administering](#)

However, ensure no network restriction blocks access to these ports.

Port	Protocol	Source	Destination
179	TCP	All nodes	All nodes
2379	TCP	All nodes	All nodes
2380	TCP	All nodes	All nodes
2381	TCP	All nodes	All nodes
4789	UDP	All nodes	All nodes
5473	TCP	All nodes	All nodes
6443	TCP	All nodes	All nodes
8000	TCP	All nodes	All nodes
9100	TCP	All nodes	All nodes
10250	TCP	localhost	localhost
15014	TCP	All nodes	All nodes
15021	TCP	All nodes	All nodes
30000	TCP	localhost	localhost
51820	UDP	All nodes	All nodes

Reserved subnets

Reserve the following subnets for Entrust PKI Hub use.

Subnet	Reserved use
10.42.0.0/16	Kubernetes Pods
10.43.0.0/16	Kubernetes ClusterIP Services

✘ Any LAN component utilizing an IP within these ranges will interfere with the operation of Entrust PKI Hub.

Software requirements

Entrust PKI Hub supports integration with the the following third-party software.

- [SIEM requirements](#)
- [Web browser requirements](#)

SIEM requirements

Entrust PKI Hub can forward logs to the following SIEM (Security Information and Event Management) systems.

- Splunk Enterprise
- Splunk Cloud

Entrust PKI Hub allows the ingestion of system records via the HTTP Event Collector (HEC) interface.

ℹ See [Starting up Entrust log-forwarder](#) for how to enable log-forwarding.

Web browser requirements

The Management Console of Entrust PKI Hub supports the following web browsers.

Browser	Supported versions
Apple Safari	5 or higher
Google Chrome	8 or higher
Microsoft Edge	All
Mozilla Firefox	9 or higher

HSM requirements

The following Entrust solutions support a Hardware Security Module (HSM) for cryptographic operations.

- Certificate Authorities (CAs)
- Timestamping Authority (TSA)
- Entrust Validation Authority (EVA)

See the following table for supported versions.

Provider	Hardware	Client drivers	Firmware	C A	TSA	V A
Entrust nShield	nShield Connect XC (Security World V3)	12.60.3 (FIPS 140-2 Level 3 mode supported)	12.60.15 or 12.60.2	✓	✓	✓
Entrust nShield	nShield 5c	13.6.3	13.2.4	✓	✓	✓
Thales	Safenet - LunaSA 7.2.02.0	Luna HSM 10.7.0 (FIPS 140-2 Level 3 mode supported)	7.7.1-20	✗	✓	✓
Thales	Thales DPoD	Luna HSM 10.7.0 (FIPS 140-2 Level 3 mode supported)	7.7.1-20	✗	✓	✓

When integrating a Hardware Security Module (HSM):

- You cannot use HSMs from different providers simultaneously, meaning that nShield and Thales HSMs cannot coexist within the same deployment.
- You can only use 1/N card sets. A card set of, for example, 2/5 cards is not supported.
- You do not need to install the client drivers because the solution already includes this software. However, these client drivers cannot be updated.

Solution-specific requirements

See the following sections for the requirements added by each deployed Entrust solution.

- [Solution-specific port requirements](#)
- [Solution-specific database requirements](#)
- [Solution-specific HSM requirements](#)

Solution-specific port requirements

See the following table for the additional open ports each Entrust solution requires.

Solution	Section
Certificate Authorities	Verifying port access for Certificate Authorities
CA Gateway	Verifying port access for CA Gateway
Certificate Enrollment Gateway	Verifying port access for Certificate Enrollment Gateway
Timestamping Authority	Verifying port access for Timestamping Authority

Solution	Section
Entrust Validation Authority	Verifying port access for Entrust Validation Authority

Solution-specific database requirements

Entrust Certificate Hub has the database requirements described in [Preparing the Certificate Hub database](#).

Solution-specific HSM requirements

See [HSM requirements](#) for the Hardware Security Module (HSM) each Entrust solution supports.

5 Starting up PKI Hub

To start up Entrust PKI Hub, perform the following operations in a node meeting the [Requirements](#).

- [Downloading the Entrust PKI Hub image](#)
- [Verifying the downloaded files](#)
- [Installing the Entrust PKI Hub image](#)
- [Running clusterctl install](#)
- [Replacing the default TLS certificate](#)
- [Configuring the proxy](#)
- [Changing the keyboard layout](#)
- [Changing the operating system timezone](#)
- [Configuring time synchronization](#)
- [Manually starting starting the chrony service](#)
- [Configuring an nShield HSM](#)

i For a multi-node or high-availability installation, perform the additional operations described in [Adding nodes](#).

Do not perform operating system modifications not covered in this guide. Specifically, do not:

- Change the system locale.
- Install antivirus, agents, or any other additional software.
- Update, modify, or remove operating system packages using mechanisms other than the one provided by Entrust.
- Modify the privileges of the `sysadmin` administrator.
- Create new users.
- Change file permissions.
- Change SELinux permissions or configuration.
- Move or delete files.
- Change partitions or mount points in a different way than documented.
- Change the kernel configuration.
- Change the boot loader configuration.
- Create or modify a `crontab`. Any automation task must be performed from an external machine accessing the system via SSH.

Downloading the Entrust PKI Hub image

See below for instructions on how to download the Entrust PKI Hub image.

To download the Entrust PKI Hub image

1. Log in to the secure trustedcare.entrust.com portal with your customer credentials.
2. Select the **PRODUCTS** tab.
3. Click **PKI Hub**.
4. Select the product version.
5. In the **SOFTWARE DOWNLOADS** tab, download the Entrust PKI Hub image that is suited to your platform.
 - Download the ISO image to install PKI Hub on VMware vSphere, Nutanix, or Microsoft Hyper-V.
 - Download the RAW image to install PKI Hub on Amazon Web Services (AWS).
 - Download the VHD image to install PKI Hub on Microsoft Azure
6. If required by the Entrust solutions you intend to deploy, download accessory tools such as SQL scripts to create the database.

Verifying the downloaded files


Generate a digest to verify the integrity of each downloaded installation and documentation file. On a Windows machine, you can run the following command line to generate the digest of the `<file>` file.

```
certutil -hashfile <file> SHA256
```

For example:

```
>certutil -hashfile c:\Users\john\Downloads\pki-hub-1.0.0.iso SHA256
SHA256 hash of c:\Users\john\Downloads\pki-hub-1.0.0.iso:
d841d57c7e1433622d219a7dea405935ff593a6831c1c94ba1c9dbde763b5baa
CertUtil: -hashfile command completed successfully.
```

On the **SOFTWARE DOWNLOADS** and **DOCUMENTATION** tabs, click the **Digest** column for each downloaded file and verify the displayed SHA-256 digest matches the generated one.

 Although TrustedCare also displays the MD5 and SHA-1 digests, we recommend using only the SHA-256 algorithm, which is more secure. Further versions of TrustedCare will remove the MD5 and SHA-1 algorithms from the digest list.

Installing the Entrust PKI Hub image

See below for installing and configuring the Entrust PKI Hub image on the supported platforms.

- [Installing the PKI Hub ISO image on an HCI](#)
- [Installing the Entrust PKI Hub RAW image on AWS](#)
- [Installing the Entrust PKI Hub VHD image on Azure](#)

 Contact Entrust support for instructions on installing Entrust PKI Hub on platforms not listed in this guide.

When completing the image installation and configuration, do not perform operating system modifications not covered in this guide. Specifically, do not:

- Install antivirus, agents, or any other additional software.
- Update, modify, or remove operating system packages using mechanisms other than the one provided by Entrust.
- Change the system locale.
- Create new users.
- Change `sysadmin` privileges
- Change file permissions.
- Change SELinux permissions or configuration.
- Move or delete files.
- Change partitions or mount points in a different way than documented.

Installing the PKI Hub ISO image on an HCI

See below for installing and configuring the PKI Hub ISO image on different HCI (hyper-converged infrastructure) providers.

- [Installing the Entrust PKI Hub ISO image on VMware vSphere](#)
- [Installing the Entrust PKI Hub ISO image on Microsoft Hyper-V](#)
- [Installing the Entrust PKI Hub ISO image on Nutanix](#)
- [Configuring a PKI Hub ISO image installation](#)

Installing the Entrust PKI Hub ISO image on VMware vSphere

See below for installing and configuring the Entrust PKI Hub image on a virtual machine hosted in VMware vSphere.

To install and configure the Entrust PKI Hub image on VMware vSphere

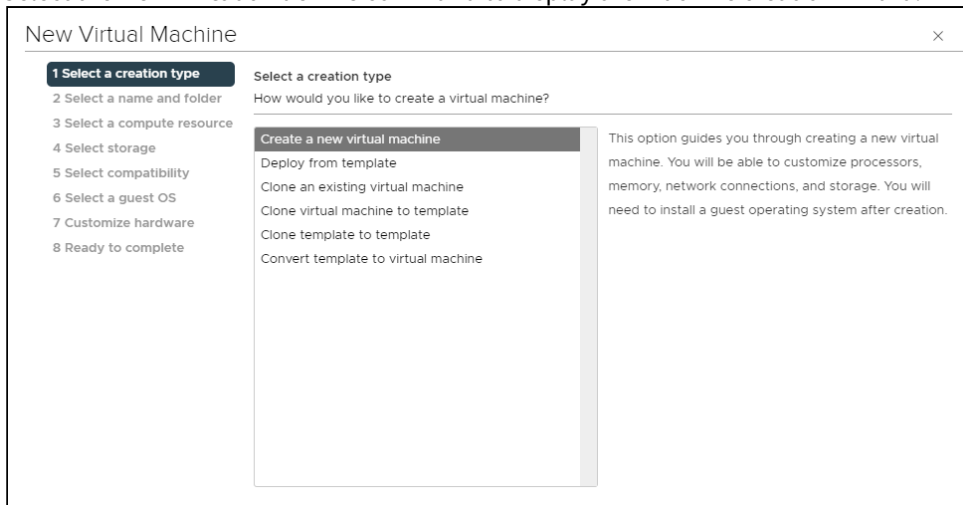
1. Download the ISO image, as explained in [Downloading the Entrust PKI Hub image](#).
2. Follow the below steps.
 - [Creating an Entrust PKI Hub virtual machine on VMware vSphere](#)
 - [Logging into Entrust PKI Hub on a VMware vSphere machine](#)
3. Configure the image installation as explained in [Configuring a PKI Hub ISO image installation](#).

Creating an Entrust PKI Hub virtual machine on VMware vSphere

Follow the steps below to install the Entrust PKI Hub ISO image as a VMware vSphere virtual machine.

To create an Entrust PKI Hub virtual machine on VMware vSphere

1. Log into your VMware vSphere portal.
2. Right-click on a node of the virtual machine navigation tree.
3. Select the **New Virtual Machine** command to display the machine creation wizard.



4. In each step of this wizard, select the values described below.
 - [Select a creation type](#)
 - [Select a name and folder](#)
 - [Select a compute resource](#)
 - [Select storage](#)
 - [Select compatibility](#)
 - [Select a guest OS](#)
 - [Customize hardware / Virtual Hardware](#)
 - [Customize hardware / VM Options](#)

Select a creation type

Configure the following settings.

Field	Value
Select a creation type	Select Create a new virtual machine.

Select a name and folder

Configure the following settings.

Field	Value
Virtual machine name	Write a name for the new virtual machine.
Location for the virtual machine	Select a folder for the new virtual machine.

Select a compute resource

Configure the following settings.

Field	Value
Select a compute resource	Select a computing resource for the new virtual machine.

Select storage

Configure the following settings.

Field	Value
Encrypt this virtual machine	Do not enable this option
Storage list	Select a storage resource for the new virtual machine.

Select compatibility

Configure the following settings.

Field	Value
Compatible with	Select ESXi 6.7 and later.

Select a guest OS

Configure the following settings.

Field	Value
Guest OS Family	Select Linux .
Guest OS Version	Select Red Hat Enterprise Linux 8 (64-bit) .

Customize hardware / Virtual Hardware

Configure the following settings in the **Virtual Hardware** tab of the **Customize hardware** page.

Field	Value
CPU	Select the number of cores recommended in CPU requirements .
Memory	Select the RAM size recommended in Memory requirements .
New Hard Disk	Select at least 1 TiB for the root disk, as explained in Disk requirements .
New Hard Disk	Click ADD NEW DEVICE > Hard disk and add a second disk for the <code>etcd</code> daemon. As explained in Disk requirements , this disk requires at least 15 GiB.
New CD/DVD Drive	Select the location of the ISO file.
New CD/DVD Drive > Status	Mark the Connect At Power On checkbox.

Customize hardware / VM Options

Configure the following settings in the **VM Options** tab of the **Customize hardware** page.

Field	Value	Description
Boot Options > Firmware	BIOS	Boot the machine with BIOS firmware.
	EFI	Boot the machine with UEFI firmware.

Logging into Entrust PKI Hub on a VMware vSphere machine

After installing the image on the machine, open a local or SSH session as the Entrust PKI Hub administrator.

Name	Initial password
sysadmin	changeme

When prompted, change the initial password with a password meeting the requirements described in [Password policy CIS benchmarks](#).

Installing the Entrust PKI Hub ISO image on Microsoft Hyper-V

See below for installing and configuring an Entrust PKI Hub image on a virtual machine hosted by a Microsoft Hyper-V hypervisor.

To install the Entrust PKI Hub image in Hyper-V

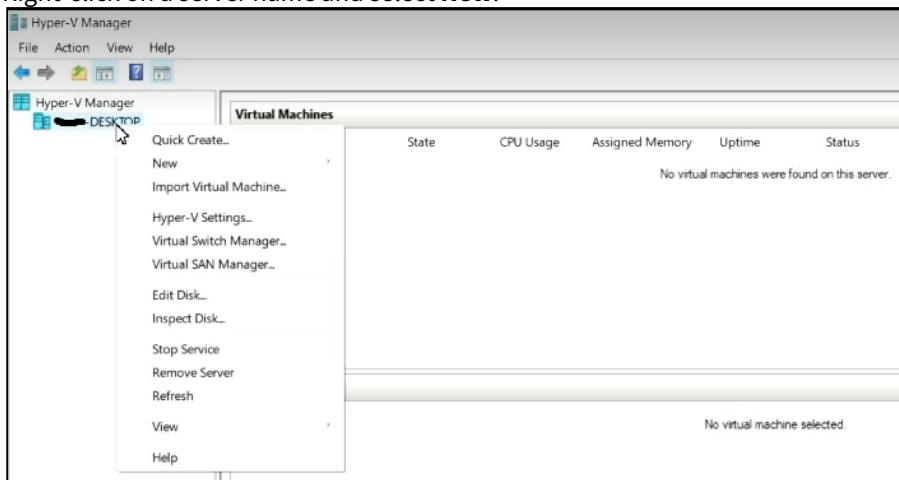
1. Download the ISO image, as explained in [Downloading the Entrust PKI Hub image](#).
2. Follow the below steps.
 - [Creating an Entrust PKI Hub virtual machine on Hyper-V](#)
 - [Configuring an Entrust PKI Hub virtual machine on Hyper-V](#)
 - [Starting an Entrust PKI Hub machine on Hyper-V](#)
3. Configure the image installation as explained in [Configuring a PKI Hub ISO image installation](#).

Creating an Entrust PKI Hub virtual machine on Hyper-V

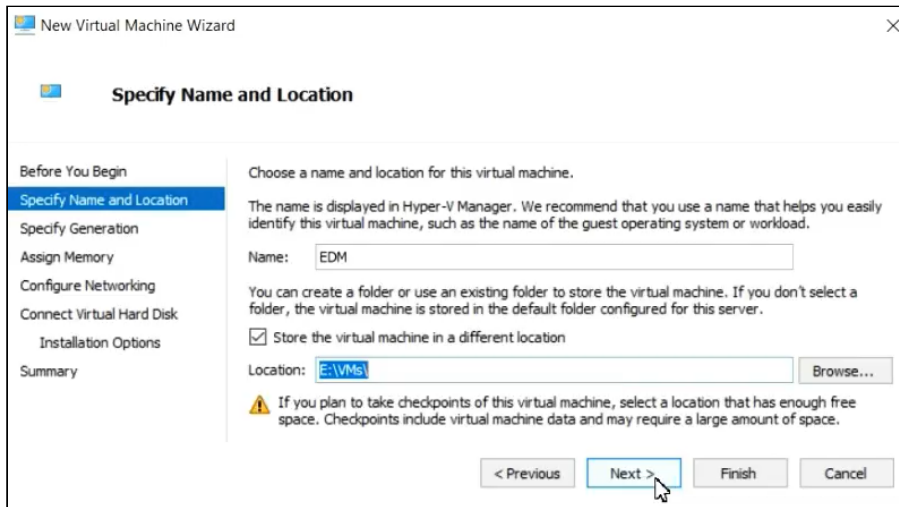
Install the Entrust PKI Hub ISO image as a new Hyper-V virtual machine.

To create an Entrust PKI Hub virtual machine on Hyper-V

1. Open the **Hyper-V Manager** console.
2. Right-click on a server name and select **New**.



3. In each step of the **New Virtual Machine Wizard** wizard, select the values described below.
 - [Specify Name and Location](#)
 - [Specify Generation](#)
 - [Assign Memory](#)
 - [Configure Network](#)
 - [Connect Virtual Hard Disk](#)
 - [Installation Options](#)
 - [Summary](#)



Specify Name and Location

Configure the following settings.

Field	Value
Name	Enter a name for the new virtual machine. For example, EDM .
Store the virtual machine in a different location	Enable this checkbox.
Location	Select a folder for the virtual machine file.

Specify Generation

Configure the following settings.

Field	Value
Choose the generation of this virtual machine	Select Generation 1 .

Assign Memory

Configure the following settings.

Field	Value
Startup memory	Select the RAM size stated in Memory requirements .

Field	Value
Use Dynamic Memory for this virtual machine	Enable this checkbox.

Configure Network

Configure the following settings.

Field	Value
Connection	Select a connection with a network meeting the Network requirements .

Connect Virtual Hard Disk

Under **Create virtual hard disk**, configure the root and `etcd` disks described in [Disk requirements](#).

Field	Root disk	etcd disk
Name	The name of the virtual machine file	The name of the virtual machine file
Location	The folder for storing the virtual machine file	The folder for storing the virtual machine file
Size	At least 1 TiB	At least 15 GiB

Installation Options

Configure the following settings.

Field	Value
Install an operating system from a bootable CD/DVD-ROM	Enable this radio button.
Image file (.sio)	Select the location of the ISO file described in Downloading the Entrust PKI Hub image .

Summary

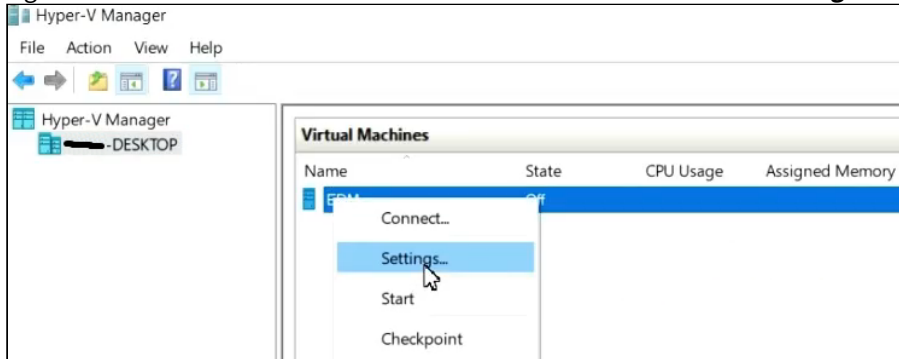
Review the selected settings and click **Finish** to complete the virtual machine creation.

Configuring an Entrust PKI Hub virtual machine on Hyper-V

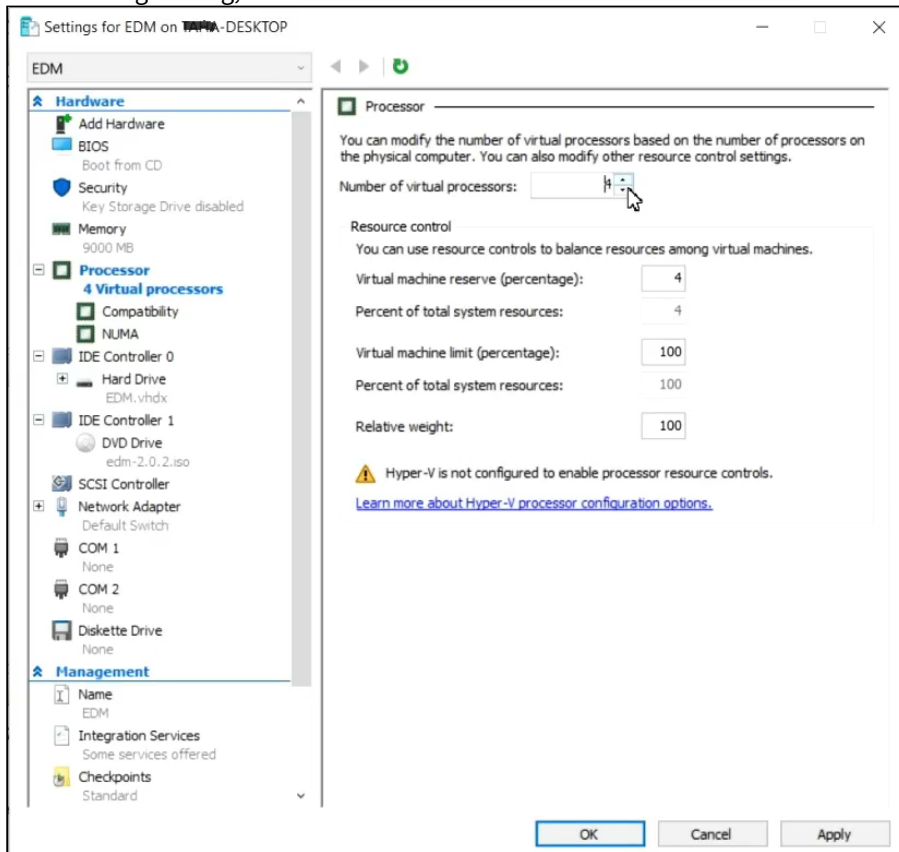
Once created, complete the new Entrust PKI Hub virtual machine as follows.

To configure an Entrust PKI Hub virtual machine on Hyper-V

1. Open the **Hyper-V Manager** console.
2. Right-click the name of the new machine virtual machine and select **Settings**.



3. In the settings dialog, select **Processors** under **Hardware**.



4. In the **Number of virtual processors** field, select the cores stated in [CPU requirements](#).
5. Click **Apply**.

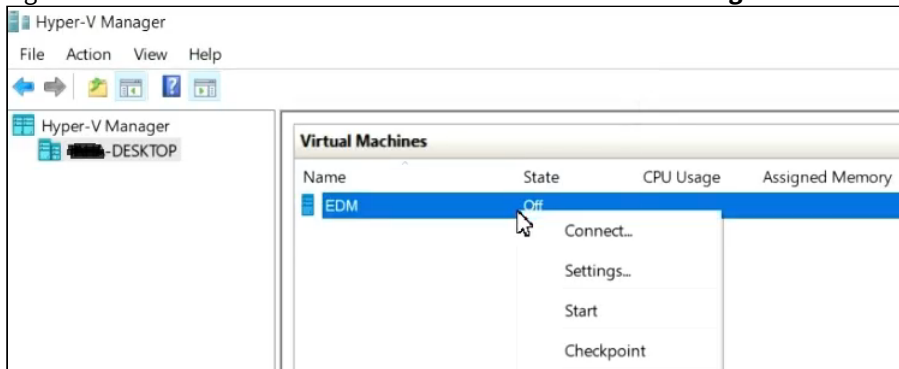
Starting an Entrust PKI Hub machine on Hyper-V

Start the new virtual machine and open a session as Entrust PKI Hub administrator.

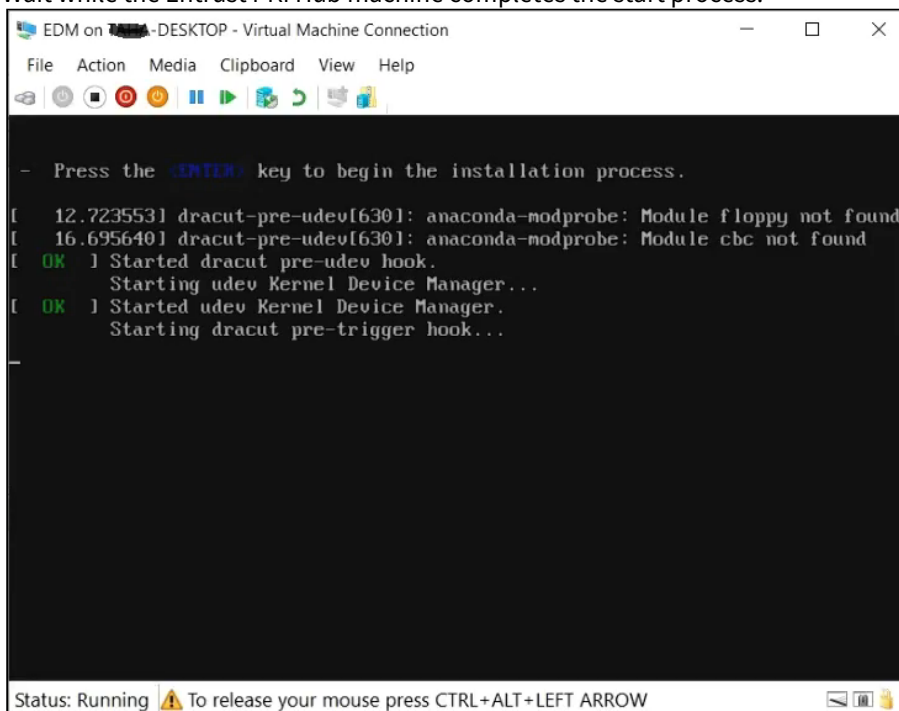
To start an Entrust PKI Hub virtual machine on Hyper-V

1. Open the **Hyper-V Manager** console.

- Right-click the name of the new virtual machine and select **Settings**.



- Wait while the Entrust PKI Hub machine completes the start process.



- Login with the `sysadmin` username and the `changeme` password.
- When prompted, change the initial password with a password meeting the requirements described in [Password policy CIS benchmarks](#).

Installing the Entrust PKI Hub ISO image on Nutanix

See below for installing and configuring an Entrust PKI Hub image on a virtual machine hosted by a Nutanix hypervisor.

 The installation and deployment steps in this guide have been tested with Nutanix version 6.5.5.7 LTS.

To install and configure the Entrust PKI Hub image on Nutanix

- Download the ISO image as explained in [Downloading the Entrust PKI Hub image](#).
- Follow the below steps.
 - [Uploading the Entrust PKI Hub image to Nutanix](#)

- [Creating an Entrust PKI Hub virtual machine on Nutanix](#)
3. Configure the image installation as explained in [Configuring a PKI Hub ISO image installation](#).

Uploading the Entrust PKI Hub image to Nutanix

Upload the Entrust PKI Hub ISO image to Nutanix, as explained in one of the following sections.

- [Uploading the Entrust PKI Hub ISO image with Nutanix Prism Element](#)
- [Uploading the Entrust PKI Hub image file with Nutanix Prism Central](#)
- [Importing the Entrust PKI Hub image to Nutanix Prism Central](#)

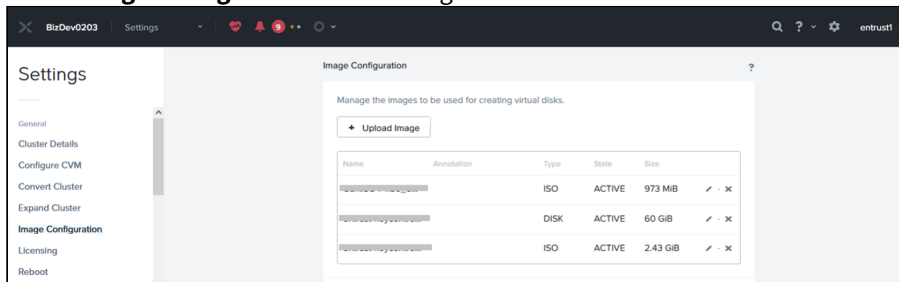
i As explained in [Creating a cluster of Entrust PKI Hub virtual machines with Nutanix Prism Center](#), uploading the image to Nutanix Prism Central allows you to create a cluster of Entrust PKI Hub machines.

Uploading the Entrust PKI Hub ISO image with Nutanix Prism Element

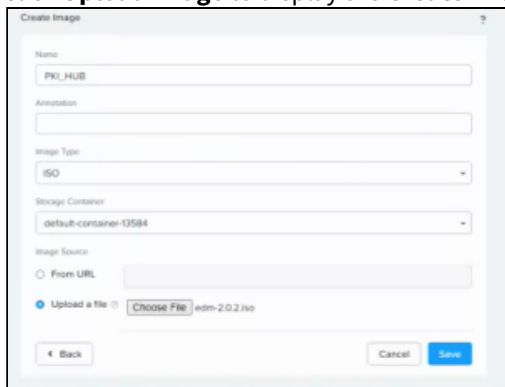
Follow the steps below to upload the Entrust PKI Hub ISO image using the Nutanix Prism Element portal.

To upload the Entrust PKI Hub ISO image with Nutanix Prism Element

1. Log into the Prism Element web portal of your Nutanix infrastructure.
2. Select the **Settings** gear icon on the top toolbar.
3. Select **Image Configuration** in the navigation sidebar.



4. Click **Upload Image** to display the **Create Image** dialog.



5. Enter a unique image name in the **Name** field. For example: **PKI_HUB**.
6. Write an optional description in the **Annotation** field.
7. Select **ISO** in the **Image Type** field.
8. Select the required container in the **Storage Container** field.
9. Click **Upload file** and select the file with ISO extension obtained in [Downloading the Entrust PKI Hub image](#).
10. Click **Save** and wait while the file uploads.

✘ Do not refresh the page while the file uploads.

11. Check the image **State** is **ACTIVE** on the image list.

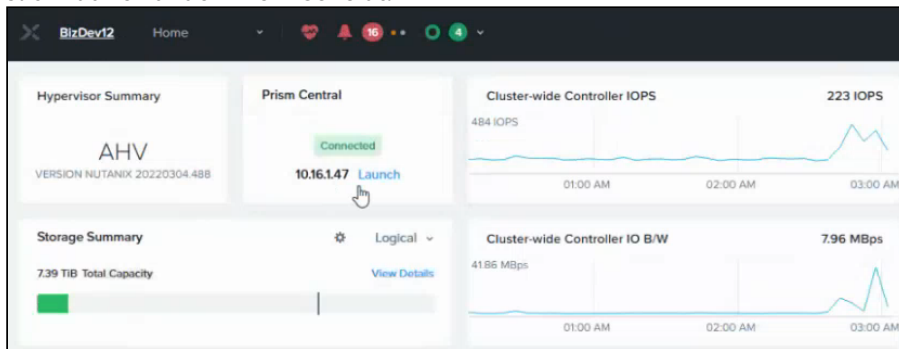
Image Configuration					
...	Created By: ...	ISO	ACTIVE	340 KIB	✓ - ✕
...	Created By: ...	DISK	ACTIVE	16 GiB	✓ - ✕
...	Created By: ...	DISK	ACTIVE	16 GiB	✓ - ✕
...		DISK	ACTIVE	60 GiB	✓ - ✕
...		DISK	ACTIVE	81 GiB	✓ - ✕
...		DISK	ACTIVE	81 GiB	✓ - ✕
...		DISK	INACTIV E	-	✓ - ✕
PKI_HUB		ISO	ACTIVE	5.36 GiB	✓ - ✕
...			ACTIVE	100 GiB	✓ - ✕
...			ACTIVE	100 GiB	✓ - ✕

Uploading the Entrust PKI Hub image file with Nutanix Prism Central

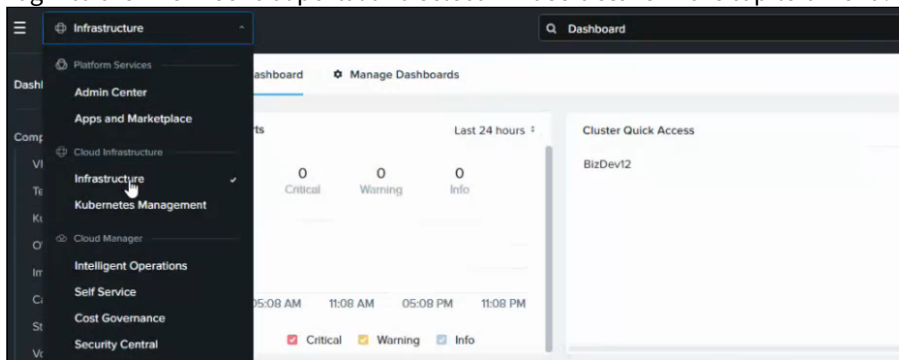
Follow the steps below for uploading the Entrust PKI Hub image using the Nutanix Prism Center portal.

To upload the Entrust PKI Hub image file with Nutanix Prism Central

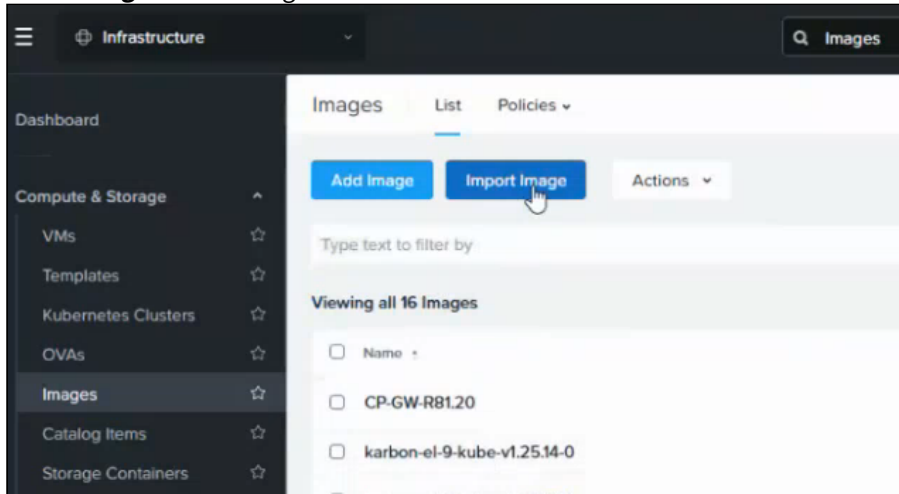
1. Log into your Nutanix Prism element web portal.
2. Click **Launch** under **Prism Central**.



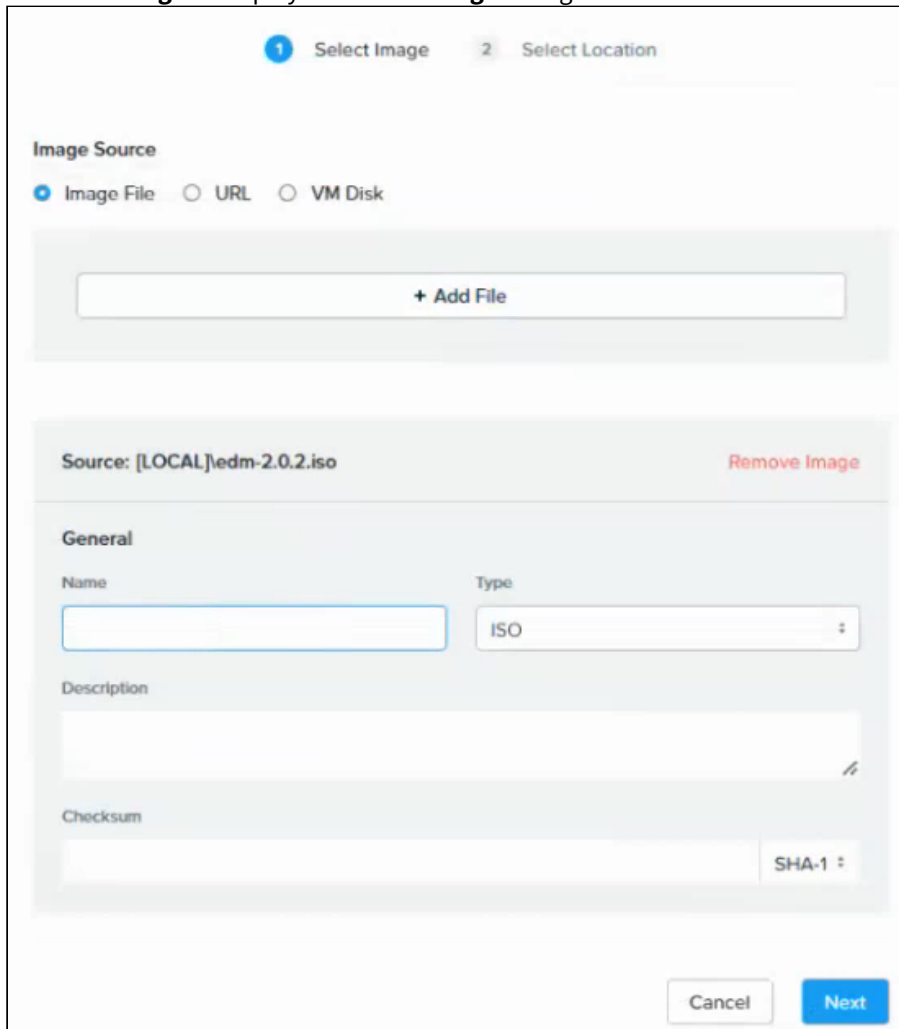
3. Log into the Prism Central portal and select **Infrastructure** in the top-left menu.



4. Select **Images** in the navigation sidebar.

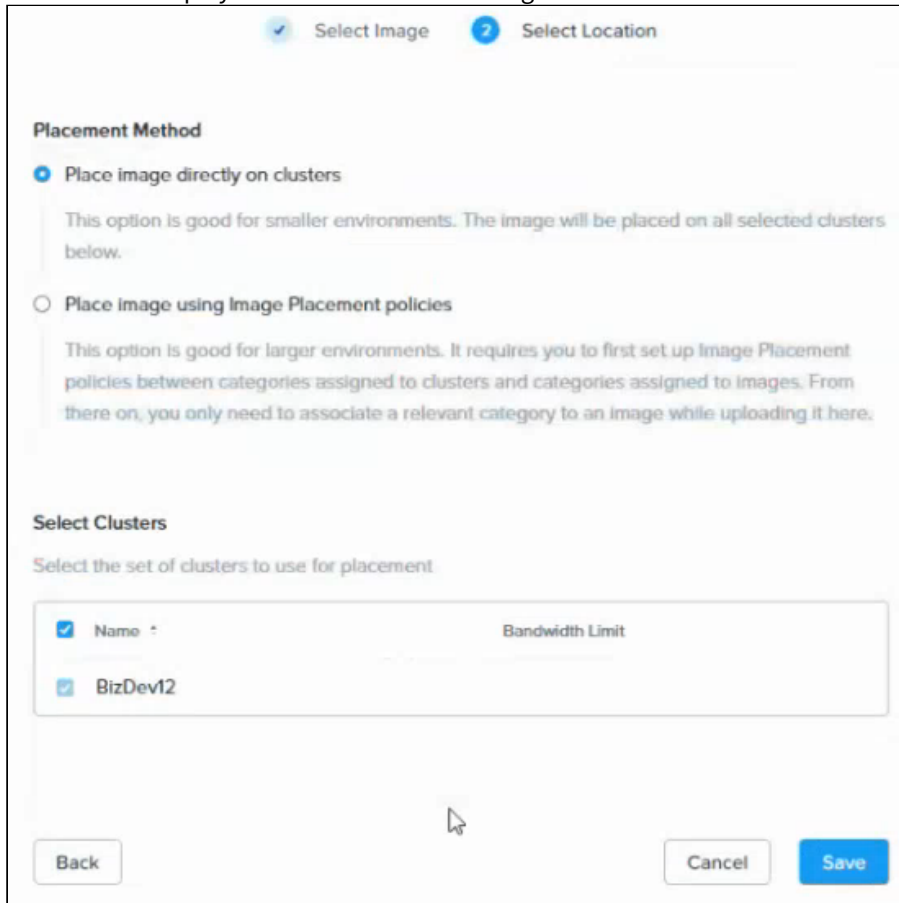


5. Click **Add Image** to display the **Select image** dialog.



6. Select **Image File** under **Image source**.
7. Click **+ Add File**

8. Select the file obtained in [Downloading the Entrust PKI Hub image](#).
9. Enter a unique image name in the **Name** field. For example: **PKI_HUB**.
10. Select **ISO** in the **Type** drop-down list.
11. Write an optional description in the **Description** field.
12. Click **Next** to display the **Select Location** dialog.



Placement Method

Place image directly on clusters

This option is good for smaller environments. The image will be placed on all selected clusters below.

Place image using Image Placement policies

This option is good for larger environments. It requires you to first set up Image Placement policies between categories assigned to clusters and categories assigned to images. From there on, you only need to associate a relevant category to an image while uploading it here.

Select Clusters

Select the set of clusters to use for placement

<input checked="" type="checkbox"/> Name	Bandwidth Limit
<input checked="" type="checkbox"/> BizDev12	

Back Cancel Save

13. Under **Select Clusters**, select the Nutanix cluster that will host the uploaded image.
14. Click **Save** and wait while the upload task completes.

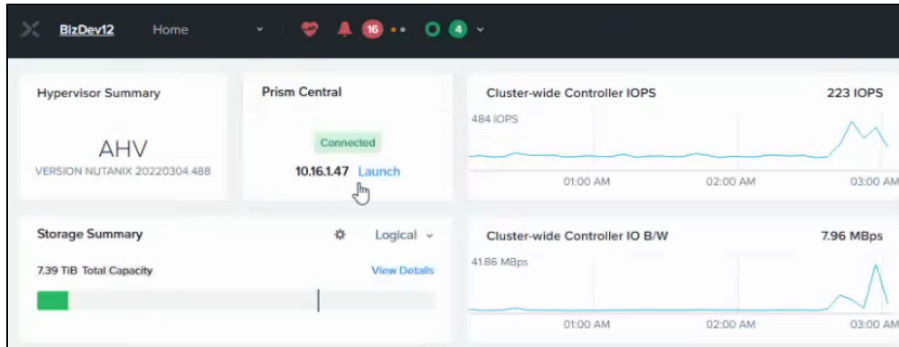
Importing the Entrust PKI Hub image to Nutanix Prism Central

If already uploaded as explained in [Uploading the Entrust PKI Hub ISO image with Nutanix Prism Element](#), you can import the PKI Hub image on Nutanix Prism Central.

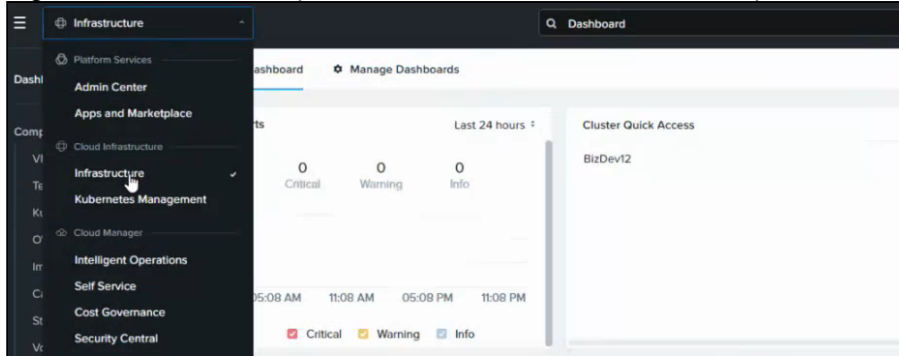
To import the Entrust PKI Hub image from Nutanix Prism Element to Nutanix Prism Central

1. Log into your Nutanix Prism element web portal.

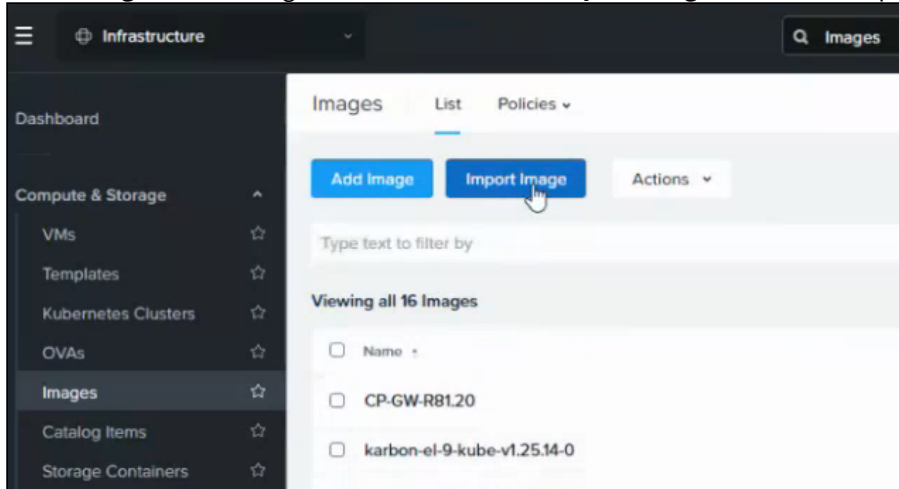
- Click **Launch** under **Prism Central**.



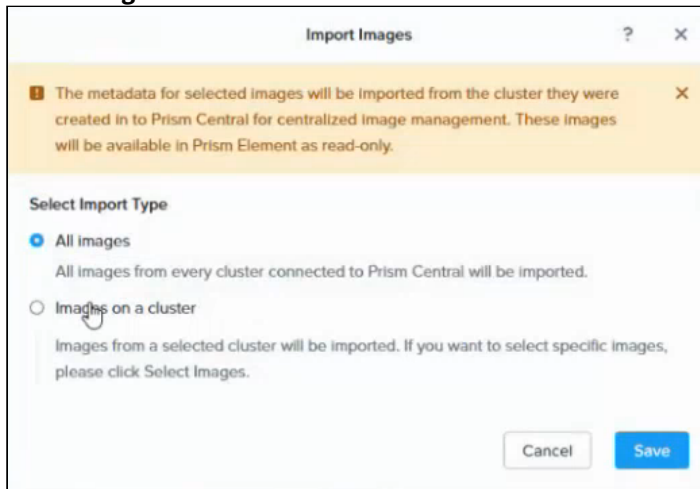
- Log into the Prism Central portal and select **Infrastructure** in the top-left menu.



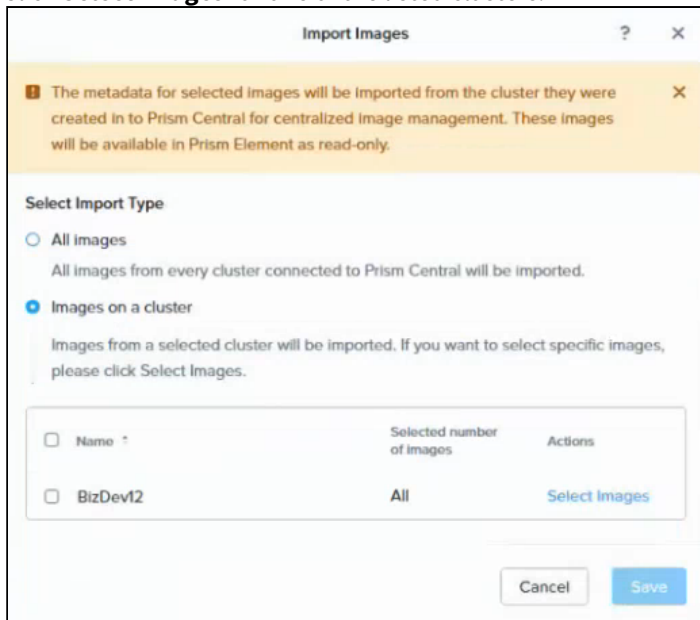
- Select **Images** in the navigation sidebar and click **Import Image** in the content pane.



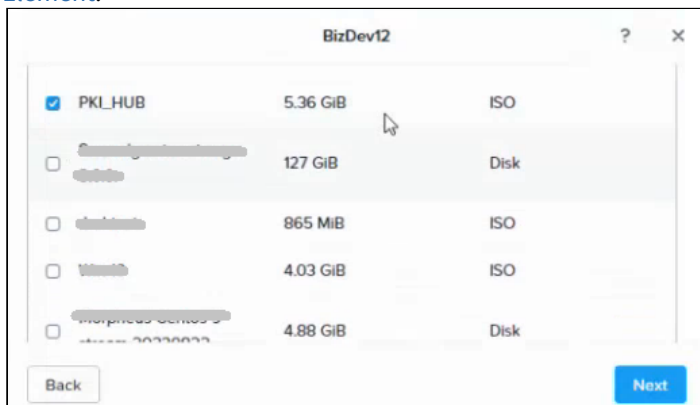
5. Select **Images on a cluster**.



6. Click **Select Images** for one of the listed clusters.



7. Select the image previously imported in [Uploading the Entrust PKI Hub ISO image with Nutanix Prism Element](#).



8. Click **Next** and wait while the import task completes.

Creating an Entrust PKI Hub virtual machine on Nutanix

You have the following options to create an Entrust PKI Hub virtual image on Nutanix.

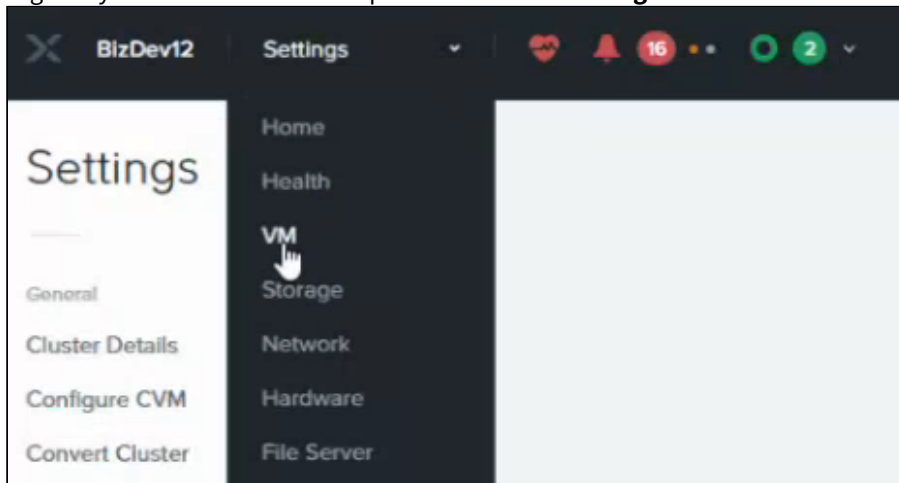
- [Creating a PKI Hub virtual machine with Nutanix Prism Element](#)
- [Creating a cluster of Entrust PKI Hub virtual machines with Nutanix Prism Center](#)

Creating a PKI Hub virtual machine with Nutanix Prism Element

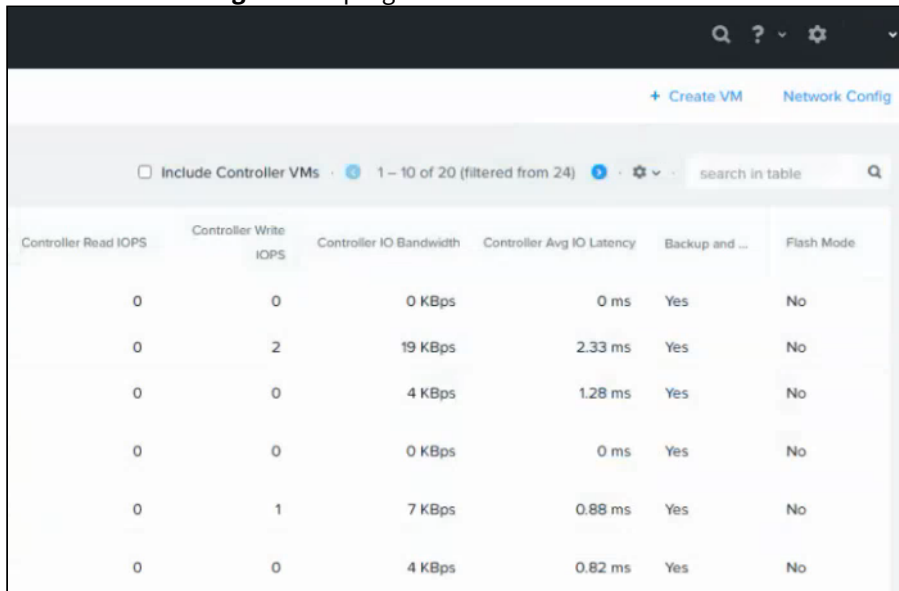
Follow the steps below to deploy the Entrust PKI Hub image as a virtual image with the Nutanix Prism Element portal.

To create a PKI Hub virtual machine with Nutanix Prism Element

1. Log into your Nutanix Prism web portal and select **Settings > VM**.



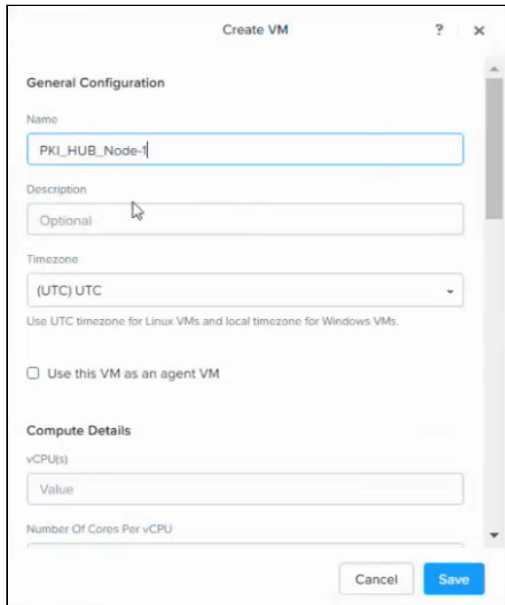
2. Click **Network Config** in the top-right corner.



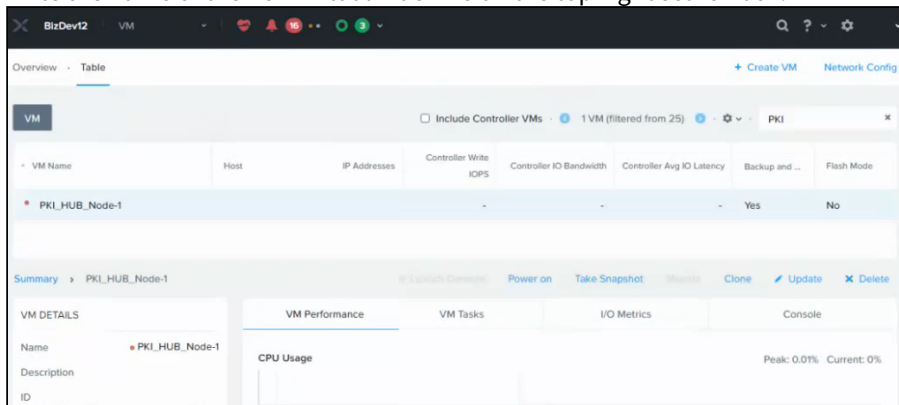
Controller Read IOPS	Controller Write IOPS	Controller IO Bandwidth	Controller Avg IO Latency	Backup and ...	Flash Mode
0	0	0 KBps	0 ms	Yes	No
0	2	19 KBps	2.33 ms	Yes	No
0	0	4 KBps	1.28 ms	Yes	No
0	0	0 KBps	0 ms	Yes	No
0	1	7 KBps	0.88 ms	Yes	No
0	0	4 KBps	0.82 ms	Yes	No

3. Create a new network or configure an existing one to meet the [Network requirements](#).
4. Click **+ Create VM** in the top-right corner.
5. Configure the following settings in the **Create VM** dialog.
 - [General Configuration](#)
 - [Compute Details](#)

- [Boot Configuration](#)
- [Disks](#)
- [Network Adapters \(NIC\)](#)



6. Click **Save** and wait while the virtual machine is created.
7. Write the name of the new virtual machine on the top-right search box.



VM Name	Host	IP Addresses	Controller Write IOPS	Controller IO Bandwidth	Controller Avg IO Latency	Backup and ...	Flash Mode
PKI_HUB_Node-1						Yes	No

8. Select the new virtual machine on the content pane and click **Power On** in the bottom menu.
9. Click **Launch Console** in the bottom menu and wait while PKI Hub starts.
10. Log in with the `sysadmin` username and the `changeme` password.
11. When prompted, change the initial password with a password meeting the requirements described in [Password policy CIS benchmarks](#).

General Configuration

Configure the following settings.

Setting	Value
Name	Enter a unique name for the new virtual machine. For example, PKI_HUB_NODE_1 .
Description	Enter an optional description for the new virtual machine.
Timezone	Select a timezone for the new virtual machine.
Use the VM as an agent VM	Do not enable this checkbox.

Compute Details

Configure the following settings.

Setting	Value
vCPU(s)	Enter the number of cores recommend in CPU requirements .
Number Of Cores Per vCPU	Select 1.
Memory	Enter the RAM size recommended in Memory requirements .

Boot Configuration

Select either **Legacy Boot** or **UEFI**.

Disks

Click the pencil edit button for **CD-ROM** and configure the following settings in the **Update Disk** dialog,

Setting	Value
Operation	Select Clone from Image Service .
Bus Type	Select SATA .
Image	Select the name previously assigned to the PKI Hub image when Uploading the Entrust PKI Hub ISO image with Nutanix Prism Element .

Click **+ Add New Disk** to successively configure the root and `etcd` disks described in [Disk requirements](#).

Setting	Root disk	etcd disk
Type	Disk	Disk
Operation	Allocate on Storage Container	Allocate on Storage Container
Bus Type	SCSI	SCSI
Storage Container	The container for the new virtual machine	The container for the new virtual machine
Size	At least 1 TiB	At least 15 GiB
Index	Next Available	Next Available

Network Adapters (NIC)


Click **+ Add New NIC** and configure the following settings in **Create NIC**.

Setting	Value
Subnet Name	Select a subnet meeting the Network requirements .
Network Connection Status	Select Connected .

Click **+ Set Affinity** to select the host that will run the virtual machine.

Creating a cluster of Entrust PKI Hub virtual machines with Nutanix Prism Center

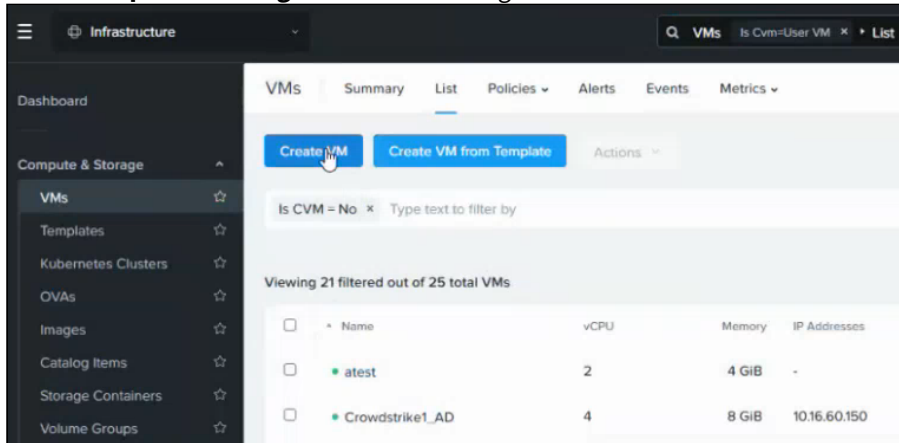
Follow the steps below to deploy a cluster of multiple Entrust PKI Hub virtual machines.

 See [Adding nodes](#) for how to join the different Entrust PKI Hub cluster machines.

To a cluster of Entrust PKI Hub virtual machines with Nutanix Prism Center

1. Log into the Nutanix Prism Center web portal.

2. Select **Compute & Storage > VMs** in the navigation sidebar and click **Create VM** in the content pane.



3. Fill in the forms on each page of the **Create VM** wizard.
 - [Configuration](#)
 - [Resources](#)
 - [Management](#)
 - [Review](#)
4. Click **Create VM**.

Configuration

Configure the following settings in the **Configuration** page of the **Create VM** wizard.

Setting	Value
Name	Enter a unique name for the new virtual machines.
Description	Enter an optional description for the new virtual machines.
Cluster	Select the Nutanix cluster that will host the virtual machines.
Number of VMs	Select the number of virtual machines you want to create.
CPU	Enter the number of cores recommend in CPU requirements .
Cores Per CPU	Select 1
Memory	Enter the RAM size recommended in Memory requirements .

Resources

In the **Resources** page of the **Create VM** wizard, click **Attach Disk** and configure the following settings.

Setting	Value
Type	Select CD-ROM .
Operation	Select Clone from Image .
Image	Select the image imported or uploaded in Uploading the Entrust PKI Hub image to Nutanix .
Bus Type	Select SATA .

Click again **Attach Disk** and configure the following settings in the **Attach Disk** dialog.

Setting	Value
Type	Select Disk .
Operation	Select Allocate on Storage Container .
Storage Container	Select a storage container for the disk of the PKI Hub virtual machine.
Capacity	Select at least 1 TiB (1024 GiB) as explained in Disk requirements .
Bus Type	Select SCSI .

Under **Network**, click **Attach to Subnet** and select a subnet meeting the [Requirements](#).

Under **Boot Configuration**, select either **Legacy Boot** or **UEFI**.

Management

In the **Management** page of the **Create VM** wizard, click the **Timezone** drop-down list and select a timezone for the new virtual machine.

Review

In the **Review** page of the **Create VM** wizard, check the settings of the new virtual machine and edit them if required.

Configuring a PKI Hub ISO image installation

Perform the following configuration steps after installing the PKI Hub ISO image in the selected platform.

- [Configuring the connection of a PKI Hub ISO installation](#)
- [Checking the connection of a PKI Hub ISO installation](#)
- [Configuring the boot mode of a PKI Hub ISO installation](#)

Configuring the connection of a PKI Hub ISO installation

See below for configuring the network connection of an Entrust PKI Hub machine.

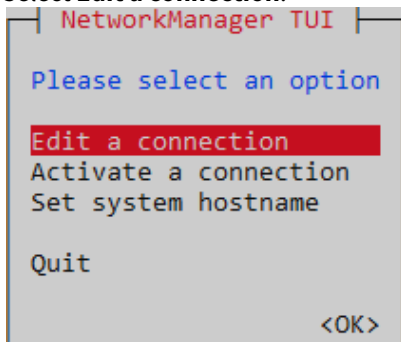
- ✘ After running the `clusterctl install` or `clusterctl node add` commands you cannot change the IP address or hostname of a node.

To configure the connection

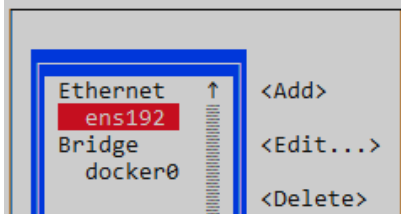
1. Log in to the machine console locally as the `sysadmin` user.
2. Run the `nmtui` tool with `sudo` permissions.

```
sudo nmtui
```

3. Select **Edit a connection**.



4. Press **Enter** and select the Ethernet connection.



5. Press **Enter** to display the **Edit connection** dialog.
6. Change the **IPv4 CONFIGURATION** mode to **Manual**.
7. Select **<Show>** for **IPv4 CONFIGURATION**.

- i** As explained in [IP address requirements](#), Entrust PKI Hub only supports IPv4 and disables IPv6 by default.

8. Press **Enter** to display the connection settings.
9. In the **Addresses** field, enter the preferred IP address and the subnet mask – for example:

```
192.168.100.4/24
```

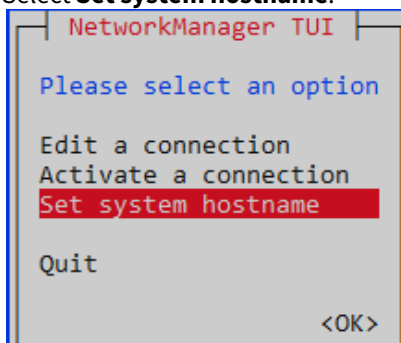
10. In the **Gateway** field, enter the IP address of the default gateway for your network connection.
11. In the **DNS servers** field, enter the IP address of each server for DNS resolution. Separate multiple IP addresses with spaces or commas.

❌ As explained in [DNS requirements](#), Entrust PKI Hub does not support accessing a DNS server through a proxy.

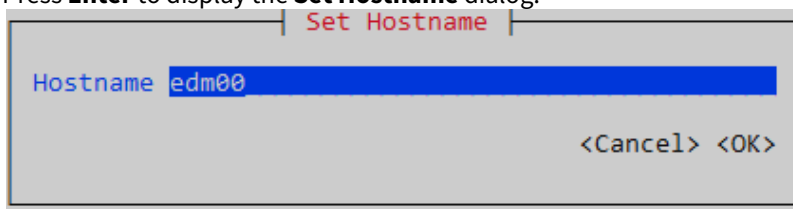
- In the **Search domains** field, enter the domain names you want to use for DNS searches. Separate multiple domains with spaces or commas.

ℹ️ Your system will automatically append the search domains to any unqualified domain names you use in your network, making it easier to access local resources without typing the full domain name every time.

- Select **<OK>** and press **Enter**.
- Select **<Back>** and press **Enter**.
- Select **Set system hostname**.



- Press **Enter** to display the **Set Hostname** dialog.



- In the **Hostname** field, type an [RFC1123](#) compliant hostname – that is:
 - Must consist of lowercase alphanumeric characters, '-' or '.'
 - Must start and end with an alphanumeric character.
- Select **<OK>** and press **Enter**.
- Press **Enter** to confirm the hostname change.
- Select **Quit** and press **Enter**.
- Make the changes effective:
 - Reboot the machine, if you are using the local console
 - Open a new SSH session, if you are using remote SSH access.

Checking the connection of a PKI Hub ISO installation

Run the `nmcli device status` command to check the connection of a machine – for example:

```
$ nmcli device status
DEVICE   TYPE      STATE      CONNECTION
ens192   ethernet  connected  ens192
docker0  bridge    connected (externally)  docker0
lo       loopback  unmanaged  --
```

Check the state is `connected` for each connection.

Run the `ping` command to verify the host can send packets to other hosts.

```
ping <hostname>
```

Where `<hostname>` is the IP address or hostname of another host.

Configuring the boot mode of a PKI Hub ISO installation

Configure the boot mode selected for the PKI Hub ISO image installation.

- [Configuring the BIOS boot on a PKI Hub ISO installation](#)
- [Configuring the UEFI boot on a PKI Hub ISO installation](#)

Configuring the BIOS boot on a PKI Hub ISO installation

As explained in [Disk requirements](#), PKI Hub requires two disks. In the BIOS settings of the machine, ensure the biggest disk is placed first in the boot order. Otherwise, the system will not boot.

Configuring the UEFI boot on a PKI Hub ISO installation

If your machine uses the UEFI boot firmware, you must import and enroll the ELRepo key.

- [Importing the ELRepo key](#)
- [Enrolling the ELRepo key](#)

Importing the ELRepo key

Run the following command to import the ELRepo key distributed with Entrust PKI Hub.

```
sudo mokutil --import /etc/pki/elrepo/SECURE-BOOT-KEY-elrepo.org.der
```

When prompted:

1. Type the password of the `sysadmin` user.
2. Type a password for the key.
3. Confirm the key password.

Enrolling the ELRepo key

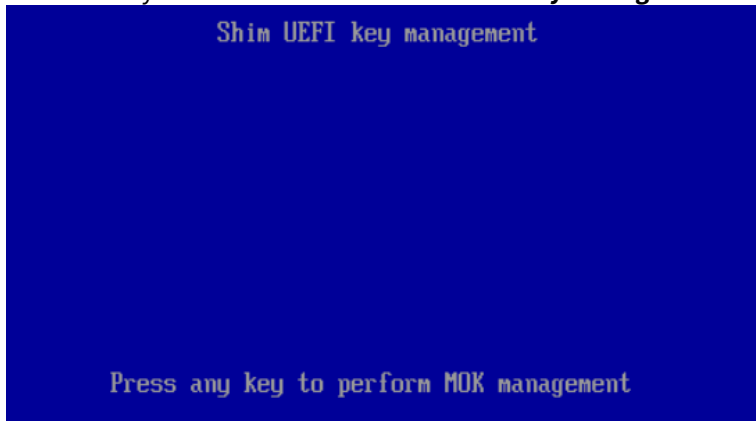
Once imported, enroll the key as explained below.

 See <http://elrepo.org/tiki/SecureBootKey> for more details on enrolling the ELRepo key.

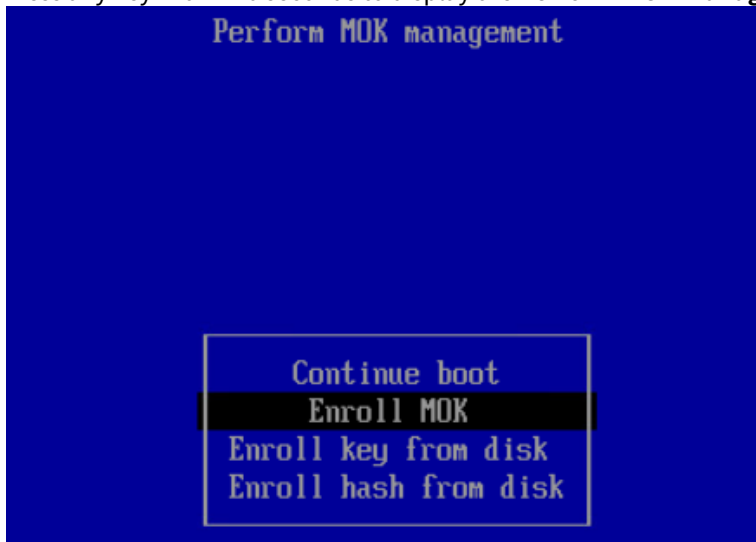
To enroll the ELRepo key

1. Log into the console of the local machine. The following operations do not support a remote console like an SSH client.

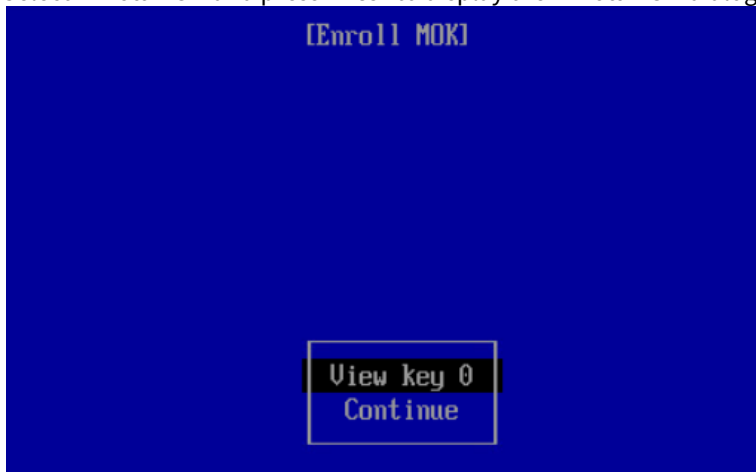
2. Reboot the system and wait for the **Shim UEFI key management** screen.



3. Press any key within 10 seconds to display the **Perform MOK management** dialog.



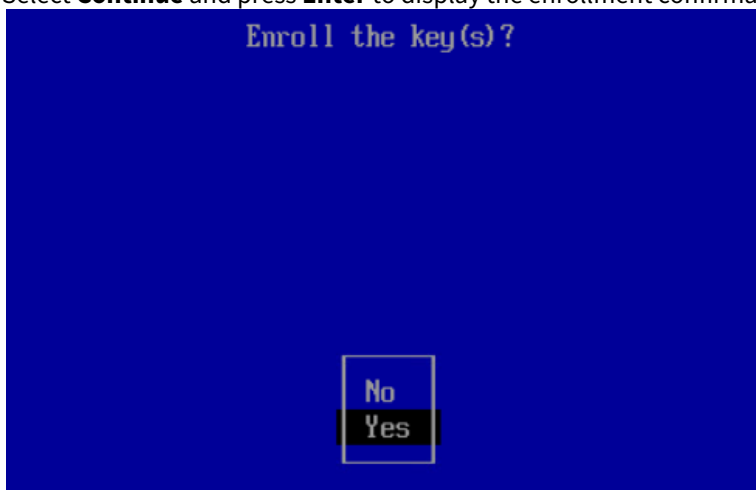
4. Select **Enroll MOK** and press **Enter** to display the **Enroll MOK** dialog.



5. Select **View key 0** and press **Enter** to display the key information.

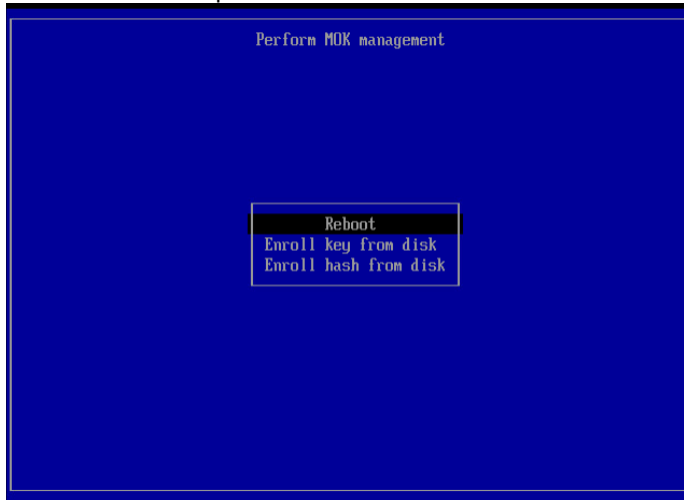


6. Check that the serial number is `0xe9d471cfb4fe136c`.
7. Check that the SHA1 fingerprint is `e1:21:a2:f6:07:2e:f2:94:de:20:0e:6b:5d:1b:49:c0:65:dc:e3:e7`.
8. Press **ESC** to return to the **Enroll MOK** dialog.
9. Select **Continue** and press **Enter** to display the enrollment confirmation dialog.




10. Select **Yes** and press **Enter** to display the password form.
11. Type the key password you selected when importing the ELRepo key.
12. Press **Enter** to return to the **Perform MOK management** dialog.

13. Select **Reboot** and press **Enter**.



Installing the Entrust PKI Hub RAW image on AWS

See below for installing and configuring the Entrust PKI Hub image in the Amazon Web Services (AWS) cloud.

 Refer to docs.aws.amazon.com for advanced configurations not covered in this guide, such as selecting the machine DNS.

To install and configure the PKI Hub image in AWS

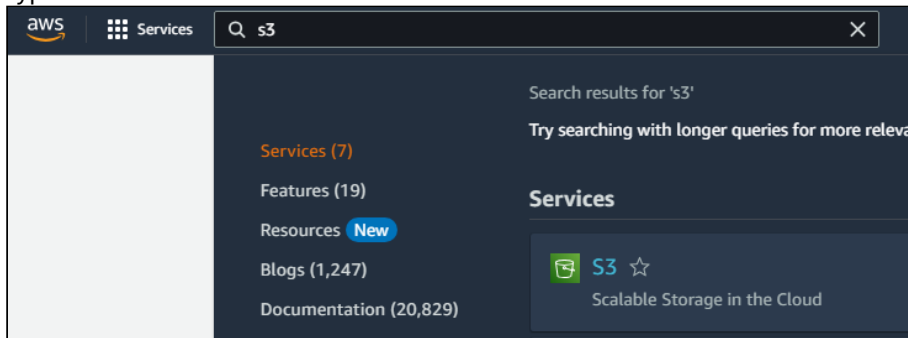
1. Download the Entrust PKI Hub RAW image as explained in [Downloading the Entrust PKI Hub image](#).
2. Log into <https://console.aws.amazon.com> as a user with permission to:
 - Create and manage S3 buckets, roles, policies, snapshots, images, and EC2 instances.
 - Run AWS CLI commands using a locally installed AWS CLI or the AWS ShellCloud.
3. Perform the steps explained below.
 - [Creating an S3 bucket](#)
 - [Uploading the RAW image](#)
 - [Configuring the IAM policy](#)
 - [Creating an IAM role](#)
 - [Creating the snapshot configuration file](#)
 - [Preparing the command-line interface](#)
 - [Importing the snapshot](#)
 - [Creating an AMI from the snapshot](#)
 - [Creating the EC2 instance](#)
 - [Opening a session into AWS](#)

Creating an S3 bucket

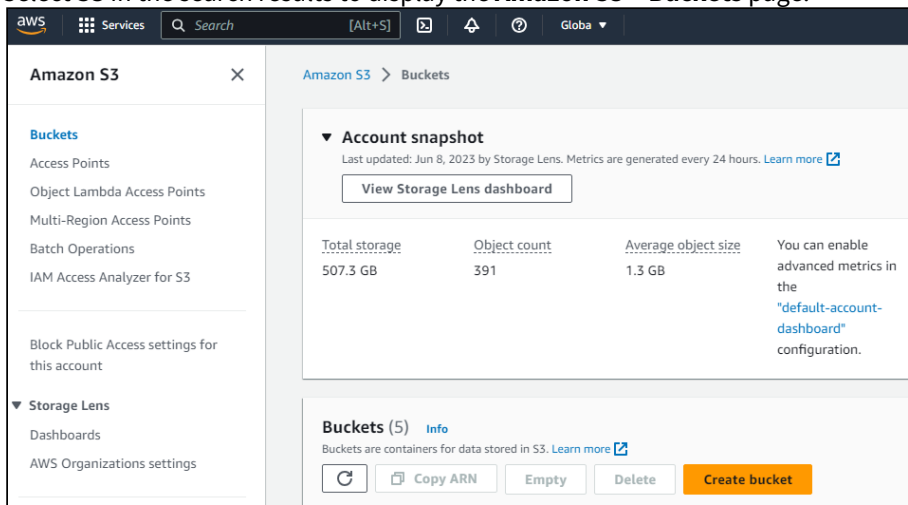
If you don't have an S3 bucket, create a new one as explained below.

To create an S3 bucket

1. Type "S3" in the search box of the AWS console.



2. Select **S3** in the search results to display the **Amazon S3 > Buckets** page.



3. Click **Create a bucket**.
4. Enter a name for the new bucket.
5. Select an AWS region for the bucket.

✘ All the resources created to deploy Entrust PKI Hub in Amazon Web Service must share the same region.

6. For the other S3 settings, you can leave the default values.
 - Object Ownership
 - Block Public Access settings for this bucket
 - Bucket Versioning
 - Default encryption
 - Advanced settings
7. Click **Create bucket**.

Uploading the RAW image

Upload the Entrust PKI Hub RAW image file to Amazon Web Services.

To upload the RAW image

1. Navigate to the **Amazon S3 page > Buckets** page of the AWS console.
2. Click the name of an S3 bucket. As explained in [Creating an S3 bucket](#), you can select an existing bucket or create a new one.
3. In the S3 bucket details page, click **Upload** to display the file upload form.

4. Click **Add Files**.
5. Select the Entrust PKI Hub image file with `.raw` extension.
6. Click **Upload** at the bottom of the page to start the upload process.

Configuring the IAM policy

For granting permission to the S3 bucket, create an IAM (Identity and Access Management) policy or reuse an existing one.

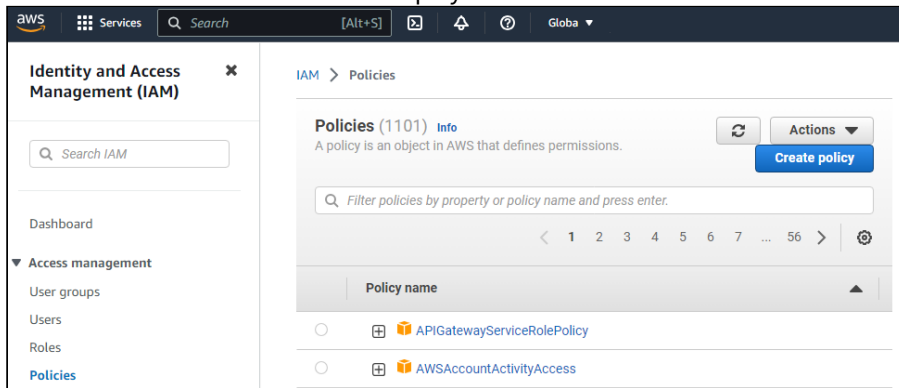
- [Creating a new IAM policy](#)
- [Updating an existing IAM policy](#)

Creating a new IAM policy

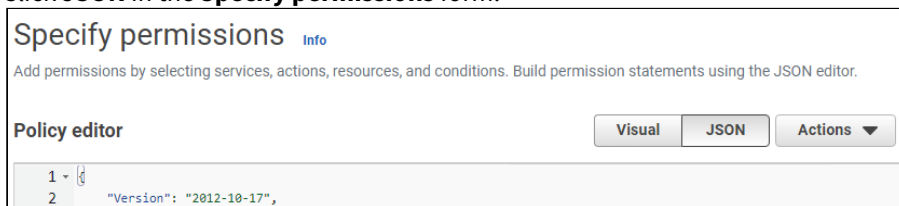
See below for creating an IAM policy granting permission to the S3 bucket.

To create an IAM policy

1. Type "IAM" in the search box of the AWS console.
2. Select **IAM** in the search results to display the IAM dashboard.



3. Select **Access management > Policies** in the navigation sidebar.
4. In the content pane, click the name of an existing IAM policy or click **Create policy** to create a new one.
5. Click **JSON** in the **Specify permissions** form.



6. Paste the following JSON code in the **Policy editor** field.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
      ]
    }
  ]
}

```

```

    ],
    "Resource": [
      "arn:aws:s3:::$S3_BUCKET_NAME",
      "arn:aws:s3:::$S3_BUCKET_NAME/*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "ec2:ModifySnapshotAttribute",
      "ec2:CopySnapshot",
      "ec2:RegisterImage",
      "ec2:Describe*"
    ],
    "Resource": "*"
  }
]
}

```

7. In the JSON code, replace `$S3_BUCKET_NAME` with the name of the S3 bucket selected when [Creating an S3 bucket](#).
8. Click **Next**.
9. Enter a name and an optional description for the new policy.
10. Click **Create policy**.

Updating an existing IAM policy

See below for how to update an existing IAM policy for granting permission to the S3 bucket.

To update an IAM policy

1. Type "IAM" in the search box of the AWS console.
2. Select **IAM** in the search results to display the IAM dashboard.



3. Select **Access management > Policies** in the navigation sidebar.
4. In the content pane, click the **+** expand button for an existing IAM policy.
5. Click **Edit**.
6. In the policy editor field, add the following code to the `Resource` array.

```

"arn:aws:s3:::$S3_BUCKET_NAME",
"arn:aws:s3:::$S3_BUCKET_NAME/*"

```

7. In the code, replace `$S3_BUCKET_NAME` with the name of the S3 bucket selected when [Creating an S3 bucket](#).

8. Click **Next**.
9. Click **Save changes**.

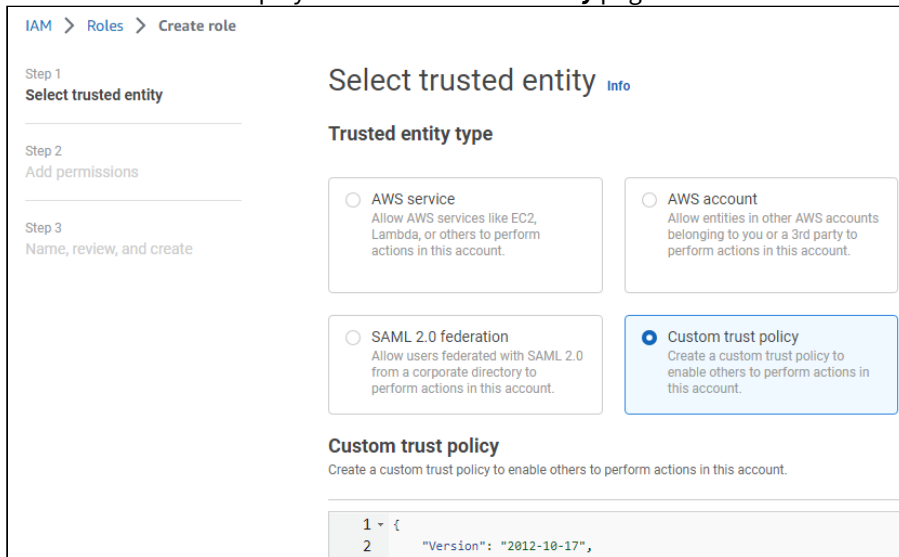
Creating an IAM role

Create an IAM (Identity and Access Management) role for the policy described in [Configuring the IAM policy](#).

✘ As indicated in the steps below, the value of the `sts:Externalid` field and the role name must both be exactly `vmimport`.

To create an IAM role

1. Type "IAM" in the search box.
2. Select IAM in the search results to display the IAM dashboard.
3. Select **Access management > Roles** in the navigation sidebar.
4. Click **Create role** to display the **Select trusted entity** page.



The screenshot shows the 'Select trusted entity' page in the AWS IAM console. The 'Trusted entity type' section has four options: 'AWS service', 'AWS account', 'SAML 2.0 federation', and 'Custom trust policy'. The 'Custom trust policy' option is selected. Below this, there is a section for 'Custom trust policy' with a text area containing the following JSON code:

```

1 {
2   "Version": "2012-10-17",

```

5. Under **Trusted entity type**, click **Custom trust policy**.
6. Paste the following code under **Custom trust policy**.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "vmie.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:Externalid": "vmimport"
        }
      }
    }
  ]
}

```

```
}  
  ]  
}
```

7. Click **Next**.
8. In the **Add permissions** page, select the policy described in [Configuring the IAM policy](#).
9. Click **Next** to display the **Role details** page.
10. In the **Role name** field, type `vmimport`.
11. Click **Create role**.

Creating the snapshot configuration file

In your local machine, create a `container.json` file with the following contents.

```
{  
  "Description": "Entrust PKI Hub AMI file",  
  "Format": "raw",  
  "UserBucket": {  
    "S3Bucket": "$AWS_S3_BUCKET",  
    "S3Key": "$AMI_FILE"  
  }  
}
```

In the file contents, replace:

- `$AWS_S3_BUCKET` with the name of the S3 bucket described in [Creating an S3 bucket](#).
- `$AMI_FILE` with the name of the image file selected when [Uploading the RAW image](#).

For example:

```
{  
  "Description": "Entrust PKI Hub AMI file",  
  "Format": "raw",  
  "UserBucket": {  
    "S3Bucket": "pki-hub-01",  
    "S3Key": "pki-hub-1.0.raw"  
  }  
}
```

Preparing the command-line interface

To run AWS commands in your machine, download and install the AWS CLI as explained in:

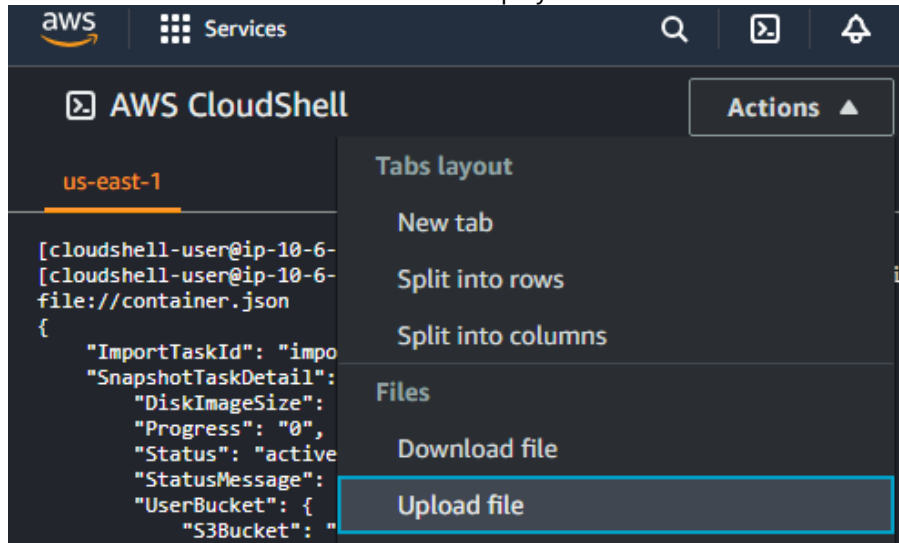
<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

Alternatively, you can use the ShellCloud provided by the AWS console. This option requires uploading the configuration file as explained below.

To upload the snapshot configuration file

1. Type "shell" in the search box of the AWS console.

2. Select **CloudShell** in the search results to display the online AWS shell.



3. In the options menu, select **Actions > Upload file**.
4. Select the `container.json` file described in [Creating the snapshot configuration file](#).
5. Click **Upload**.

Importing the snapshot

Run the following AWS command to import the Entrust PKI Hub image as an EC2 snapshot.

```
aws ec2 import-snapshot --disk-container file://container.json
```

For example:

```
$ aws ec2 import-snapshot --disk-container file://containers.json
{
  "ImportTaskId": "import-snap-03b38da24cb5fdde1",
  "SnapshotTaskDetail": {
    "DiskImageSize": 0.0,
    "Progress": "0",
    "Status": "active",
    "StatusMessage": "pending",
    "UserBucket": {
      "S3Bucket": "edm-01",
      "S3Key": "edm-1.0.0_2023-06-06-10_38_16.raw"
    }
  },
  "Tags": []
}
```

Use the value of the `ImportTaskId` field to check the status of the import process.

```
aws ec2 describe-import-snapshot-tasks --import-task-ids <ImportTaskId>
```

For example:

```
aws ec2 describe-import-snapshot-tasks --import-task-ids import-snap-03b38da24cb5fdde1
{
  "ImportSnapshotTasks": [
    {
      "ImportTaskId": "import-snap-03b38da24cb5fdde1",
      "SnapshotTaskDetail": {
        "DiskImageSize": 10740563968.0,
        "Format": "raw",
        "SnapshotId": "snap-03ea2ef99eb98d255",
        "Status": "completed",
        "UserBucket": {
          "S3Bucket": "edm-01",
          "S3Key": "edm-2.0.0_2023-06-06-10_38_16.raw"
        }
      }
    },
    "Tags": []
  ]
}
```

In the command output, check the value of the `Status` field.

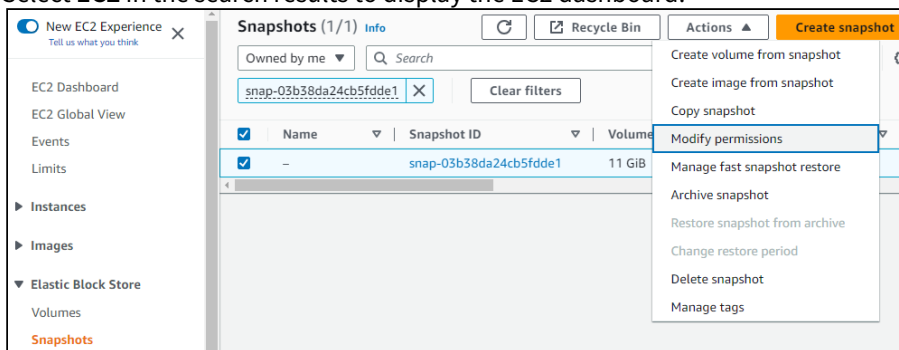
- If this value is `active`, rerun the command after 5 minutes to recheck the status.
- If this value is `completed`, the import process has already finished. Copy the `SnapshotId` value to use it in the next step [Creating an AMI from the snapshot](#).

Creating an AMI from the snapshot

Create an Amazon Machine Image (AMI) from the imported snapshot.

To create the AMI

1. Type "EC2" in the search box.
2. Select **EC2** in the search results to display the EC2 dashboard.



3. Select **Elastic Block Store > Snapshots** in the navigation sidebar.
4. In the search box, paste the `SnapshotId` value returned when [Importing the snapshot](#).
5. Press **Enter**.
6. In the content pane, check the box for the newly imported snapshot.

7. In the options menu, select **Actions > Create image from snapshot**.
8. Configure the following settings.
 - [Image settings](#)
 - [Block device mappings](#)
9. Click **Create image**.

Image settings

Enter a name and an optional description for the image. For the other settings, you can leave the default values.

Block device mappings

Configure the root volume and add one for the `etcd` daemon. See below for the required settings.

Setting	Root volume	etdc volume
Size	1 TiB or more	15 GB or more
Volume type	SSD (Solid-state Drive)	General Purpose SSD (gp3)

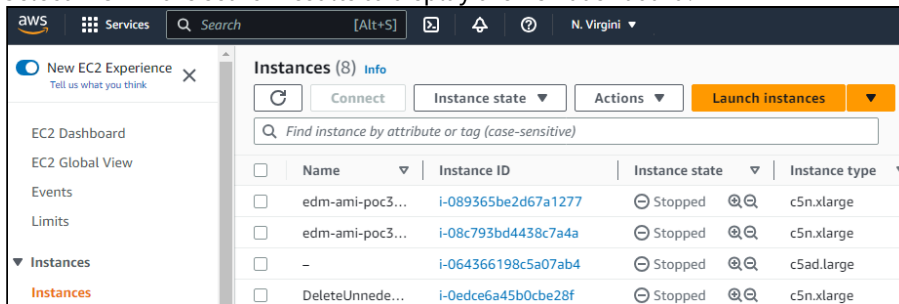
For the other settings, you can leave the default values.

Creating the EC2 instance

Create an EC2 instance for running the Entrust PKI Hub image.

To create the EC2 instance

1. Type "EC2" in the search box.
2. Select **IEC2** in the search results to display the EC2 dashboard.



3. Select **Instances > Instance** in the navigation sidebar.
4. In the options menu, click **Launch instance**.
5. Configure the following settings.
 - [Name and tags > Name](#)
 - [Application and OS Images \(Amazon Machine Image\)](#)
 - [Instance type](#)
 - [Key pair \(login\)](#)
 - [Network settings > Firewall \(security groups\)](#)
 - [Configure storage](#)
 - [Advanced Details > User data](#)
6. Click **Launch instance**.

Name and tags > Name

Enter a name for the new EC2 instance.

Application and OS Images (Amazon Machine Image)

Select the machine described in [Creating an AMI from the snapshot](#).

Instance type

Select an EC2 instance type. See the table below for the minimum type required by each installation mode.


Installation	Number of deployed solutions	Minimum instance type
Single-node	1	c5n.xlarge
Single-node	>1	c5n.2xlarge
Multi-node	Any	c5n.xlarge

Key pair (login)

Select an existing key pair, or create a new one for SSH connections.

Network settings > Firewall (security groups)

Select or create a security group with permission to open the ports described in [Required open ports](#).

 See <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html> for how to create a security group.

Configure storage

Select the default volume size or higher.

Advanced Details > User data

Paste the following code.

```
#!/bin/bash
/opt/entrust/scripts/configure-os.sh
```

 This code will allow the `sysadmin` Entrust PKI Hub administrator to log in using the SSH key.

Opening a session into AWS

After creating and configuring the ECS instance:

1. Refresh the EC2 instance list until the instance status changes from **Initializing** to **2/2 checks passed**.
2. Wait a few minutes more.

You can then connect to the instance with SSH:

1. Copy the instance IP.
2. Open an SSH session into the instance.
3. Authenticate with the following credentials.
 - The username `sysadmin` of the default Entrust PKI Hub administrator.
 - The SSH key selected when [Creating the EC2 instance](#) .
4. When prompted, change the `changeme` initial password with a password meeting the requirements described in [Password policy CIS benchmarks](#).

i If you encounter an error during the initial connection, please try again after some time as the machine is currently being configured.

Installing the Entrust PKI Hub VHD image on Azure

See below for installing and configuring an Entrust PKI Hub image on the Microsoft Azure cloud.

i Refer to learn.microsoft.com/azure for advanced configurations not covered in this guide, such as selecting the machine DNS.

To install and configure the PKI Hub image in Azure

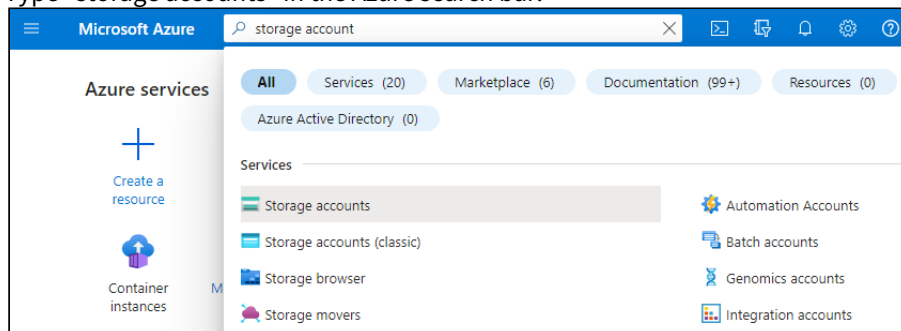
1. Download the Entrust PKI Hub VHD image as explained in [Downloading the Entrust PKI Hub image](#).
2. Log into <https://portal.azure.com> as a user with permission to create and manage storage accounts, images, network rules, SSH keys, and virtual machines.
3. Perform the steps described below.
 - [Creating the Azure storage account](#)
 - [Uploading the VHD image file to Azure](#)
 - [Creating the Azure image](#)
 - [Creating the Azure network rules](#)
 - [Creating the SSH key for Azure](#)
 - [Creating the Azure virtual machine](#)
 - [Opening a session into Azure](#)

Creating the Azure storage account

Select an existing Azure storage account or create a new one as explained below.

To create an Azure storage account

1. Type "storage accounts" in the Azure search bar.



2. Select **Storage accounts** in the search results.
3. Click+ **Create** on the **Storage accounts** page.
4. Configure the following settings on the **Create a storage account** page.
 - [Subscription](#)
 - [Resource group](#)
 - [Storage account name](#)
 - [Region](#)
5. Click **Review** to display the configured settings.
6. Click **Create** to create the storage account.

Subscription

Select your Azure user subscription.

Resource group

Select an existing resource group or create a new one.

✘ All the resources created to deploy Entrust PKI Hub in Azure must share the same resource group.

Storage account name

Enter a name for the new storage account name.

Region

Select a region for the new storage account.

⚠ All the resources created to deploy Entrust PKI Hub in Azure must share the same region.

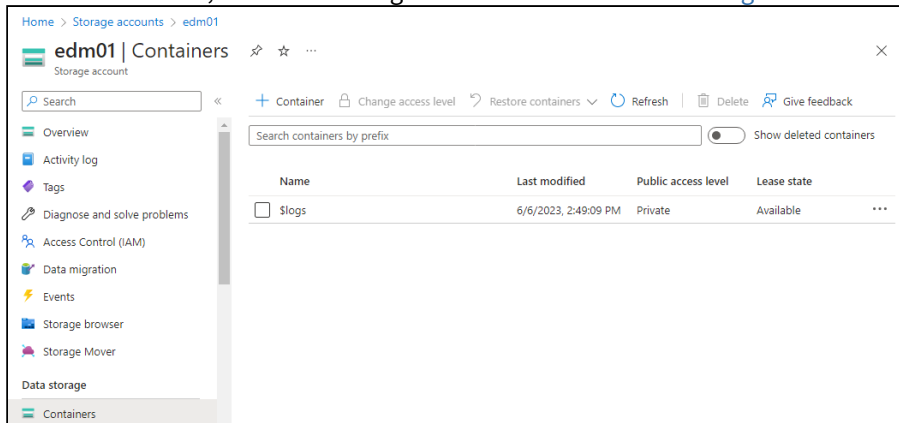
Uploading the VHD image file to Azure

Upload to Azure the Entrust PKI Hub image file with `.vhd` extension.

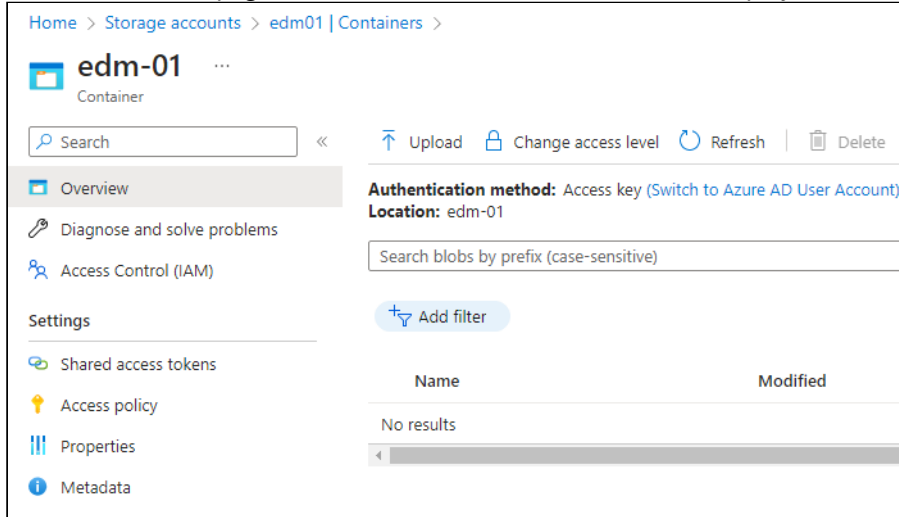
i See [Downloading the Entrust PKI Hub image](#) for how to obtain this file.

To upload the VHD image file

1. In the Azure Portal, select the storage account described in [Creating the Azure storage account](#).



2. In the sidebar menu of the storage settings page, select **Data storage > Containers**.
3. On the **Containers** page, click **+ Container**
4. Enter a name for the new container and click **Create**.
5. On the **Containers** page, click the name of the new container to display the container details.



6. On the container details page, click **Upload**.
7. Select the Entrust PKI Hub image file with `.vdh` extension and wait while the file uploads.

Creating the Azure image

Create an Azure image

- [Creating the Azure image with Azure Portal](#)
- [Creating the Azure image with Azure CLI](#)

Creating the Azure image with Azure Portal

See below for how to create the Entrust Development Manager image using the Azure Portal.

To create the image in the Azure Portal

1. Type "images" in the Azure Portal search bar.
2. Click **Images** on the search results.
3. Click **+ Create** on the **Images** page.
4. Configure the following settings in the **Create an image** page.
 - [Project details](#)
 - [Instance details](#)
 - [OS disk](#)
5. Click **Review + create** to validate the image settings.
6. Click **Create** to create the new image.

Project details

Configure the following settings under **Project details**.

Setting	Value
Subscription	Select your Azure subscription.
Resource group	Select the same resource group selected when Creating the Azure storage account .

Instance details

Configure the following settings under **Instance details**.

Setting	Value
Name	Enter a unique name for the new image.
Region	Select the same region selected when Creating the Azure storage account .

OS disk

Configure the following settings under **OS disk**.

Setting	Value
OS type	Select Linux .
Storage blob	Select the VHD image described in Uploading the VHD image file to Azure .
Account type	Select Standard SSD .
Host caching	Select Read-only .

Creating the Azure image with Azure CLI

Run the following command to create the image with the Azure command-line interface.

```
az image create --resource-group $GROUP --location $LOCATION --name $IMAGE \
--source $SOURCE --os-type linux --storage-sku Standard_LRS --os-disk-caching
ReadOnly
```

See the table below for a description of each parameter.

Option	Value
GROUP	The name of the resource group created in Creating the Azure storage account .
IMAGE	The name of the VHD image previously uploaded in Creating the Azure storage account .
LOCATION	The location of the resource group created in Creating the Azure storage account .
SOURCE	The OS disk source.

Creating the Azure network rules

Create a Network Security Group with rules granting access to the [Required open ports](#).


Creating the SSH key for Azure

Create or upload a key for authenticating the Azure machine administrator in remote SSH connections.

Creating the Azure virtual machine

Create an Azure Virtual Machine for running Entrust PKI Hub.

- [Creating the Azure virtual machine with Azure Portal](#)
- [Creating the Azure virtual machine with Azure CLI](#)

 As explained in [IP address requirements](#), all the nodes of a multi-node installation require a static hostname and IP address.

Creating the Azure virtual machine with Azure Portal

When using the Azure portal to create the virtual machine, set the following configuration.

- [Disk](#)
- [Networking](#)
- [Advanced](#)

 You can leave the default values for the settings not listed on this page.

Basics

Set the following values in the **Basics** tab of the **Create a virtual machine** page.

Setting	Value
Project details / Subscription	Select your Azure subscription.
Project details / Resource group	Select the resource group described in Creating the Azure storage account .
Instance details / Virtual machine name	Enter a name for the new virtual machine.
Instance details / Region	Select the region shared by the rest of Azure resources.
Instance details / Image	Select the image described in Creating the Azure image .
Instance details / Size	Select one of the following: <code>Standard_D4s_v3</code> , <code>Standard_D4ds_v5</code> , <code>Standard_D4ds_v4</code> , <code>Standard_D4as_v4</code> , <code>Standard_F4s</code> .
Administrator account / Authentication type	Select SSH public key .
Administrator account / SSH public key source	Select the key described in Creating the SSH key for Azure .
Administrator account / Key pair name	Select the name of the key described in Creating the SSH key for Azure .
Inbound port rules / Public inbound ports	Select None .
Licensing type / License type	Select Other .

Disk

Set the following values in the **Disk** tab of the **Create a virtual machine** page.

Setting	Value
OS disk / OS disk size	Select 1 TiB (P30) or higher.
OS disk / OS disk type	Select Premium SSD (locally-redundant storage) or higher.

Under **Data disks**, click **Create and attach a new disk**, and set the following value.

Setting	Value
Size	15 GiB or higher

Networking

Set the following values in the **Networking** tab of the **Create a virtual machine** page.

Setting	Value
NIC network security group	Select Advanced .
Configure network security group	Select the network security group described in Creating the Azure network rules .

Advanced

Paste the following code in the **Custom data** field of the **Advanced** tab.

```
#!/bin/bash
/opt/entrust/scripts/configure-os.sh
```

 This code will allow the `sysadmin` Entrust PKI Hub administrator to log in using the SSH key.

Creating the Azure virtual machine with Azure CLI

Run the following command to create the virtual machine with the Azure command-line interface.

```
az vm create --resource-group $GROUP --location $LOCATION --name $VM --image $IMAGE
--size $SIZE --storage-sku Premium_LRS --os-disk-size-gb $DISK_SIZE --os-disk-caching
ReadOnly --data-disk-sizes-gb $EXTRA_DISK_SIZE --data-disk-caching ReadOnly --
authentication-type ssh --admin-username azureuser --ssh-key-name $SSH_KEY --nsg $NSG
--custom-data $CUSTOM_DATA
```

See below for a description of each parameter.

- [GROUP](#)
- [LOCATION](#)
- [VM](#)
- [IMAGE](#)
- [SIZE](#)
- [DISK_SIZE](#)
- [EXTRA_DISK_SIZE](#)
- [SSH_KEY](#)
- [NSG](#)
- [CUSTOM_DATA](#)

GROUP

The name of the resource group created in [Creating the Azure storage account](#).

LOCATION

The location of the resource group created in [Creating the Azure storage account](#).

VM

The name of the Virtual Machine that will be created.

IMAGE

The name of the VHD image previously uploaded in [Creating the Azure storage account](#).

SIZE

One of the following:

- Standard_D4s_v3
- Standard_D4ds_v5
- Standard_D4ds_v4
- Standard_D4as_v4
- Standard_F4s

DISK_SIZE

1024 GiB or more. For example:

```
--os-disk-size-gb 1024
```

EXTRA_DISK_SIZE

15 GiB or more. For example:

```
--data-disk-size-gb 15
```

SSH_KEY

The name of the SSH key pair created in [Creating the SSH key for Azure](#).

NSG

The network security group created in [Creating the Azure network rules](#).

CUSTOM_DATA

The path of a local text file with the following contents.

```
#!/bin/bash
/opt/entrust/scripts/configure-os.sh
```


Setting	single-node	multi-node
Requirements	Does not need the disk performance requirements described in Disk requirements . Specifically, fsync latency is not an issue in this mode.	All the Requirements .
Supported number of nodes	One	One or more. See Recommended number of nodes for details.
Supported operations	You cannot perform the operations described in Adding nodes , Backing up the state , Recovering from disaster , or Restoring the state .	All
Supported updates	You cannot upgrade to a newer version or migrate to a multi-node installation.	All

Replacing the default TLS certificate


During installation, Entrust PKI Hub generates an insecure self-signed certificate for securing communications with Grafana, the Management Console, and the solution services. You must replace this certificate before running Entrust PKI Hub in a production environment.

- [TLS certificate subject names](#)
- [TLS certificate algorithms](#)
- [Issuing the TLS certificate](#)
- [Installing the TLS certificate](#)
- [Reusing as CA Gateway TLS certificate](#)

TLS certificate subject names

The Entrust PKI Hub TLS certificate must include one of the following fields.

- DNS Subject Alternative Name (SAN)
- Subject Common Name (CN)

 When both fields are present, the Subject Common Name is ignored.

TLS certificate algorithms

The Entrust PKI Hub TLS certificate must be generated using either:

- The RSA algorithm with a key length of 2048 bits or more.
- The ECDSA algorithm with a P-256 elliptic curve.

Issuing the TLS certificate

To get the Entrust PKI Hub TLS certificate, you can:

- Use your corporate PKI.
- Purchase the certificate at store.entrust.com. To generate the certificate request, Entrust provides an online form at entrust.com/resources/certificate-solutions/tools/open-ssl-csr-command-builder

Installing the TLS certificate

Run the `clusterctl certificate` command to install the Entrust PKI Hub TLS certificate.

✘ When running Entrust PKI Hub in high availability, also install the TLS certificate in the load balancer.

Reusing as CA Gateway TLS certificate

If the CA Gateway solution is deployed, you can use the same TLS certificate for Entrust PKI Hub and CA Gateway.

ℹ See the CA Gateway configuration reference for how to select this TLS certificate in CA Gateway.

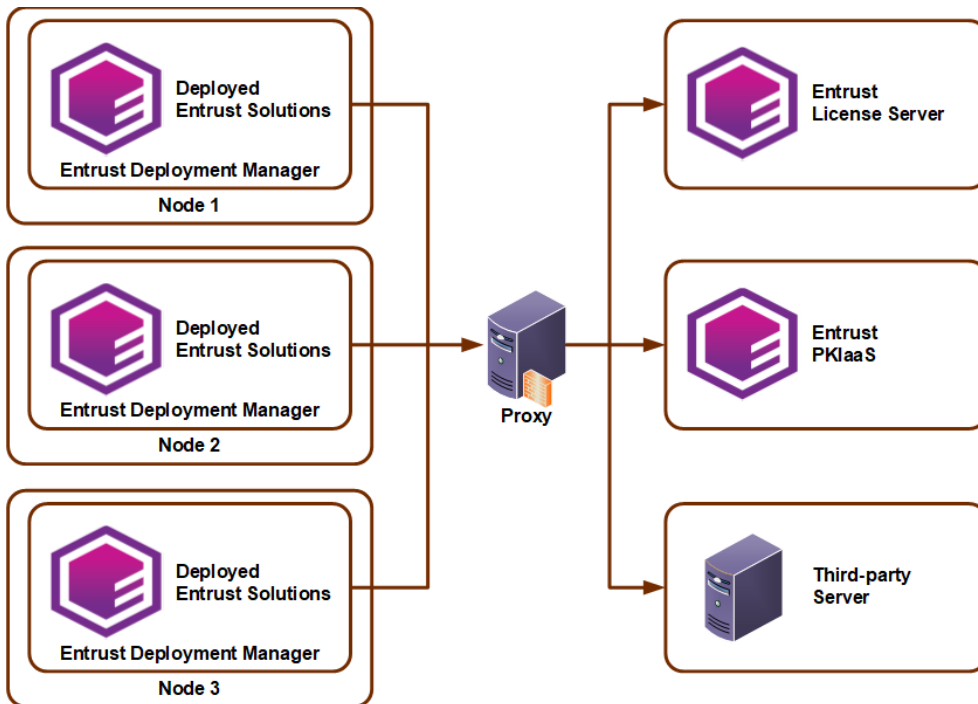
Configuring the proxy

Some solutions require outgoing connections. For example, with:

- Entrust PKIaaS
- An Entrust license server.
- A third-party server integrated with the solution

If these connections pass through a proxy server, run the `clusterctl proxy set` command to configure this proxy server in Entrust PKI Hub.

✘ As explained in [DNS requirements](#), Entrust PKI Hub does not support accessing a DNS server through a proxy.



Changing the keyboard layout

The default keyboard layout for Entrust PKI Hub Installations is `en-US`. Run the following command to list the available keyboard layouts.

```
localectl list-locales
```

Run the following command to set the `<layout>` keyboard layout.

```
sudo localectl set-keymap <layout>
```

For example:

```
$ sudo localectl set-keymap es-ES
```

Changing the operating system timezone

The default timezone for Entrust PKI Hub Installations is UTC. Run the following command to list the available time zones.

```
timedatectl list-timezones
```

Run the following command to set the `<timezone>` timezone.

```
sudo timedatectl set-timezone <timezone>
```

For example


```
$ sudo timedatectl set-timezone Europe/Madrid
```

Configuring time synchronization

The Entrust PKI Hub installation configures

`chrony`

to use the NTP server provided by DHCP.

 In multi-node installations, all the nodes must have synchronized dates to prevent communication errors.

To modify this default configuration, edit the `chrony` configuration file in all the installation nodes.

```
/etc/chrony.conf
```

Do not modify the following lines in any case.

```
bindcmdaddress 0.0.0.0  
cmdallow all
```

 See a reference of the `chrony` parameters at <https://chrony-project.org/doc/4.4/chrony.conf.html>

Manually starting starting the chrony service

Run the following command to manually start the `chrony` service.

```
sudo systemctl restart chronyd.service
```

 Re-run this command after each node restart.

Configuring an nShield HSM

Perform the following configuration steps if any solution requires an Entrust nShield HSM (Hardware Security Module). Skip them if you intend to use an HSM from another vendor.

- [Selecting the platform for creating the Entrust nShield Security World](#)
- [Selecting the drivers for creating the Entrust nShield Security World](#)
- [Adding a cknfastrc file to the Entrust nShield Security World](#)
- [Configuring kmdata/config/config in Entrust nShield Security World](#)

- [Registering Entrust PKI Hub nodes as Entrust nShield clients](#)

i For a complete guide on Security World, see <https://nshielddocs.entrust.com/security-world-docs/v12.80/connect-ug-nix/create-manage-security-world.html#CreatingSecurityWorld>

Selecting the platform for creating the Entrust nShield Security World

You can create the Entrust nShield Security World on the machine running the Timestamping Authority solution, or on another machine of your choice.

Selecting the drivers for creating the Entrust nShield Security World

Section [HSM requirements](#) details the version of the built-in client drivers Entrust solutions use to connect with Entrust nShield HSMs. To avoid potential incompatibilities, use client drivers of the same version when creating the Entrust nShield Security World.

Adding a cknfastrc file to the Entrust nShield Security World

To use the `cknfastrc` file in Timestamping Authority:

1. Copy the file into the Security World `kmdata` folder that will be imported later as part of the Timestamping Authority configuration.
2. Edit the file and add the following line:

```
CKNFAST_LOADSHARING=1
```

3. Save the file changes.

Configuring kmdata/config/config in Entrust nShield Security World

The following parameters in the `kmdata/config/config` file only support the default value.

Parameter	Default
<code>impath_addr</code>	0.0.0.0
<code>impath_port</code>	9004

i To use these default values, simply omit the parameters in the configuration.

Registering Entrust PKI Hub nodes as Entrust nShield clients

Entrust nShield requires registering each Entrust PKI Hub node as a client. When using an Entrust nShield HSM, repeat the below steps for each node.

To register an Entrust PKI Hub node as Entrust nShield client

1. Run the client registration wizard as explained in <https://nshielddocs.entrust.com/security-world-docs/v12.80/connect-ug-nix/configure.html#ConfigureConnectClient>
2. When prompted **Please enter your client IP address**, type the node IP address and click **Yes**.
3. When prompted **Do you want to save the IP in the config?** click **Yes**.
4. When prompted **Please choose the client permissions** click **Unprivileged**.
5. When prompted **Do you want secure authentication enabled on this client?** click **No**.

6 Logging into the Management Console

Logging into the web Management Console to manage Entrust solutions and browse logs.

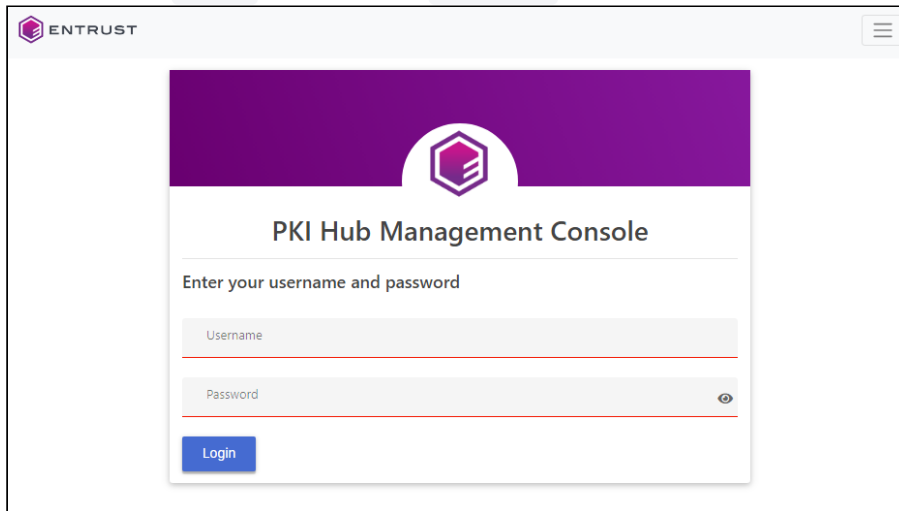
To log into the Management Console

1. Open the following URL in a Web browser.

```
https://<machine>/management-console
```

Where `<machine>` is the IP address or domain name of the machine hosting Entrust PKI Hub.


2. Log in with the `admin` username and `changeme` password.



3. After the first login, you will be prompted to change these initial credentials. Fill in the **Change Password** form and click **SAVE**.
4. Perform the actions described in the following sections.

7 Setting or updating the license

The PKI Hub license determines the Entrust solutions you are allowed to run on PKI Hub. See below for how to set or update this license using the Management Console.

 Alternatively, you can set the license with the `clusterctl license import` command.


To set or update the license

1. Log into the Management console as explained in [Logging into the Management Console](#).
2. Click **License** in the sidebar.
3. Click **Select File** in the content pane and select the license file with the `.lic` extension.
4. Check the details of the uploaded license.

8 Starting up Entrust solutions

See below for configure, deploy and manage the Entrust solutions distributed with Entrust PKI Hub.

- [Starting up Certificate Authorities](#)
- [Starting up CA Gateway](#)
- [Starting up Certificate Enrollment Gateway](#)
- [Starting up Certificate Hub](#)
- [Starting up Timestamping Authority](#)
- [Starting up Entrust Validation Authority](#)
- [Starting up Entrust log-forwarder](#)

 As explained in [Setting or updating the license](#), the user license determines the Entrust solutions you can activate.

Starting up Certificate Authorities

See below for starting up the Certificate Authorities solution.

- [Preparing the Certificate Authorities deployment](#)
- [Configuring and deploying Certificate Authorities](#)
- [Creating Certificate Authority tenants](#)
- [Creating Certificate Authority instances](#)
- [Issuing certificates with Certificate Authority instances](#)
- [Changing the HSM vendor](#)

See [Browsing logs with Grafana](#) for how to browse Certificate Authorities logs.

Preparing the Certificate Authorities deployment

Prepare the deployment of the Certificate Authorities solution as explained in the following sections.

- [Creating the Certificate Authorities database](#)
- [Verifying port access for Certificate Authorities](#)

Creating the Certificate Authorities database

Create a database in a PostgreSQL 15+ DBMS and restrict incoming connections to those originating from the PKI Hub host.

To restrict incoming connections on a PostgreSQL database

1. Edit the following PostgreSQL configuration file.

```
pg_hba.conf
```

2. Add the following line.

```
host    all    all    <host>    scram-sha-256
```

Where `<host>` is the IP address of the Entrust PKI Hub host.

Verifying port access for Certificate Authorities

In addition to the ports listed in [Required open ports](#), ensure no network restriction blocks access to the following ports.

- 4443
- 7443
- 8880

i The deployment of the Certificate Authorities solution automatically opens these ports in the firewall of the machines hosting Entrust PKI Hub.

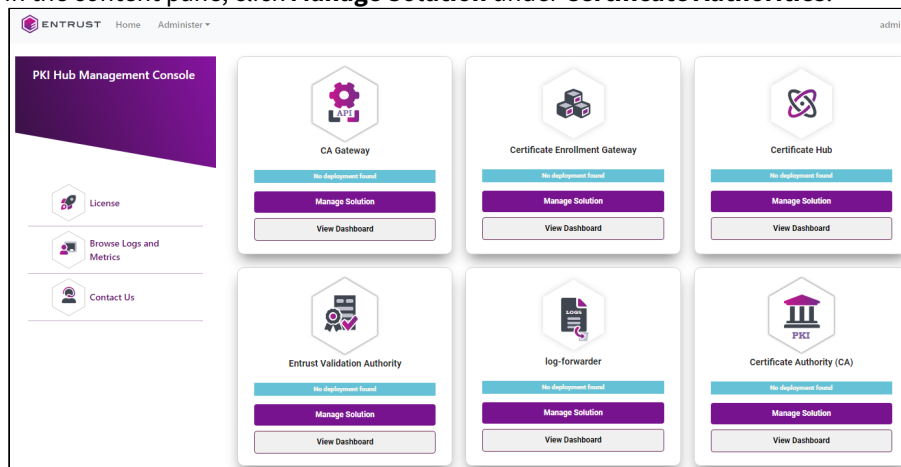
Configuring and deploying Certificate Authorities

See below for configuring and deploying the Certificate Authorities solution with the Management Console.

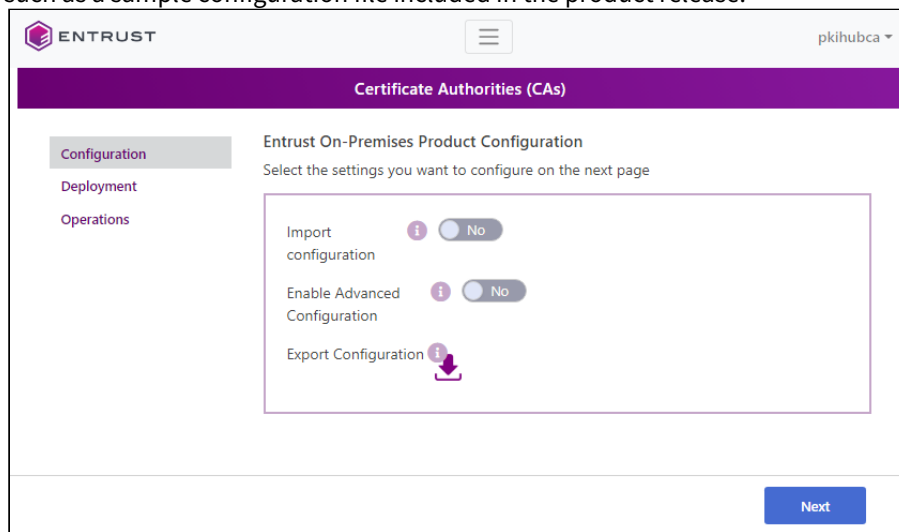
i Repeat the following steps each time a configuration update is required. Do not forget to click **Deploy** to make the changes effective.

To configure and deploy Certificate Authorities with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).
2. In the content pane, click **Manage Solution** under **Certificate Authorities**.



3. Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



4. Active the **Enable Advanced Configuration** if you want to configure the full set of configuration parameters supported by the solution.
5. Click **Next**.
6. Configure the solution settings described in the following sections.
 - [Database](#)
 - [HSM](#)
 - [General](#)
7. Click **Validate** to validate the configured settings.
8. Correct any detected configuration error until the **Validate** option displays no warnings.
9. Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
10. Click **Deploy**.
11. Check the deployment information. Specifically, The **Important Information** field displays the URI of the online endpoints.
12. Back up the solution data in case a restore is required.
 - Select the **Configuration** command in the sidebar and click **Export Configuration** to export the configuration settings.
 - Manually backup the `kmdata.tar` file that contains the signing key for the Hardware Secure Module (HSM).

Database

Select the **Database** tab of the **Configuration** page to configure the connection with the database described in [Creating the Certificate Authority database](#).

- [Database URL](#)
- [Database Name](#)
- [Database username](#)
- [Database password](#)
- [Enable SSL mode for the PostgreSQL database](#)
- [CA Certificate\(s\)](#)

Database URL

The URL for connecting with the DBMS.

Mandatory: Yes.

Database Name

The name of the database

Mandatory: Yes.

Database username

The username for connecting with the database.

Mandatory: Yes.

Database password

The password for for connecting with the database.

Mandatory: Yes.

Enable SSL mode for the PostgreSQL database

yes to enable SSL security in the database connection, **no** otherwise

Mandatory: Yes.

CA Certificate(s)


The CA certificates for validating the SSL certificate of the database.

Mandatory: When **Enable SSL mode for the PostgreSQL** database is **yes**.

HSM

Select the **HSM** tab of the **Configuration** page to configure the Hardware Security Module (HSM).

- [HSM](#)
- [Vendor](#)
- [HSM PIN](#)
- [Host to download the nShield kmdata](#)
- [Username to download the nShield kmdata](#)
- [Password to download the nShield kmdata](#)
- [Key application type \(APPNAME\)](#)
- [Key unique identifier](#)

 See [HSM requirements](#) for the supported HSM and [Configuring an nShield HSM](#) for the additional steps required by Entrust nShield HSMs.

HSM

An identifier for the HSM in Certificate Authority.

Mandatory: Yes.

Vendor

The identifier of the HSM manufacturer.

Vendor	Description
none	A built-in software PKCS #11 module (not recommended).
nshield	An Entrust nShield HSM. See HSM requirements for the supported versions.

Mandatory: Yes.

HSM PIN

The PIN for accessing the HSM.

Mandatory: Yes.

Host to download the nShield kmdata

The domain name of the IP address of the host for downloading the kmdata configuration of the HSM.

Mandatory: When the value of **Vendor** is **nShield**.

Username to download the nShield kmdata

The username for logging into the host and downloading the kmdata configuration of the HSM.

Mandatory: When the value of **Vendor** is **nShield**.

Password to download the nShield kmdata

The password for logging into the host and downloading the kmdata configuration of the HSM.

Mandatory: When the value of **Vendor** is **nShield**.

Key application type (APPNAME)

The value of the `APPNAME` parameter in the HSM.

Mandatory: When the value of **Vendor** is **nShield**.

Key unique identifier

The unique identifier of the Certificate Authority key in the HSM.

Mandatory: When the value of **Vendor** is **nShield**.

General

Select the **General** tab of the **Configuration** page to configure the CRL (Certificate Revocation List) generation.

Field	Description	Default
Hostname	The domain name or IP address of the host where the Certificate Authorities solution will publish the issued CRLs.	—
CRL Generation (in days)	The number of days between each CRL issuance.	When omitting this value, the Certificate Authorities solution does not issue CRLs.

Each CA instance will publish a non-partitioned Certificate Revocation List in the following URI.


```
http://{hostname}/crl/{organization}/{caid}/crl.crl
```

Where:

- `{hostname}` is the value of the **Hostname** field.
- `{organization}` is the name of the organization selected when [Creating Certificate Authority instances](#).
- `{caid}` is the CA identifier selected when [Creating Certificate Authority instances](#).


Creating Certificate Authority tenants

Create and configure the tenants that will manage the Certificate Authorities. Each tenant can manage one or several Certificate Authorities.

 CA tenant's user sessions expire after 30 minutes, requiring a re-login.

To create a Certificate Authority tenant

1. Select **Administer > Roles** in the Management Console menu.
2. Create a role with the **Solutions > Manage and Operate Certificate Authorities (CAs)** permission.
3. Select **Administer > Users** in the Management Console menu.
4. Create a user with the following settings.
 - [Name](#)
 - [Password](#)
 - [Roles](#)

 The role will be effective only on users created after the role creation.

Name

Enter the name of the Certificate Authority tenant.

Name length	Supported characters
3-13 characters	Lowercase letters, numbers, dashes ("-"), and underscores ("_").

Email

Enter the email of the Certificate Authority tenant.

Password

Write and confirm a password for the Certificate Authority tenant.

Roles

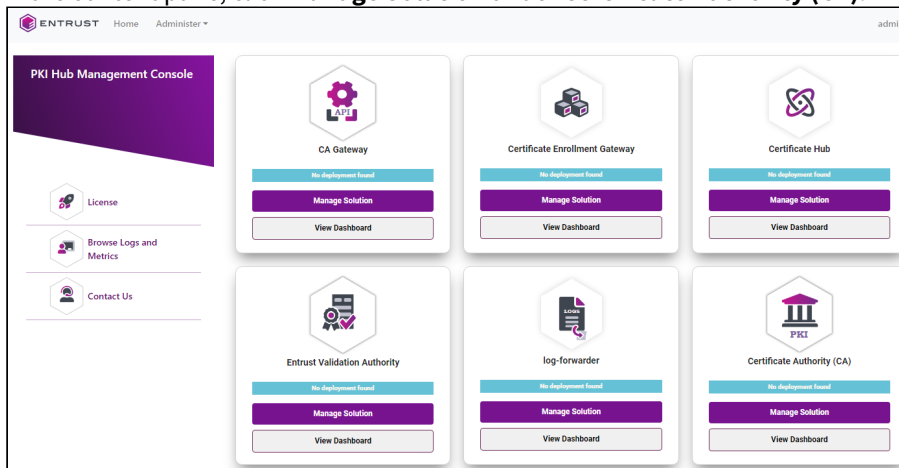
Select the role with the **Manage Certificate Authority** permission previously created.

Creating Certificate Authority instances

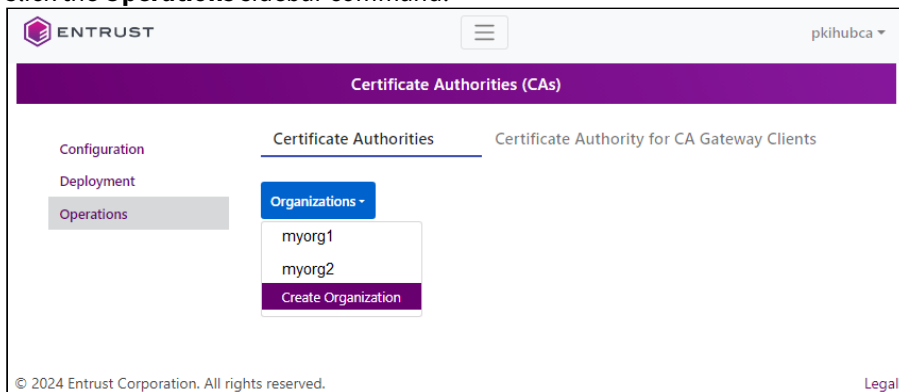
See below for how to create root or subordinate Certificate Authority instances.

To create a Certificate Authority instance


1. Log in to the Management Console as one of the users created in [Creating Certificate Authority tenants](#). This user will be the tenant of the new Certificate Authority.
2. In the content pane, click **Manage Solution** under **Certificate Authority (CA)**.



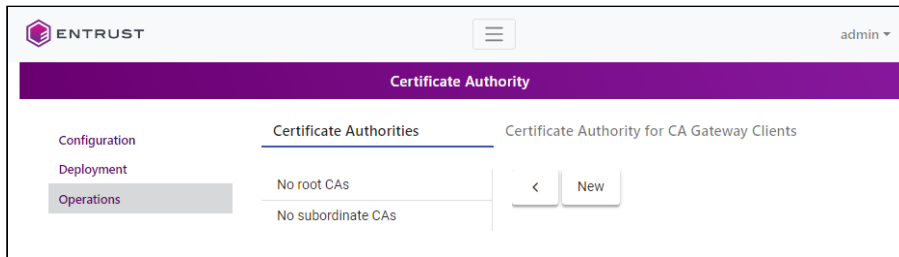
3. Click the **Operations** sidebar command.



4. Click **Organizations** and select the organization to which the new CA will belong. You can also select **Create Organization** and create a new organization.

 Do not use organization names such as 'test' and 'testuser' because they are used for testing operations.

5. Click **New**.



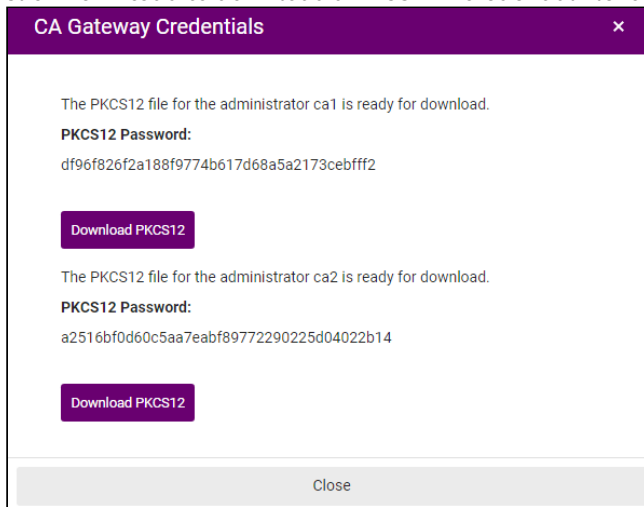
6. Configure the following settings.


- [CA Type](#)
- [CA ID](#)
- [CA Key Type](#)
- [Certificate Profiles](#)
- [Issuer CA ID](#)
- [Expiration Date](#)
- [Attributes](#)
- [Auditors](#)
- [Administrators](#)
- [OCSP Key Type](#)

7. Click **Submit** to create the new Certificate Authority.

8. Copy the password of the client PKCS #12 for consuming the Certificate Authority services.

9. Click **Download** to download a PKCS #12 credential file for each of the selected [Administrators](#).



 Do not lose the PKCS #12 files or passwords, as you cannot obtain them later.

CA Type

The type of Certificate Authority.

Type	Description
Root Certificate Authority	A top-level Certificate Authority.


Type	Description
Issuing Certificate Authority	An intermediate CA that is signed by a Root CA or another Subordinate CA.
External Root Certificate Authority	A Root Certificate Authority that was not created with the Certificate Authorities solution.

Mandatory: Yes.

CA ID

A unique identifier for the new Certificate Authority within its organization.

Identifier length	Supported characters
3-18 characters	Lowercase letters, numbers, dashes ("-"), and underscores ("_").


 Do not reuse the identifier of a Certificate Authority for up to 24 hours after it has been deleted.

Mandatory: Yes.

CA Key Type

The type of key the new Certificate Authority will use to sign certificates.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Mandatory: Yes.

Certificate Profiles

The profiles the Certificate Authority will support for issuing certificates. See the [Certificate profiles reference](#) for a description of each profile.

Mandatory: Select at least one profile.

Issuer CA ID

The identifier of the root Certificate Authority.

Mandatory: When **CA Type** is **Issuing Certificate Authority**.

Expiration Date

The expiration date for the certificate signing certificate of the Certificate Authority.

Mandatory: No. This value defaults to the following dates.

CA Type	Default expiration date
Root Certificate Authority	20 years after the certificate is issued
Issuing Certificate Authority	10 years after the certificate is issued


Attributes

The value of each attribute in the Distinguished Name (DN) of the Certificate Authority certificate.

Mandatory: Set at least the **CN** attribute of the Distinguished Name.

Auditors


The names of the users that will have auditing permission on the Certificate Authority.

 On Certificate Authority creation, the Certificate Authorities solution will automatically generate client PKCS #12 files to authenticate these users.

Mandatory: No. When omitting this value, the Certificate Authority will not have users with auditor permission.

Administrators

The names of the users who will have administrative permissions on the new Certificate Authority. A CA can have multiple administrators and an administrator can be assigned to different CAs.

 On Certificate Authority creation, the Certificate Authorities solution will automatically generate client PKCS #12 files to authenticate these users.

Mandatory: Add the name of at least one administrator.

OCSF Key Type

The type of key to sign OCSF responses at the following endpoint.

```
http://{pkihub}/ocsp/{organization}/{caid}
```

- `{pkihub}` is the domain name or IP address of the machine running PKI Hub.
- `{organization}` is the identifier of the CA organization.
- `{caid}` is the value of the **CA ID** field.

Mandatory: When **CA Type** is **Issuing Certificate Authority**.

Issuing certificates with Certificate Authority instances

To issue certificates with a Certificate Authority instance, you must send requests to the embedded CA Gateway. See below for the supported modes.

- [Issuing certificates with a REST client](#)
- [Issuing certificates with Certificate Hub](#)

Issuing certificates with a REST client

See below for issuing certificates with the REST API exposed by the embedded CA Gateway of the Certificate Authorities solution.

To issue certificates with a REST client

1. Install a REST client.
2. As the client credential, select the administrator PKCS #12 automatically generated when [Creating Certificate Authority instances](#).
3. Import the Swagger specification, which is available at the following endpoint.

```
https://{pkihub}:7443/cagw
```

Where `{pkihub}` is the hostname or IP address of the machine hosting PKI Hub.

4. Send a REST requests to the issuing CA endpoint – for example:

```
curl --request POST --header "Accept: application/json" --header "Content-Type: application/json" -d @enrollments.json --cert-type P12 --cert $P12:$PWD https://{PKIHUB}:7443/cagw/v1/certificate-authorities/$ORG_ID~$CA_ID/enrollments | jq .
```

Where :


- `$P12` is the path of the PKCS#12 file.
- `$PWD` is the password of the PKCS 12 file.
- `$PKIHUB` is the hostname or IP address of the machine hosting PKI Hub.
- `$ORG_ID` is the identifier of the organization to which the CA belongs.
- `$CA_ID` is the identifier of the issuing CA.

Issuing certificates with Certificate Hub

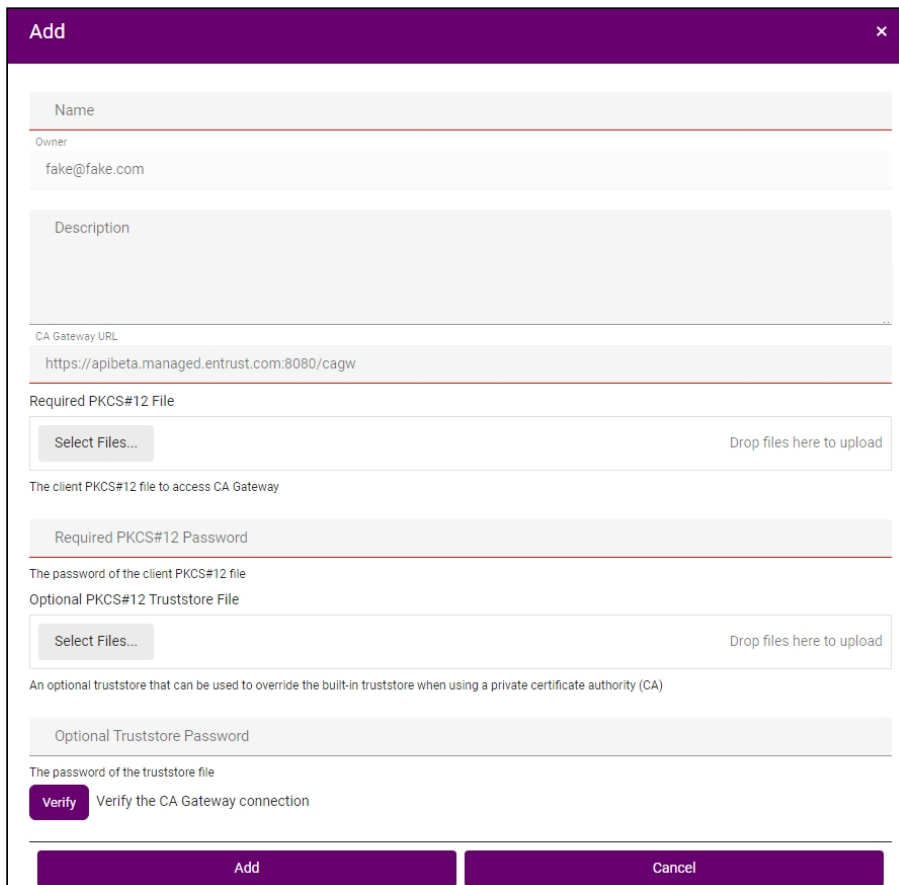
See below for instructions on adding the embedded CA Gateway service, provided by the Certificate Authorities solution, to Certificate Hub. Once added, Certificate Hub will support requesting certificates from all CA instances.

To add a CA Gateway instance to Certificate Hub

1. Install a Certificate Hub instance as explained in [Starting up Certificate Hub](#).
2. Open a web browser using the URL set when deploying Certificate Hub.
3. Authenticate using the method and credentials provided by your Certificate Hub global administrator.

 Once logged in, click your username at the top-right corner and then **Help** to display the Certificate Hub user guide.

4. In the menu bar, select **Control > CA Gateways**.
5. On the **CA Gateways** page, click **Add**.
6. Configure the following values in the **Add** dialog.
 - [Name](#)
 - [CA Gateway URL](#)
 - [Required PKCS#12 File](#)
 - [Required PKCS#12 Password](#)
 - [Optional PKCS#12 Truststore File](#)



Add [Close]

Name

Owner
fake@fake.com

Description

CA Gateway URL
https://apibeta.managed.entrust.com:8080/cagw

Required PKCS#12 File
Select Files... Drop files here to upload

The client PKCS#12 file to access CA Gateway

Required PKCS#12 Password

The password of the client PKCS#12 file

Optional PKCS#12 Truststore File
Select Files... Drop files here to upload

An optional truststore that can be used to override the built-in truststore when using a private certificate authority (CA)

Optional Truststore Password

The password of the truststore file

Verify Verify the CA Gateway connection

Add **Cancel**

7. Click **Add** to confirm the CA Gateway creation.

Name

Write a name for the CA Gateway instance in Certificate Hub.

CA Gateway URL

Enter the following URL:

```
https://<hostname>:7443/cagw
```

Where `<hostname>` is the domain name or IP address of the machine hosting Entrust PKI Hub.

Required PKCS#12 File

Click **Select Files** and import the administrator PKCS #12 that is automatically generated when [Creating Certificate Authority instances](#).

Required PKCS#12 Password

Enter the PKCS#12 password that is automatically displayed when [Creating Certificate Authority instances](#).

Optional PKCS#12 Truststore File

Click **Select Files** and import PKCS#12 containing the certification chain of the TLS certificate described in [Replacing the default TLS certificate](#).



To put the certification chain on a PKCS #12 file, you can use free tools such as <https://keystore-explorer.org>

Changing the HSM vendor

As explained in [Configuring and deploying Certificate Authorities](#), the **Vendor** field of the **HSM** configuration page allows selecting the following Hardware security modules.

Vendor	Description
none	A built-in software PKCS #11 module (not recommended).
nshield	An Entrust nShield HSM. See HSM requirements for the supported versions.

On test environments, you can change the **Vendor** parameter value of an already deployed Certificate Authorities solution.

- [Changing vendor from none to nShield](#)
- [Changing vendor from nShield to none](#)

Changing vendor from none to nShield

See below for changing the value of the **Vendor** parameter from **none** to **nShield**.

To change the vendor from none to nShield

1. Create a new database, as explained in [Creating the Certificate Authorities database](#), or recreate the public schema of the database.
2. Run the following command.

```
sudo kubectl delete namespace pkihub-v202410180954
```

3. Set the **Vendor** field of the [HSM](#) configuration page to **nShield**.
4. Save the configuration.
5. Redeploy the solution.

Changing vendor from nShield to none

See below for changing the value of the **Vendor** parameter from **nShield** to **none**.

i Use `pkihub` as solution identifier when running the `clusterctl solution config export` and `clusterctl solution config import` commands.

To change the vendor from nShield to none

1. Create a new database, as explained in [Creating the Certificate Authorities database](#), or recreate the public schema of the database.
2. Run the following command.

```
sudo kubectl delete namespace pkihub-v202410180954
```

3. Set the **Vendor** field of the [HSM](#) configuration page to **none**.
4. Save the configuration.
5. Export the configuration files with the `clusterctl solution config export` command.
6. Delete the `config` folder of the HSM installation.
7. Delete the `kmdata.tar` file of the HSM installation.
8. Import the configuration files with the `clusterctl solution config import` command.
9. Redeploy the solution.

Starting up CA Gateway

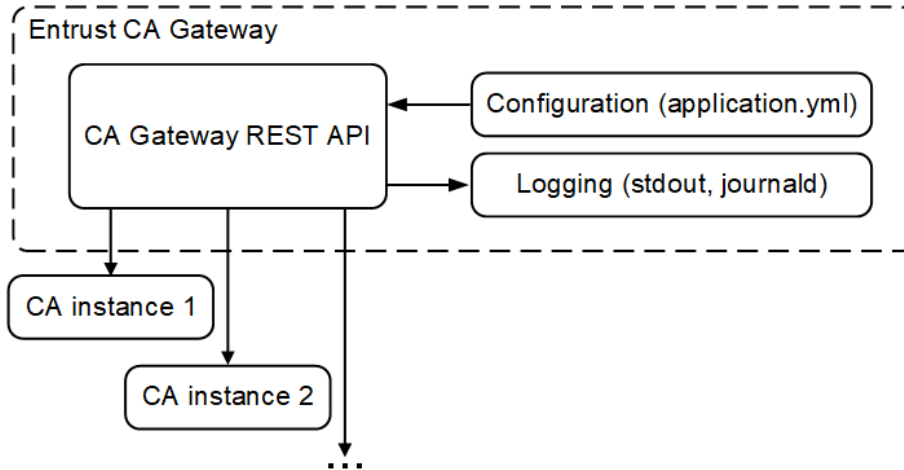
See below for starting up the CA Gateway solution.

- [CA Gateway overview](#)
- [Preparing the CA Gateway deployment](#)
- [Integrating Certificate Authorities with CA Gateway](#)
- [Configuring and deploying CA Gateway](#)
- [Issuing public trust certificates with CA Gateway](#)
- [Administering CA Gateway](#)
- [CA Gateway health endpoints](#)
- [Other CA Gateway endpoints](#)
- [CA Capabilities reference](#)

See [Browsing logs with Grafana](#) for how to browse Certificate Authority logs.


CA Gateway overview

CA Gateway is a lightweight, container-based module implementing a CA-agnostic Certificate Lifecycle and Policy Management API. Using CA Gateway, your applications can provide certificate issuance, renewal, and revocation actions across different Certification Authorities (CAs). CA Gateway provides policy retrieval capabilities so applications can customize API and user-facing dialogs to ensure that certificate actions conform to organizational policies.



See below for a description of each component.

- [Client](#)
- [Integrator](#)
- [Tenant](#)
- [Managed CA](#)

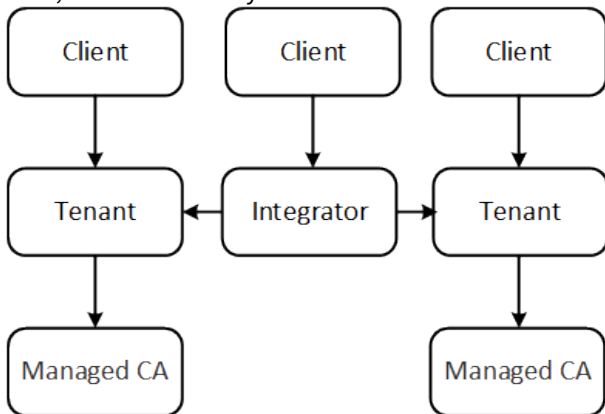
 CA-specific plugins communicate with the underlying CAs through mutually authenticated TLS.

Client

Each client is an authorized end entity of the CA Gateway API and is mapped either to a tenant or an integrator.

- Clients mapped to an integrator can access many Managed CAs.
- Clients mapped to a tenant can access only that tenant's Managed CA.

Thus, each CA Gateway client can access one or several CAs.



The CA Gateway API is regularly updated to add functionalities. Therefore, client applications:

- Should tolerate and ignore new fields.
- Should be recompiled against the new data model of each CA Gateway release.

Integrator

Each integrator is an access controller for one or more tenants.

Tenant

Each tenant is an access controller for a Managed CA. Thus, each tenant:

- Has only one integrator.
- Controls access to a different CA.

Managed CA


Each managed CA is a collection of information that CA Gateway uses to connect to a CA. For example:

- Microsoft Active Directory Certificate Services.
- AWS Certificate Manager Private Certificate Authority.
- Entrust CA (Entrust Authority Security Manager, Entrust mPKI).

Preparing the CA Gateway deployment

Prepare the CA Gateway deployment as explained in the following sections.

- [Verifying port access for CA Gateway](#)
- [Obtaining the CA Gateway server certificate](#)

 As explained in [Starting up PKI Hub](#), do not perform system operations other than those described in this guide. Specifically, the user and group identifiers 1339 are reserved for CA Gateway images, so the host server should not use them.

Verifying port access for CA Gateway

In addition to the ports listed in [Required open ports](#), ensure no network restriction blocks access to port 8444.

i CA Gateway deployment automatically opens this port in the firewall of the machines hosting Entrust PKI Hub.

Obtaining the CA Gateway server certificate

CA Gateway requires a digital certificate for securing communications between the CA Gateway and authorized clients. See below to generate this certificate for a production environment.

- [Generating the server key pair](#)
- [Obtaining the key pair CSR](#)
- [Obtaining the server certificate](#)
- [Importing the server certificate into the keystore](#)
- [Importing CA certificates into a truststore](#)
- [Reusing the PKI Hub TLS certificate](#)

! The certificate must contain the server's fully qualified domain name (FQDN) as a DNS type Subject Alternative Name (subjectAltName) extension.

Generating the server key pair

To generate the server key pair, run the following command.

```
keytool -genkeypair -alias <ALIAS> -dname <DN> -keyalg <KEYALG> -keysize <KEYSIZE>
-sialg sha256WithRSA -ext san=dns:<DNS> -keystore <KEystore> [-keypass <KEYPASS>] [-
storepass <STOREPASS>]
```

See the following table for a description of each flag.

Flag	Value
-alias	An alias for the key pair.
-dname	The DN for the key pair (and later, the certificate). Use the DN format expected by the CA that will issue the certificate.
-keyalg	The algorithm for the key pair (for example, RSA).
-keysize	The Key size. Select a secure key size (for example, 2048).
-ext	The DNS-type value of the Subject Alternative Name (subjectAltName) extension.
-keystore	The full path of the keystore file. If the keystore does not exist, the keytool utility will create it.

Flag	Value
-keypass	The password of the private key. When you omit this option, the tool prompts for a password.
-storepass	The password for the keystore. When you omit this option, the tool prompts for a password.

Obtaining the key pair CSR

Create a Certificate Signing Request (CSR) by entering the following command:

```
keytool -certreq -alias <ALIAS> -file <FILE> -storetype pkcs12 -keystore <KEYSTORE>
[-storepass <STOREPASS>]
```

For example:

```
> keytool -genkeypair -alias example_alias -dname "cn=CA Gateway,ou=CA
Entry,o=Example,c=US" -keyalg RSA -keysize 2048 -sigalg sha256WithRSA -ext
san=dns:domain.example.com -keystore /CAGW/config/keystore.ks
> keytool -certreq -alias example_alias -file /tmp/cagw/cagw_csr.txt -keystore /CAGW/
config/keystore.ks
```

See the following table for a description of each option.

Option	Value
-alias	The alias previously specified when Generating the server's key pair.
-file	The full path of the CSR file.
-keystore	The full path of the keystore file.
-storepass	The password of the keystore. When you omit this option, the tool prompts for a password.

Obtaining the server certificate

Issue the certificate with either:

- Your Security Manager CA.
- A trusted certificate provider such as the Entrust Certificate Services at store.entrust.com.

Importing the server certificate into the keystore

Import the certificate into the keystore:

```
keytool -importcert -alias <ALIAS> -file <FILE> -keystore <KEYSTORE>
```

For example:

```
keytool -importcert -alias example_alias -file /tmp/cagw/cagw_cert.p7b-keystore /
home/myuser/cagw/config/keystore.ks
```

See the following table for a description of each option.

Option	Value
-alias	The alias previously specified when Generating the server's key pair.
-file	The full path of the PKCS #7 file containing the certificate and the certificate chain.
-keystore	The full path of the keystore file.

Importing CA certificates into a truststore

For each managed Certificate Authority, CA Gateway requires the following certificates.

CA type	Required certificates
Root	The self-signed root CA certificate.
Subordinate	The complete CA certificate chain, from the subordinate CA certificate up to the root CA certificate.

Import these certificates in either:

- The Truststore used when Importing the server certificate into the keystore.
- A new Truststore.

To import a CA certificate into a truststore using the Java `keytool` utility, run the following command.

```
keytool -importcert -trustcacerts -alias <ALIAS> -file <FILE> -keystore <KEYSTORE> [-
storepass <STOREPASS>]
```

For example:

```
keytool -import -trustcacerts -alias managed_ca1 -file /tmp/cagw/managed_ca1.cer
-keystore /home/myuser/cagw/config/keystore.ks
```

See the following table for a description of each parameter.

Option	Value
-alias	The alias of the CA certificate.
-file	The full path of the CA certificate file.
-keystore	The full path of the Java keystore file. If not present, the keystore is created.
-storepass	The password of the Java keystore. When you omit this option, the tool prompts for a password.

Reusing the PKI Hub TLS certificate

As TLS certificate for CA Gateway, you can use the same TLS server certificate described in [Replacing the default TLS certificate](#)

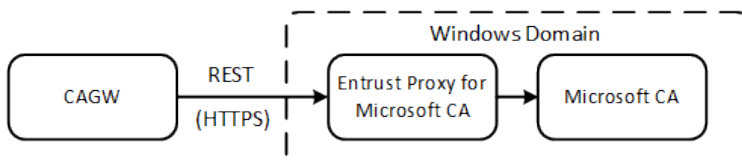
Integrating Certificate Authorities with CA Gateway

The below sections explain how to integrate CA Gateway with Certificate Authorities of different providers.

- [Integrating a Microsoft CA](#)
- [Integrating an ECS CA](#)
- [Integrating a Security Manager CA](#)

Integrating a Microsoft CA

As illustrated by the following figure, CA Gateway manages each Microsoft CA instance through an Entrust Proxy for Microsoft CA.



In this architecture, CA Gateway is a client of Microsoft CA. See in the following sections how to configure the Entrust Proxy for Microsoft CA and CA Gateway to manage Microsoft CAs.

- [Installing the Entrust Proxy for Microsoft CA](#)
- [Issuing the SSL certificates](#)
- [Creating a client authentication template for Microsoft CA](#)
- [Generating a client keystore for CA Gateway](#)
- [Generating a truststore for CA Gateway](#)
- [Generating the server keystore of the Entrust Proxy for Microsoft CA](#)
- [Configuring the logs of the Entrust Proxy for Microsoft CA](#)
- [Running the Entrust Proxy for Microsoft CA](#)
- [Creating the CA enrollment agents](#)
- [Creating the RA recovery agents](#)
- [Creating the RA enrollment agents](#)
- [Enabling supply in the request](#)

- [Configuring Request Handling in the Microsoft CA](#)
- [Enabling SAN attributes in the enrollment request](#)

i Only Microsoft Enterprise CA is supported; standalone CA is not supported.

Installing the Entrust Proxy for Microsoft CA

Install the Entrust Proxy for Microsoft CA, as explained in the following sections.

- [System requirements of the Entrust Proxy for Microsoft CA](#)
- [Configuring the Windows domain account](#)
- [Downloading Entrust Proxy for Microsoft CA](#)
- [Running the Entrust Proxy for Microsoft CA installer](#)

System requirements of the Entrust Proxy for Microsoft CA

To install the Entrust Proxy for Microsoft CA, you need a machine with Windows Server 2016 (x64) or above and one of the following LTS (Long Term Support) Java distributions.

- Oracle Java x86_64 version 17
- OpenJDK 17
- AdoptOpenJDK 17

To check the Java version and architecture details, run:

```
java -XshowSettings:properties -version
```

Configuring the Windows domain account

Configure the Windows login account of the Entrust Proxy for Microsoft CA.

If the Entrust Proxy for Microsoft CA, the Domain Controller, and the Microsoft CA share the same server, you can select the following user and startup type combinations.

User	Service startup type
A local service account	Automatic or Automatic (Delayed Start)
A user of the Enterprise Admin group	Automatic (Delayed Start)

If Entrust Proxy for Microsoft CA, the Domain Controller, and the Microsoft CA are on different servers, you can only select the following combination.

User	Service startup type
A user of the Enterprise Admin group	Automatic or Automatic (Delayed Start)

In either case, enable only the following user permissions.

- Issue and Manage Certificates
- Request Certificates

Downloading Entrust Proxy for Microsoft CA

To download the Entrust Proxy for Microsoft CA:

1. Log in trustedcare.entrust.com
2. Go to **PRODUCTS > Authority**
3. Select your CA Gateway version.
4. Click the download link of the Entrust Proxy for Microsoft CA.
5. Unzip the compressed file's contents to your selected installation directory on the Windows machine. For example, in `c:\mscaproxy`

Running the Entrust Proxy for Microsoft CA installer

Run the following command as an administrator to register the Entrust Proxy for Microsoft CA as a Windows service.

```
MSCAProxy.exe install /p
```

When prompted, type the domain user's username in one of the following formats:

- UPN (User Principal Name)
- `<domainName>\<sAMAccountName>`

Type the password of the domain user and type `y` for allowing the log-on as a service. The installer does not wait for you to press the **Enter** key.

Issuing the SSL certificates

CA Gateway and the Entrust Proxy for Microsoft CA communicate with HTTP over SSL and mutual authentication. Thus, two SSL certificates are required:

- A server SSL certificate for the Entrust Proxy for Microsoft CA.
- A client authentication certificate for CA Gateway.

You can obtain both SSL certificates from any CA. Those steps are outside the scope of this document.

Creating a client authentication template for Microsoft CA

Create an authentication template for enabling client authentication in Microsoft CA.

To create a client authentication template for Microsoft CA

1. Go to **Certificate Authority**.
2. Right-click **Certificate Templates** and select **Manage**.
3. Right-click the **User** template and select **Duplicate Template**.
4. In the **General** tab of the **Properties of New Template** dialog, set **Template display name** to **Client Authentication**.
5. In the **Subject Name** tab, enable **Supply in the request**.
6. In the **Extensions** tab, edit **Application Policies** to remove **Encrypting File System** and **Secure Email**.
7. Go to **Certificate Authority**.
8. Right-click **Certificate Templates** and select **New >Certificate Template to Issue**.
9. Select **Client Authentication** from the list.

Generating a client keystore for CA Gateway


Generate a `mscaproxyclient.jks` keystore containing:

- The private key of CA Gateway for client authentication.

- The key's certificate.
- The certificate's chain.

See below the required steps.

- [Generating and certifying the key pair](#)
- [Importing the keys and the certificate](#)
- [Deleting temporary files](#)

 The following instructions create a Java KeyStore (JKS) with the Java `keytool` command line utility. Consider using a more secure PKCS#12 type instead.


Generating and certifying the key pair

In a temporary directory under the Microsoft Proxy Server, run the following commands to generate and certify a key pair.

```
keytool -genkey -noprompt -alias mscaproxyclient -dname "cn=mscaproxy client" -keyalg
RSA -keysize 2048 -keystore mscaproxyclient.jks -storepass <STOREPASS> -keypass
<KEYPASS>
```

```
keytool -certreq -alias mscaproxyclient -file mscaproxyclient.csr -keystore
mscaproxyclient.jks -storepass <STOREPASS>
```

```
certreq.exe -f -attrib "CertificateTemplate:ClientAuthentication" -config
"<HOST>\<CA>" mscaproxyclient.csr CertChainFileOut mscaproxyclient.p7b
```

 Depending on the Microsoft CA setup, you may need to manually approve the request and retrieve the certificate.

See the following table for a description of the main parameters.

Option	Value
-attrib	The name of the template authentication template for Microsoft CA you previously created.
-config	The keystore configuration in "<HOST>\<CA>" syntax. Where <HOST> is the Microsoft CA's hostname, and <CA> is the CA name defined when configuring Microsoft CA in CA Gateway.
-dname	A valid certificate distinguished name.
-keypass	The password of the private key

Option	Value
-keystore	The name of the keystore file. Copy this file into the CA Gateway's server
-storepass	The keystore password.

Importing the keys and the certificate

Import the keys and the certificate into the keystore.

```
keytool -import -noprompt -alias mscaproxyclient -file mscaproxyclient.p7b -keystore mscaproxyclient.jks -storepass <STOREPASS>
```

Deleting temporary files

Delete the temporary files.

```
del CertChainFileOut
del CertChainFileOut.rsp
del mscaproxyclient.csr
del mscaproxyclient.p7b
```

Generating a truststore for CA Gateway

You need a `truststore.jks` truststore containing the CA chain of the Entrust Proxy for Microsoft CA's server key.

⚠ The following instructions create a Java KeyStore (JKS) with the Java `keytool` command line utility. Consider using a more secure PKCS#12 type instead.

To generate a truststore for CA Gateway

1. Create an SSL directory under the Entrust Proxy for Microsoft CA installation. For example:

```
c:\mscaproxy\ssl
```

2. In this directory, run the following command to include the certificate of the root CA and all the intermediate CAs.

```
keytool -import -noprompt -alias <CA_ALIAS> -file <CA_ALIAS>.cer -keystore truststore.jks -storepass <STOREPASS>
```

3. Copy the new `truststore.jks` truststore in the CA Gateway server.


Generating the server keystore of the Entrust Proxy for Microsoft CA

You need a keystore containing:

- The SSL authentication certificate of the Entrust Proxy for Microsoft CA.
- The private key of the certificate.
- The validation chain of the certificate.

See below the required steps.

- [Generating the keystore](#)
- [Setting the Subject Name](#)
- [Adding the keystore password to the configuration](#)
- [Adding the truststore password to the configuration](#)
- [Restarting CA Gateway](#)


 The following instructions create a Java KeyStore (JKS) with the Java `keytool` command line utility. Consider using a more secure PKCS#12 type instead.

Generating the keystore

Go to the SSL directory containing the `truststore.jks` file previously generated. For example:

```
c:\mscaproxy\ssl
```

Run the following commands to generate the key.

 The below commands use the default Web Server certificate template. If you need to customize any settings of the Web Server certificate template, use a copy of it.

```
keytool -genkey -noprompt -alias mscaproxy -dname "cn=MS CA proxy server FQDN"  
-keyalg RSA -keysize 2048 -keystore mscaproxy.jks -storepass <STOREPASS> -keypass  
<KEYPASS>
```

```
keytool -certreq -alias mscaproxy -ext SAN=dns:MS CA proxy server FQDN -file  
mscaproxy.csr -keystore mscaproxy.jks -storepass <STOREPASS>
```

```
certreq.exe -f -attrib "CertificateTemplate:WebServer" -config "MS CA host name\CA  
name" mscaproxy.csr CertChainFileOut mscaproxy.p7b
```

```
keytool -import -noprompt -alias mscaproxy -file mscaproxy.p7b -keystore  
mscaproxy.jks -storepass <STOREPASS>
```

```
del CertChainFileOut
del CertChainFileOut.rsp
del mscaproxy.csr
del mscaproxy.p7b
```

Where:

- "MS CA proxy server FQDN" is the fully qualified domain name of your Entrust Proxy for Microsoft CA's server.
- <STOREPASS> is the password of the keystore.
- <KEYPASS> is the password of the private key.

Setting the Subject Name

Edit the `application.yml` file of the Entrust Proxy for Microsoft CA installation folder.

```
config\application.yml
```

Uncomment all lines (by removing #) and assign to `subject-dn` the distinguished name set with `-dname` when generating the client keystore. For example:

```
subject-dn: "cn=mscaproxy client"
```

Adding the keystore password to the configuration

Edit the following file.

```
MS CA Proxy Installation\config\key-store-password.scrt
```

Set the following parameter:

```
decrypted=<STOREPASS>
```

Where <STOREPASS> is the password of the keystore described in [Generating the keystore](#).

Adding the truststore password to the configuration

Edit the following file:

```
MS CA Proxy Installation\config\trust-store-password.scrt
```

Set the following parameter.

```
decrypted=<STOREPASS>
```

Where <STOREPASS> is the password of the keystore described in [Generating the keystore](#).

Restarting CA Gateway

If the Entrust Proxy for Microsoft CA is running, execute the following command as an administrator to restart it.

```
MSCAProxy.exe restart
```

Configuring the logs of the Entrust Proxy for Microsoft CA

Configure the Entrust Proxy for Microsoft CA logs as follows.

- [Setting the log level of the Entrust Proxy for Microsoft CA](#)
- [Selecting the folder for the Entrust Proxy for Microsoft CA logs](#)

Setting the log level of the Entrust Proxy for Microsoft CA

To enable log recording and set the log level, edit the following file of the Entrust Proxy for Microsoft CA installation folder.

```
config\application.yml
```

Assign to the `com.entrust.mscaproxy` parameter one of the supported log levels. In increasing severity.

- TRACE
- DEBUG
- INFO
- WARN
- ERROR
- FATAL
- OFF

For example:

```
logging:  
  level:  
    com.entrust.mscaproxy: INFO
```

Selecting the folder for the Entrust Proxy for Microsoft CA logs

By default, the Entrust Proxy for Microsoft CA stores the logs in the `logs` subfolder of the installation folder. See below how to select a different folder.

To select the log folder

1. Edit the `MSCAProxy.xml` file located in the installation folder.
2. Set the desired location in the `logpath` parameter.
3. Restart the system to make changes effective.

Running the Entrust Proxy for Microsoft CA

Administrators can run and manage the Entrust Proxy for Microsoft CA with the following commands.

- `MSCAProxy.exe start`

- `MSCAProxy.exe stop`
- `MSCAProxy.exe restart`

Once started, you can check the correct execution of the Entrust Proxy for Microsoft CA using a Chrome browser.

To check the execution of the Entrust Proxy for Microsoft CA

1. Run the following command to generate a PKCS#12 from the `mscaproxyclient.jks` keystore.

```
keytool -importkeystore -srckeystore mscaproxyclient.jks -destkeystore  
mscaproxyclient.p12 -srcstoretype JKS -srcstorepass <SRCSTOREPASS>  
-deststoretype PKCS12 -deststorepass <DESTSTOREPASS>
```

2. Import the generated `mscaproxyclient.p12` file into Chrome.
3. Go to:


```
https://<proxyserver>:8443/MSCAProxy/rest/status/ping
```

4. Check the server response. The "MS CA proxy is running" message indicates a correct operation.

Creating the CA enrollment agents

You must create a CA Enrollment Agent (EA) before creating the RA recovery agents the RA enrollment agents.

- [Publishing the enrollment template](#)
- [Creating an enrollment certificate for the CA Administrator](#)

 A CA enrollment agent is self-enrolled and internal to the CA, while a RA enrollment agent is co-located with CA Gateway.

Publishing the enrollment template

If not already published, publish the enrollment agent template as explained in this section.

To publish the enrollment agent template

1. In the Microsoft CA server machine, run MMC.
2. Under the certificate authority name, right-click **Certificate Templates**.
3. Select **New > Certificate Template to issue**.
4. Select **Enrollment Agent**.

Creating an enrollment certificate for the CA Administrator

Create an enrollment certificate for the CA administrator user of the Microsoft CA server.

 Do not export the CA administrator's enrollment key.

To create an enrollment certificate for the administrator

1. In the Microsoft CA server machine, run MMC.
2. Under the **Personal** node, right-click **Certificates** and select **All Tasks > Request New Certificate**.
3. Follow the wizard instructions. When prompted, select the **Enrollment Agent** template.

Creating the RA recovery agents

If you want to store and recover keys generated by the Microsoft CA, create one or more recovery agents as explained below.

To create a recovery agent

1. In the Microsoft CA server machine, run MMC.
2. Under the Certificate Authority node, right-click **Certificate Template**, and select **Manage**.
3. Right-click **Key Recovery Agent** and select **Duplicate Template**.
4. Configure the following settings in each tab of the **Properties of the New Template** dialog.
 - [General](#)
 - [Request Handling](#)
 - [Issuance Requirements](#)
 - [Security](#)
5. Under the Certificate Authority node, right-click **Certificate Template** and select **New > Certificate Template to issue**.
6. Select the newly created template.
7. Create a user in Active Directory.
8. Under the **Personal** node, right-click **Certificates** and select **Tasks > Advanced Operations > Enroll On Behalf Of**
9. Follow the wizard instructions. When prompted, select the newly created user.
10. Right-click the issued certificate and select **Export**.
11. Follow the wizard instructions. In the **Export Private Key** dialog, select **Yes, export the private key**.

General

Click this tab and write a name for the new template in the **Template display name** field.

Request Handling

Click this tab and check the **Allow private key to be exported** box.

Issuance Requirements

Click this tab and set the following values.

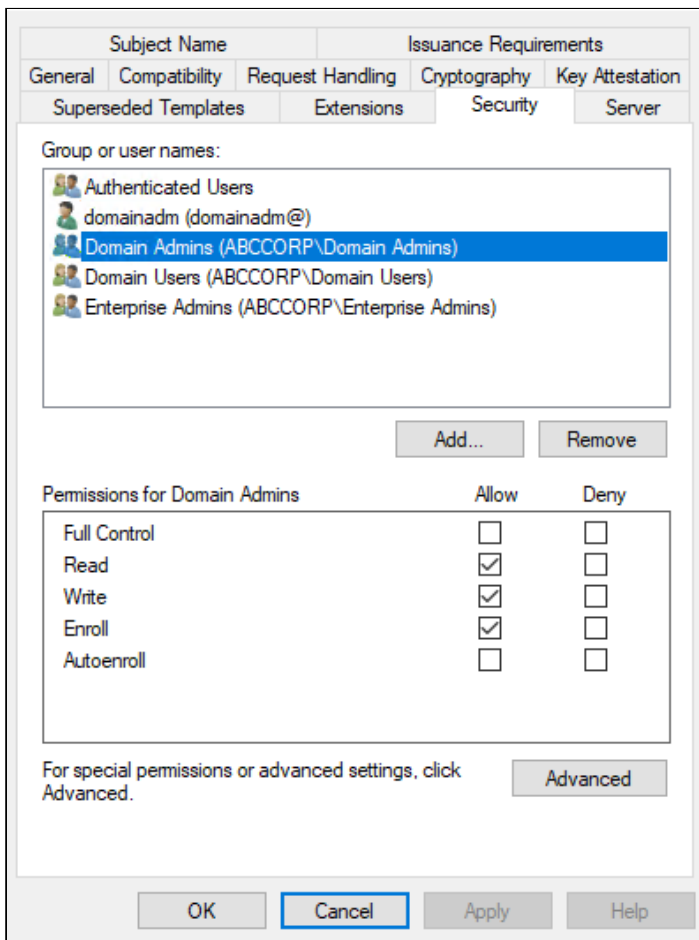
Parameter	Value
CA certificate manager approval	Disable this option
This number of authorized signatures	1
Policy type required in signature	Application policy
Application Policy	Certificate Request Agent

Security

Click this tab and assign the following permissions to the **Domain Admins** user group.

Permissions for Domain Admins	Allow	Deny
Full Control	✘	
Read	✔	
Write	✔	
Enroll	✔	
Autoenroll	✘	

For example:



Creating the RA enrollment agents

To publish the issued certificates in Active Directory, you need one or more RA (Registration Authority) enrollment agents. See below for the supported credential generation modes.

- [Creating RA enrollment agent credentials in a keystore file](#)

- [Creating RA enrollment agent credentials in a PKCS#11 HSM](#)

Creating RA enrollment agent credentials in a keystore file

You can create the RA enrollment agent credentials in the following file formats.

- PKCS#12 (Personal Information Exchange Syntax Standard).
- JKS (Java KeyStore).
- JCEKS (Java Cryptography Extension KeyStore).
- PFX (Personal Information Exchange).

See the example below for how to create them in PKCS#12.

To create RA enrollment agent credentials in PKCS#12

1. In the Microsoft CA server machine, run MMC.
2. Under the Certificate Authority node, right-click **Certificate Template**, and select **Manage**.
3. Right-click **Enrollment Agent** and select **Duplicate Template**.
4. Configure the following settings in each tab of the **Properties of the New Template** dialog.
 - [General](#)
 - [Request Handling](#)
 - [Issuance Requirements](#)
 - [Security](#)
5. Under the Certificate Authority node, right-click **Certificate Template** and select **New >Certificate Template to issue**.
6. Select the newly created template.
7. Create a user in Active Directory.
8. Under the **Personal** node, right-click **Certificates** and select **Tasks > Advanced Operations > Enroll On Behalf Of**.
9. Follow the wizard instructions. When prompted, select the newly created user.
10. Right-click the issued certificate and select **Export**.
11. Follow the wizard instructions. In the **Export Private Key** dialog, select **Yes, export the private key**.

General

Click this tab and write a name for the new template in the **Template display name** field.

Request Handling

Click this tab and check the **Allow private key to be exported** box.

Issuance Requirements

Click this tab and set the following values.

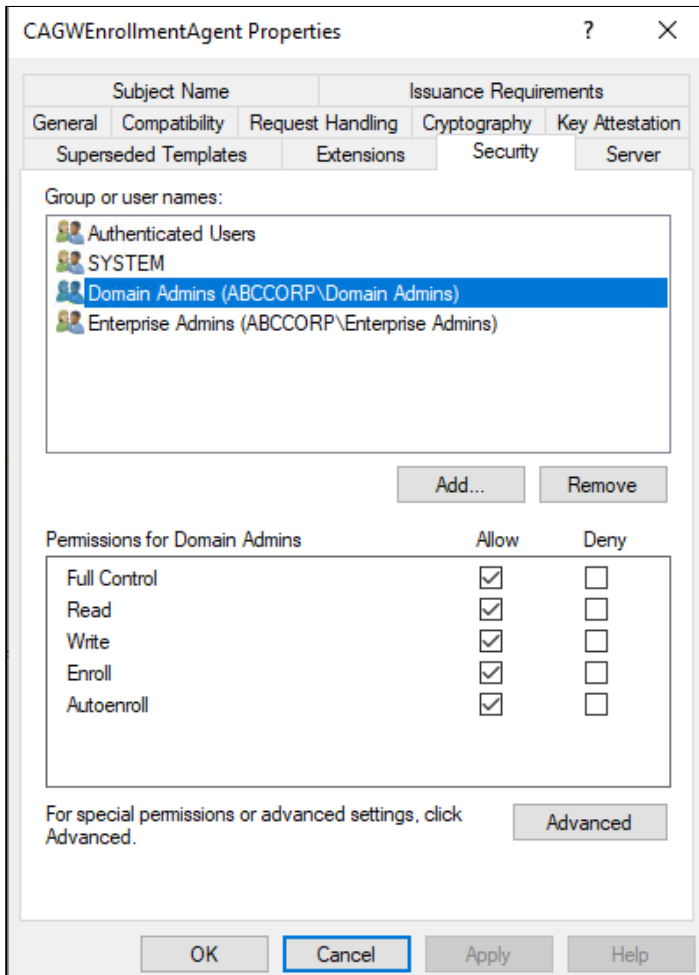
Parameter	Value
This number of authorized signatures	1
Policy type required in signature	Application policy
Application Policy	Certificate Request Agent

Security

Click this tab and assign the following permissions to the **Domain Admins** user group.

Permissions for Domain Admins	Allow	Deny
Full Control	✓	
Read	✓	
Write	✓	
Enroll	✓	
Autoenroll	✓	

For example:



Creating RA enrollment agent credentials in a PKCS#11 HSM

When creating enrollment agents for the Microsoft CA, you can generate keys in a PKCS#11 HSM along with a CSR. When processing this CSR, the Microsoft CA issues a certificate chain for the RA Enrollment Agent that you can import into the HSM to pair with the private key.

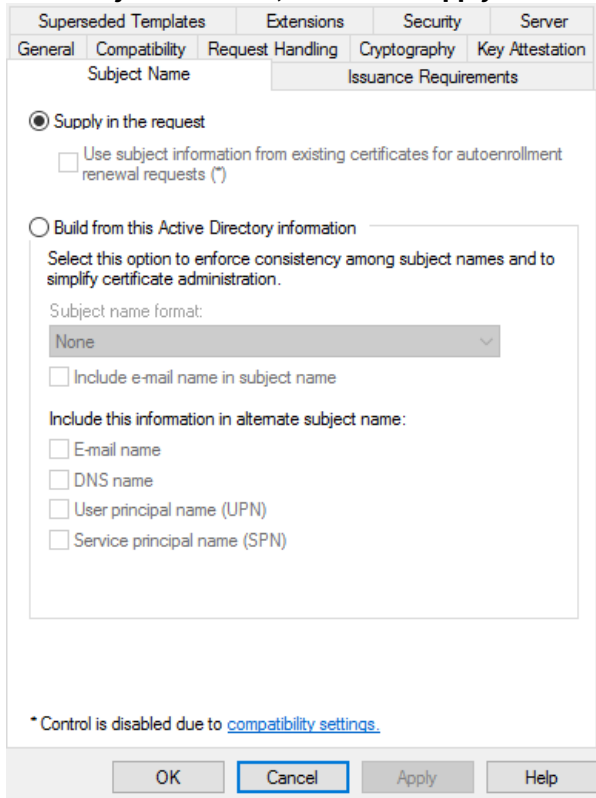
 See the integration guides of the supported HSM for the required operations.

Enabling supply in the request

In all the managed Microsoft CA templates for issuing entity certificates, make sure that the Subject Name is supplied by the certificate request.

To enable supply in the request in a template

1. Go to **Certificate Authority**.
2. **Right-click Certificate Templates and select Manage.**
3. Right-click the template and select **Properties**.
4. In the **Subject Name** tab, enable the **Supply in the request** radio button.



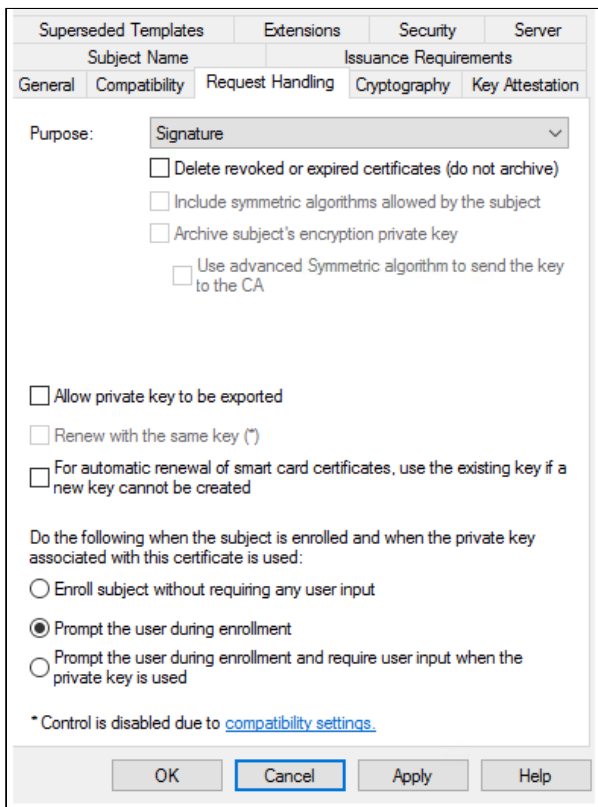
Configuring Request Handling in the Microsoft CA

If the Microsoft CA settings configure **Request Handling** as follows.

Parameter	Value
Purpose	Signature

Parameter	Value
Delete revoked or expired certificates	
Allow private key to be exported	
For automatic renewal of smart card certificates, use the existing key if a new key cannot be created	
Do the following when the public subject is enrolled and when the private key associated with this certificate is used	Prompt the user during enrollment

As we see, the **Archive subject's encryption private key** option is disabled when selecting the **Signature** template.



Superseded Templates Extensions Security Server

Subject Name Issuance Requirements

General Compatibility Request Handling Cryptography Key Attestation

Purpose: Signature

Delete revoked or expired certificates (do not archive)

Include symmetric algorithms allowed by the subject

Archive subject's encryption private key

Use advanced Symmetric algorithm to send the key to the CA

Allow private key to be exported

Renew with the same key (*)

For automatic renewal of smart card certificates, use the existing key if a new key cannot be created

Do the following when the subject is enrolled and when the private key associated with this certificate is used:

Enroll subject without requiring any user input

Prompt the user during enrollment

Prompt the user during enrollment and require user input when the private key is used

* Control is disabled due to [compatibility settings](#).

OK Cancel Apply Help

Enabling SAN attributes in the enrollment request

For Microsoft CA to construct the `SubjectAltName` in the issued certificate, you must enable the following flag.

```
Config_CA_Accept_Request_Attributes_SAN
```

You can enable this flag in your remote certificate services implementation or the Microsoft CA server machine, as explained below.

See the [MS-CSRA] Microsoft document for more information on this flag.

To enable Config_CA_Accept_Request_Attributes_SAN in the Microsoft CA machine

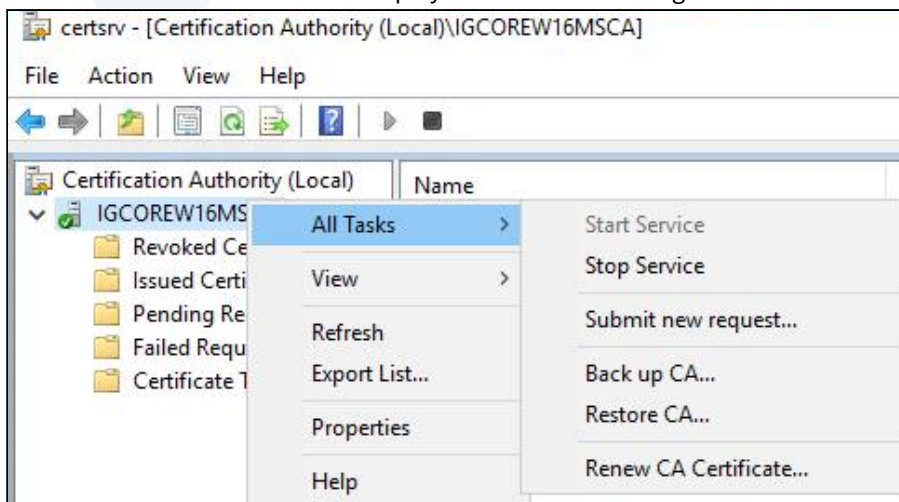
1. Log into the Windows machine hosting the Microsoft CA server.
2. Run the `regedit` command to open the Registry Editor.
3. Select the following registry key (`<CA_CN>` is the Common Name of the Microsoft CA).

```
HKLM\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration<CA_CN>/
PolicyModules\CertificateAuthority_MicrosoftDefault.Policy/EditFlags
```

4. Calculate an OR of the current key value and `0x00040000` . For example, if the current value is `11014e` , calculate:

```
0x00011014e OR 0x00040000 = 0x0001514e
```

5. Set the OR result as the new key value.
6. Run the `certsrv` command to display the CA service settings.



7. In the navigation tree, right-click the CA name.
8. Select **All Tasks > Stop service** to stop the Microsoft CA server.
9. Select **All Tasks > Start service** to restart the Microsoft CA server.

Integrating an ECS CA

See below for configuring CA Gateway for integrating CAs of the Entrust Certificate Services (ECS).

- [Issuing the SSL certificate](#)
- [Creating the API username and key](#)
- [Adding tracking information to the certificate requests](#)

See the [CA Capabilities reference](#) for a complete description of the operations supported by these CAs.

Issuing the SSL certificate

Generate the SSL certificate that CA Gateway will use to authenticate enrollment operations with the ECS-managed CA.

- [Generating the key pair](#)
- [Generating the certificate signing request](#)
- [Issuing the certificate](#)
- [Generating the SSL PKCS#12](#)

Generating the key pair

Generate the key pair.

```
openssl genrsa -out key.pem 2048
```

Generating the certificate signing request

Generate the certificate signing request.

```
openssl req -new -key key.pem -out csr.pem
```

When requested for the Common Name, enter a domain or subdomain verified in your account.

Issuing the certificate

Process the certificate signing request to issue a certificate.

To issue the certificate

1. As a Super Admin user, log in to the ECS Portal.
2. Navigate to **Create > SSL/TLS**
3. In the create wizard, paste the generated PEM request contents.
4. Select one of the following extended key usages:
 - **Client Authentication**
 - **Client and Server Authentication.**
5. Complete the wizard steps.
6. Navigate to **Certificates > Managed Certificates > ECS Certificates.**
7. Record the **Tracking ID** value for future use.
8. Go to **Actions > Pickup.**
9. Type the password, if required.
10. Select the **WS_FTP** server type.
11. Download a Zip file containing the issued certificate, the certification chain, and the root certificate.

Generating the SSL PKCS#12

Generate a PKCS#12 containing the SSL keys and certificates. For example:

```
openssl pkcs12 -export -in ServerCertificate.crt -certfile chain.pem -inkey key.pem -out restapi.p12
```

You will later set this PKCS#12 as either a file path or a base64 encoding. To encode the PKCS#12 in Base64, run:

```
base64 restapi.p12 -w 0 > restapi.txt
```

Where the `-w 0` option formats the output as one line without line breaks.

Creating the API username and key


Create a username and a key for authenticating with the Entrust Certificate Services API.

To create the API username and key

1. Log in to the Entrust Certificate Services portal as a Super Admin user.
2. Navigate to **Administration > Advanced Settings > Localization**.
3. Select **English** in the **Account language** list and click **Save**.

 As reported in the 3.0.1 release notes, the CA Gateway API may return the "Unspecified" certificate revocation reason when selecting other locales.

4. Navigate to **Administration > Advanced Settings > API**
5. Use the recorded Tracking ID value to select the SSL certificate.
6. Click on **Generate credentials**
7. Record the displayed username and key.

 The system will not display the key again.

Adding tracking information to the certificate requests

In addition to the fields required by the CA Gateway API, certificate requests for ECS CAs must include tracking information. For example:

```
"properties": {
  "requesterEmail": "requester@mail.com",
  "requesterPhone": "123456789",
  "requesterName": "Request Name",
  "trackingInfo": "tracking info test",
  "additionalEmails": "test1@mail.com, test2@mail.com",
  "text4": "this is custom text 4",
  "date1": "2022-07-01T12:24:27.627Z",
  "number3": 33
}
```

As we see in the example, this tracking information can include custom fields to meet customer requirements.

Integrating a Security Manager CA

To connect and perform operations with an Entrust Security Manager CA, CA Gateway requires an administrator profile issued by the Security Manager CA. For information about creating this administrator profile, see the following sections.

- [Enabling TLS 1.0 and TLS 1.1](#)
- [Creating a certificate type for the administrator profile](#)
- [Creating a new certificate definition policy for the certificate type](#)
- [Mapping the certificate definition policy to the certificate type](#)


- [Creating a client policy for the administrator profile](#)
- [Creating a role for the administrator profile](#)
- [Creating a user entry for the administrator profile](#)
- [Creating the administrator profile](#)
- [Backfilling the Security Manager database with user certificate state changes](#)

See the [CA Capabilities reference](#) for a complete description of the operations supported by these CAs.

Enabling TLS 1.0 and TLS 1.1

Some early versions of Security Manager require older versions of TLS disabled by default. To enable these versions:

1. Edit the `application.yml` configuration file.
2. Enable the selected TLS versions under `cagw.deploy.enable`.

 See [CA Gateway - Configuration Reference](#) for a description of each `application.yml` configuration parameter.

Creating a certificate type for the administrator profile

Create a certificate type for the administrator profile CA Gateway will use to connect and perform operations with Security Manager CA.

To create a certificate type for the administrator profile

1. Export the certificate specifications from the Security Manager CA:
 - a. Log in to Security Manager Administration for the Security Manager CA.
 - b. Select **File > Certificate Specifications > Export**.
 - c. Save the file to a location on the computer.
2. Open the certificate specifications file in a text editor.
3. Add the following to the `[Certificate Types]` section:

```
ent_cagwxap_rsa1=enterprise,CAGW Admin,CA Gateway XAP Administrator
```

4. Add the following to the `[Extension Definitions]` section:

```
[ent_cagwxap_rsa1 Certificate Definitions]
1=Dual Usage; Single key dual usage key pair Certificate Type
[ent_cagwxap_rsa1 Dual Usage Extensions]
keyusage=2.5.29.15,c,m,BitString,101; digitalSignature(0) and
keyEncipherment(2)
; Encodes the entAdminServicesClients policy OID (2.16.840.1.114027.10.4)
certificatepolicies=2.5.29.32,n,o,DER,300D300B06096086480186FA6B0A04
```

5. Save and close the file.
6. Import the certificate specifications back into the Security Manager CA:
 - a. Log in to Security Manager Administration for the Security Manager CA.
 - b. Selecting **File > Certificate Specifications > Import**.
 - c. Select the file you edited earlier.

Creating a new certificate definition policy for the certificate type

The certificate type created in [Creating a certificate type for the administrator profile](#) has a Dual Usage certificate definition. You must create a new certificate definition policy for this certificate definition that disables private key backup and enforces generating the key at the client application.

To create a new certificate definition policy for the new certificate type

1. Log in to Security Manager Administration for the Security Manager CA.
2. In the tree view, expand **Security Policy > User Policies**.
3. Select **Dual Usage Policy**.
4. Select **Policies > User Policies > Selected User Policy > Copy**.
The **Copy User Policy** dialog box appears.
5. In the **Label** field, enter `Dual Usage CAGW Admin Policy`.
6. In the **Common name** field, enter `Dual Usage CAGW Admin Policy`.
7. In the **Add to** drop-down list, select the searchbase where you want to store the user policy.
8. Under **Policy Attributes**:
 - Deselect **Backup private key**.
 - Select **Generate key at client**.
9. Click **OK**.
10. If prompted, authorize the operation. The operation may require more than one authorization. See the Security Manager Administration documentation for details.

Mapping the certificate definition policy to the certificate type

After creating a certificate definition policy, you must map this certificate definition policy to the certificate type.

To map the certificate definition policy to the certificate type

1. Log in to Security Manager Administration for the Security Manager CA.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > Certificate Types > CAGW Admin > Dual Usage**.
3. In the **Certificate definition policy** drop-down list, select **Dual Usage CAGW Admin Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation. The operation may require more than one authorization. See the Security Manager Administration documentation for details.

Creating a client policy for the administrator profile

Create a client policy for the administrator profile CA Gateway will use to connect and perform operations with Security Manager CA.


To create a new client policy for the administrator profile

1. Log in to Security Manager Administration for the Security Manager CA.
2. In the tree view, expand **Security Policy > User Policies**.
3. Select **Administrator Policy**.
4. Select **Policies > User Policies > Selected User Policy > Copy**.
The **Copy User Policy** dialog box appears.
5. In the **Label** field, enter `CAGW Admin Policy`.
6. In the **Common name** field, enter `CAGW Admin Policy`.
7. In the **Add to** drop-down list, select the searchbase where you want to store the user policy.
8. Under **Policy Attributes**, select **Permit Server Login usage**.
9. Click **OK**.
10. If prompted, authorize the operation. The operation may require more than one authorization. See the Security Manager Administration documentation for details.

Creating a role for the administrator profile

To connect and perform operations with a Security Manager CA, CA Gateway requires an administrator profile issued by the Security Manager CA. This profile must have a role with the following permissions.

Permission category	Permissions
Certificates	Administer at least one certificate category. Currently, CA Gateway supports only Enterprise certificate types.
Certificate Types	Administer at least one certificate type.
Groups	<ul style="list-style-type: none"> • View • Administer at least one group
License Information	View
Roles	<ul style="list-style-type: none"> • View • Administer at least one role.
Searchbases	<ul style="list-style-type: none"> • View • Administer at least one searchbase.
Security Policy	<ul style="list-style-type: none"> • Force CRLs • View User Policy • View Security Policy • Export Certificate Specification
User Templates	Administer at least one template
User - General	<ul style="list-style-type: none"> • View • Add • Reactivate • Deactivate/Remove • Change DN • Modify properties • Revoke certificates • Update key pairs • Set for key recovery • Cancel key recovery • Modify key update options • View activation code • Reissue activation code
User - Advanced	Change user's role

 Refer to the Security Manager Administration documentation for more details on role configuration.

To create a new role for the administrator profile

1. Log in to Security Manager Administration for the Security Manager CA.
2. In the tree view, expand **Security Policy > Roles**.
3. Select **Policies > Roles > New** to create a new role. Alternatively, you can copy the **Administrator** role because this role includes most of the permissions required for the new role.
 - a. Select **Administrator**.
 - b. Select **Policies > Roles > Selected Role > Copy** . A copy of the role appears at the bottom of the list of roles in the tree view, and the new role’s properties appear in the right pane.
4. Click the **Role** tab.
 - a. Into the **Unique name** field, enter `CAGW Admin Role` .
 - b. In the **Authorizations** field, enter 1.
 - c. In the **User Policy** drop-down list, select **CAGW Admin Policy**. This is the client policy you created earlier.
 - d. Unselect the **End User** check box. This check box should already be deselected.
5. Click the **Permissions** tab.
6. Configure the permissions documented in the above table and click **Apply**.
7. If prompted, authorize the operation. As explained in the Security Manager Administration documentation, the operation may require more than one authorization.
8. A **Permission Dependencies** pop-up dialog may list additional permissions required for the role to function properly. Add these missing permissions to the role.

Creating a user entry for the administrator profile

Create a user entry in Security Manager for the administrator profile.

To create a user entry for the administrator profile

1. Log in to Security Manager Administration for the Security Manager CA.
2. Select **Users > New User** to display the **New User** dialog.
3. Select the following tabs to configure the corresponding fields.
 - [Naming](#)
 - [General](#)
 - [Certificate Info](#)
 - [Key Update Options](#)
4. Click **OK**.
5. If prompted, authorize the operation. The operation may require more than one authorization. See the Security Manager Administration documentation for details.
6. Copy the reference number and authorization code required to create the administrator profile. You will require them later to create and activate the user’s Entrust digital ID. For more details about how the Registration number and Authorization codes are used, see the Security Manager Administration documentation.

Naming

Configure the following fields under this tab.

Field	Value
Type	Select a user type.

Field	Value
User fields	Enter a value for all configuration fields of the selected user type.
Add to	Select a searchbase for the user – for example, select CA Domain Searchbase to add the user entry to the default searchbase.

General

Configure the following fields under this tab.

Field	Value
User role	Select the role described in Creating a role for the administrator profile .
User group(s)	Assign the user to one or more groups.

Certificate Info

Configure the following fields under this tab.

Field	Value
Category	Select Enterprise .
Certificate Type	Select the role described in Creating a role for the administrator profile .

Key Update Options

Under this tab, enable the **Use default key update policy** option.

Creating the administrator profile

To connect and perform operations with a Security Manager CA, CA Gateway requires an administrator profile that is issued by the Security Manager CA.

To create the administrator profile

1. Install JDK (Java Development Kit) 17 and set the `JAVA_HOME` environment library.
2. Log in to trustedcare.entrust.com
3. Go to **PKI > Authority > CA Gateway**.
4. Download the Profile Creation Utility for your preferred operating system:
 - `cagw-profilecreationutility-linux64-version.zip` for Linux 64-bit.
 - `cagw-profilecreationutility-win64-version.zip` for Windows 64-bit.
5. Extract the file contents.

6. Run the CA Gateway Profile Creation Utility as explained in the following sections.
 - [Creating the administrator profile on software](#)
 - [Using the Profile Creation Utility to create the administrator profile on hardware](#)

Creating the administrator profile on software

As explained in this section, you can store the administrator profile in software as an Entrust Profile File (EPF).

To create the administrator profile on software

1. Run the `<VERSION>` version of the CA Gateway Profile Creation Utility.
 - `cagw-profilecreationutility-<VERSION>/bin/pcu.sh` for Linux.
 - `cagw-profilecreationutility-<VERSION>/bin/pcu.bat` for Windows.
2. Once on the main menu, select option **2** for **Create Entrust profile** .
3. Select option **1** for **File on disk**
4. In **Take settings from an existing entrust.ini file (y/n)?** enter **y** for yes.
5. In **Enter full path to entrust.ini**, enter the path of the local file.
6. In **Enter reference number**, enter the reference number you obtained when creating a user entry for the administrator profile.
7. In **Enter authorization code**, enter the authorization code you obtained when creating a user entry for the administrator profile.
8. In **Enter profile name**, enter a file name for the EPF file. Do not include a file name extension because the utility automatically appends a .epf extension. If you include a .epf extension in the name, it will be added to the file twice.
9. In **Enter profile directory**, enter the directory for the EPF file. The name of this file is the name previously entered in **Enter profile name**.
10. In **Enter profile password**, enter a new password to encrypt and MAC the contents of the EPF.

Using the Profile Creation Utility to create the administrator profile on hardware

For information about creating the administrator profile on hardware, see the CA Gateway integration guide for your hardware security module (HSM).

Backfilling the Security Manager database with user certificate state changes

Security Manager 8.3.30 introduced a feature for tracking certificate events. This new feature was added to support Entrust CA Gateway and some clients, such as Entrust Certificate Hub. When the state of a certificate changes – for example, from ACTIVE to ON HOLD – the new state is recorded in the `UserCertTracking` database table. This feature:

- Applies only to certificate state changes in release 8.3.30 or later.
- Does not capture certificate state changes that occurred before release 8.3.30.

To get a complete certificate status history, run a script that backfills the `UserCertTracking` table in the Security Manager database with the missing certificate status events. Security Manager must be one of the following releases:

- Security Manager 8.3.62 or later
- Security Manager 10.0.30 or later

For information about backfilling the UserCerttracking table in Security Manager, see knowledge article 89407 on Entrust TrustedCare:

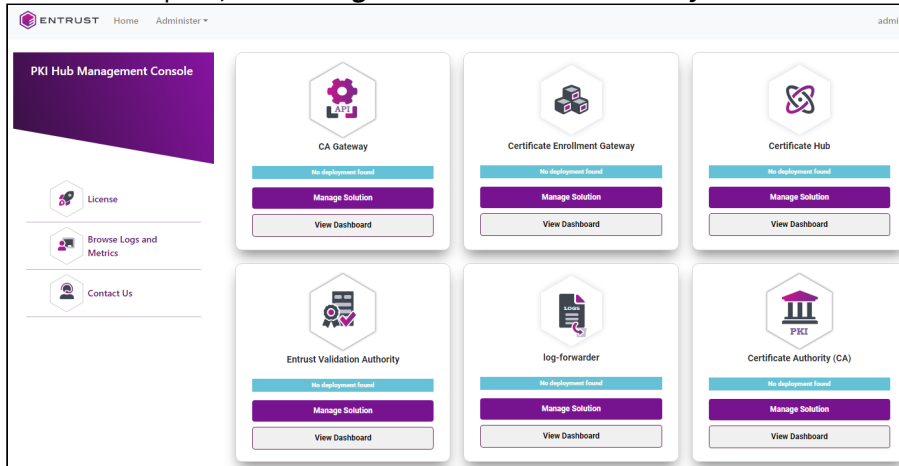
https://trustedcare.entrust.com/articles/en_US/Technote/Backfilling-the-UserCertTracking-table-in-the-Security-Manager-database-for-CA-Gateway

Configuring and deploying CA Gateway

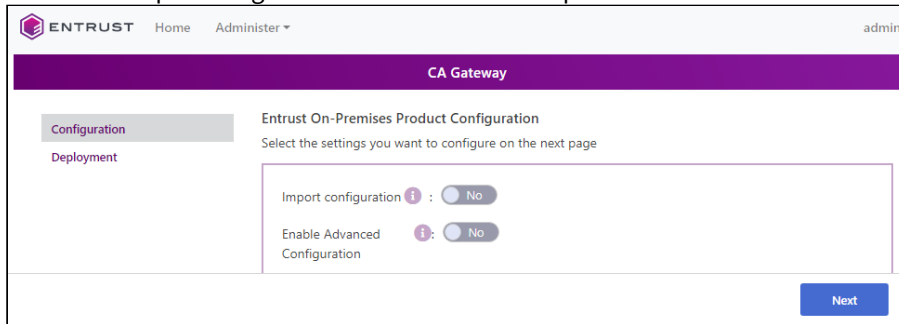
See below for configuring and deploying CA Gateway with the Management Console.

To configure and deploy CA Gateway with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).
2. In the content pane, click **Manage Solution** under **CA Gateway**.



3. Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



4. Click **Next**.
5. Configure the solution settings described in the following sections.
 - [Logging](#)
 - [Server](#)
 - [Connector filters](#)
 - [Authorities](#)
 - [Profiles](#)
 - [Tenants](#)
 - [Clients](#)
 - [Cmpv2](#)
 - [TLS CRL-settings](#)

1. Click **Validate** to validate the configured settings.
2. Correct any detected configuration error until the **Validate** option displays no warnings.
3. Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
4. Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
5. Click **Deploy**.

Logging

Set the following parameters in the **Logging** tab of the **Configuration** page.

- [CAGW Logging](#)
- [JTK Logging](#)
- [JSSE Logging](#)

CAGW Logging

Configure the following setting under this section.

Level

The level of detail for the root CA Gateway's logger. In increasing severity:

- TRACE
- DEBUG
- INFO
- WARN
- ERROR
- FATAL
- OFF

Each level sets the lowest message level to show. For example, the `WARN` level provides messages with the `WARN`, `ERROR`, and `FATAL` status.

Required: No. This optional parameter defaults to `INFO`.

JTK Logging

Configure the following parameter under this section.

- [Enabled](#)
- [JTK Logs Level](#)

Enabled

`true` to enable JSTK logging, `false` otherwise.

Required: No. This optional parameter defaults to `false`.

JTK Logs Level

The level of detail for the JSTK logs.

- 0
- 1
- 2
- 3
- 4

Required: No. This optional parameter defaults to 0.

JSSE Logging

Configure the following parameters under this section.

- [Enabled](#)

- [JSSE Logs Level](#)

Enabled

`true` to enable JSSE (Java Secure Socket Extension) logging, `false` otherwise.

Required: No. This optional parameter defaults to `false`.

JSSE Logs Level

The level of detail for the JSSE logs. Supported values are:

- `ssl`
- `ssl:handshake`
- `all`

Required: No. This optional parameter defaults to `ssl`.

Server

Set the following parameters in the **Server** tab of the **Configuration** page.

- [Comma-separated list of Ciphers](#)
- [Key Alias](#)
- [Key Store](#)
- [Key Store Password](#)
- [Key Store Type](#)
- [Trust Store Type](#)
- [Trust Store](#)
- [Trust Store Password](#)

Comma-separated list of Ciphers

The list of allowed SSL ciphers – for example:

```
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA,TLS_RSA_WITH_AES_128_GCM_SHA256
```

Key Alias

The alias of the SSL key in the keystore.

Mandatory: Yes.

Key Store

The keystore that contains the SSL server certificate. Click **Select Files** to import this keystore from file.

Key Store Password

The password of the keystore that contains the server SSL certificate.

Mandatory: Yes.

Key Store Type

The type of keystore containing the SSL server certificate.

Type	Description
jks	Java keystore
pkcs12	PKCS #12 keystore

Mandatory: Yes.

Trust Store Type

The type of truststore containing the CA certificates.

Type	Description
jks	Java truststore
pkcs12	PKCS #12 truststore

Mandatory: Yes.

Trust Store

Click **Select Files** to import the truststore file.

Trust Store Password

The password of the truststore that contains the CA certificates.

Mandatory: Yes.

Connector filters

For each connector filter, set the following parameters in the **Connector filters** tab of the **Configuration** page.

- [Name](#)
- [Connector name](#)
- [Filter Settings](#)

Name

A descriptive name of the filter.

Mandatory: Yes

Connector name

Select the name of the filter connector

- [com.entrust.CAAuthorization](#)
- [com.entrust.CertificateEvents](#)

- [com.entrust.CertTransparency](#)

Mandatory: Yes

com.entrust.CAAuthorization

Filter to conduct CA Authorization checks for certificates intended for public trust. When selecting this filter, configure the following settings under [Filter Settings](#):

- [check-domains-external-to-cs](#)
- [check-domains-from-csr](#)
- [dns-server<i>.<setting>](#)
- [issuer-string](#)
- [log-server.<i>.<setting>](#)

com.entrust.CertificateEvents

Convenience filter to:

1. Read a certificate.
2. Extract data from the certificate.
3. Add the data to the response so that the caller does not have to immediately decode the certificate.

This filter does not require configuring [Filter Settings](#).

com.entrust.CertTransparency

Filter to:

1. Collect a set of signed CT log server responses.
2. Ask the underlying CA if the certificates for public trust include these responses in an SCT List extension.

When selecting this filter, configure the following settings under [Filter Settings](#):

- [connection-timeout-millis](#)
- [ct-policy-json](#)
- [log-server.<i>.<setting>](#)
- [proxy-host-name \(filter\)](#)
- [proxy-port](#)
- [socket-timeout-millis](#)

Filter Settings

Configure the settings required by each connector selected in the [Connector name](#) list.

- [check-domains-external-to-cs](#)
- [check-domains-from-csr](#)
- [connection-timeout-millis](#)
- [ct-policy-json](#)
- [dns-server<i>.<setting>](#)
- [issuer-string](#)
- [log-server.<i>.<setting>](#)
- [proxy-host-name](#)
- [proxy-port](#)
- [socket-timeout-millis](#)

check-domains-external-to-cs

`true` for CA Gateway to make CAA checks for domains in the subjectAltNames field external to the CSR, `false` otherwise.

Mandatory: No. This optional parameter defaults to `true`.

check-domains-from-csr

`true` for CA Gateway to make CAA checks for domains inside the CSR, `false` otherwise.

Mandatory: No. This optional parameter defaults to `true`.

connection-timeout-millis

The connection timeout for the HTTP communication with the log server, in milliseconds.

Mandatory: No. This optional parameter defaults 5000 milliseconds.

ct-policy-json

The number of log server responses CA Gateway must wait for.

i CA Gateway can cope with slow running or unresponsive log servers when the number of servers configured under `log-server.<i>.<setting>` exceeds the number of required responses.

The general form of this JSON value is:

```
{
  sct-policy: [
    [<months-threshold>, <threshold-equals>, <google-min-responses>, <non-google-min-responses>]
  ],
  insurance: <insurance>
}
```

See the following table for a description of each parameter.

Parameter	Value
months-threshold	The applicability of the <code>sct-policy</code> policy according to the certificate lifetime, as a number of months. When defining multiple policies, this value determines which policy to apply for issuing a certificate. On the other hand, specifying a high value ensures this policy applies to all certificates issued.
threshold-equals	<code>true</code> for comparing the months-threshold and the actual certificate lifetime with the equals operator ('='); <code>false</code> for comparing with the less than or equals operator ('<=').

Parameter	Value
google- min- responses	The minimum number of Google-compatible log server responses to include in the issued certificate.
non-google- min-responses	The minimum number of non-Google-compatible log server responses to include in the issued certificate.
insurance	The number of log server responses to collect above the following minimum: <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <code>google-min-responses + non-google-min-responses</code> </div>

Mandatory: No. This optional parameter defaults to:

```
{
  sct-policy:[
    [39,true,0,1]
  ],
  insurance:0
}
```

In the configuration, you can flatten this default value to:

```
{sct-policy:[[38,true,0,1]],insurance:0}
```

dns-server<i><setting>

The DNS settings, where "i" is an index starting at 0. You can omit this index when defining a single DNS.

<setting>	Value	Default
ip	The IP address of the local DNS server that CA Gateway will use to look up the DNS issuer resource record.	-
port	The port of the DNS server.	53
timeout-first-seconds	The timeout of the first DNS lookup attempt, in seconds.	3
timeout-second-seconds	Timeout of the second DNS lookup attempt, in seconds. Applicable if the first attempt results in an error.	7

<setting>	Value	Default
timeout-dsquery-seconds	Timeout in seconds of the Delegation Signer (DS) query when querying DNSSEC support.	7

Mandatory: Yes.

issuer-string

The CAA issuer name, as expected in the DNS resource record. Real-world examples include:

- entrust.net
- pki.goog

The name is owned and defined by the issuer and registered in DNS for any CA to check.

Mandatory: Yes

log-server.<i>.<setting>

The settings of each log server CA Gateway must contact to request the signed CT response. Therefore, you must define at least one server, with <i> starting a 0.

<setting>	Value
name	A friendly name for the log server. For example: "Google Log Server".
url	The URL of the log server
google	True if the SCTs produced by this log server are Google Chrome compatible.
public-key	The public key of the log server, as a Base64 DER-encoded public key. Log servers typically advertise their keys publicly.
tls-trust-anchor	The trust anchor for the CT Filter to perform the TLS handshake with the log server, as a Base64 DER-encoded certificate.

Mandatory: Yes.

proxy-host-name

The hostname of the proxy for routing log server communications.

Mandatory: No.

proxy-port

The proxy port for accessing the [proxy-host-name \(filter\)](#) proxy.

Mandatory: Only when using the [proxy-host-name \(filter\)](#) proxy.

socket-timeout-millis

The TCP Socket timeout for the HTTP communication with the log server, in milliseconds.

Mandatory: No. This optional parameter defaults 5000 milliseconds.

Authorities

For each certificate authority, set the following field values in the **Authorities** tab of the **Configuration** page.

- [Minimum keysize](#)
- [Authority settings](#)

Minimum keysize

Configure the minimum sizes supported by all authority keys.

- [Specify whether to disable, log or reject key size violations](#)
- [Minimum RSA Keysize](#)
- [Minimum ECC Keysize](#)

Specify whether to disable, log or reject key size violations

The performed action when an authority key does not meet the [Minimum RSA Keysize](#) or [Minimum ECC Keysize](#) size requirements.

Value	Action
off	Nothing
log	Log the key size
block	Reject the key

Minimum RSA Keysize

The minimum size for all authority RSA keys, as a number of bytes.

Minimum ECC Keysize

The minimum size for all authority ECC (elliptic curve) keys, as a number of bytes.

Authority settings

To register an authority, click the + button and configure the following settings.

- [Choose a key name](#)
- [Name](#)
- [Issuer DN](#)
- [Minimum keysize](#)
- [Connector Name](#)

Choose a key name

Type a unique identifier for the authority.

Mandatory: Yes.

Name

A friendly name for the CA.

Mandatory: Yes

Issuer DN

The Distinguished Name (DN) of the CA. For example:

```
CN = Certificate Authority, O = Entrust, Inc, C = US
```

```
CN = "Entrust Class 2 Client CA", OU = "(c) 2010 Entrust, Inc.", OU =
"www.entrust.net/CPS is incorporated by reference", O = "Entrust, Inc.", C = US
```

As explained in [RFC 2253](#), you can surround the value of each DN field with quote (" ASCII 34) characters, which are not part of the value. Inside the quoted value, the following characters can occur without any escaping:

- "
- ,
- "="
- "+"
- "<"
- ">"
- "#"
- ";"

Mandatory: Yes.

Minimum keysize

Configure the minimum sizes supported by this specific authority authority key.

- [Specify whether to disable, log or reject key size violations](#)
- [Minimum RSA Keysize](#)
- [Minimum ECC Keysize](#)

Specify whether to disable, log or reject key size violations

The performed action when this authority key does not meet the [Minimum RSA Keysize](#) or [Minimum ECC Keysize](#) size requirements.

Value	Action
off	Nothing
log	Log the key size
block	Reject the key

Minimum RSA Keysize

The minimum size for this authority RSA key, as a number of bytes.

Minimum ECC Keysize

The minimum size for this authority ECC (elliptic curve) key, as a number of bytes.

Connector Name

The CA connector name. See below for the supported values.

- [com.entrust.ECS](#)
- [com.entrust.MicrosoftCA](#)
- [com.entrust.SecurityManager](#)

com.entrust.ECS

To integrate an ECS certificate authority, select this connector and configure the following settings.

- [ECS URL](#)
- [User Name](#)
- [API Key](#)
- [Enrollment Agent PKCS#12 File](#)
- [Enrollment Agent PKCS#12 Password](#)
- [CA Certificate](#)
- [CA Certificate Chain](#)
- [Client ID defined in ECS for all domain operations](#)
- [Proxy Hostname](#)
- [Proxy Port](#)
- [Proxy username](#)
- [Proxy password](#)

ECS URL

Set this parameter to:

```
https://api.entrust.net/enterprise/v2
```

Mandatory: Yes.

User Name

The API username for consuming the ECS CA services. See the CA Gateway guide for how to obtain this username.

Mandatory: Yes.

API Key

The API key for consuming the ECS CA services. See the CA Gateway guide for how to obtain this key.

Mandatory: Yes.

Enrollment Agent PKCS#12 File

The PKCS#12 file containing the CA certificate. Click **Select Files** to import this file.

Enrollment Agent PKCS#12 Password

The password of the PKCS#12 file containing the CA certificate.

CA Certificate

The DER and Base64 encoding of the ECS issuing CA certificate. CA Gateway returns the selected certificate when querying the following resource with `$field` set to `ca.cert`.

```
GET /v1/certificate-authorities
```

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: Yes.

CA Certificate Chain

The DER and Base64 encoding of the certificate in the `<i>` position of the ECS CA certificate chain. For example, the certificate specified with the `ca.certchain.0` parameter is the certificate of the CA that issued the certificate specified with the `ca.cert` parameter.

CA Gateway returns the selected certificate when querying the following resource with `$field` set to `ca.chain`.

```
GET /v1/certificate-authorities
```

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: Yes.

Client ID defined in ECS for all domain operations

The client identifier defined in ECS for all domain operations sent to the ECS API.

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: No. This optional parameter defaults to 1.

Proxy Hostname

The hostname of the proxy for accessing the ECS CA server.

Mandatory: Only when traffic to the ECS CA server passes through a proxy.

Proxy Port

The port for accessing the proxy server selected with the `proxy-host-name` parameter.

Mandatory: Only when traffic to the ECS CA server passes through a proxy.

Proxy username

The username for authenticating in the proxy server selected with the `proxy-host-name` parameter.

Mandatory: Only when the proxy requires authentication.

Proxy password

The password for authenticating in the proxy server selected with the [proxy-host-name](#) parameter.

Mandatory: Only when the proxy requires authentication.

Additional ECS Properties

Click **+ ECS Properties** to add the following settings.

- [api-key](#)
- [ca.cert](#)
- [ca.certchain.<i>](#)
- [client-id-domains](#)
- [ecs-url](#)
- [enrollment-agent-p12](#)
- [enrollment-agent-p12-password](#)
- [proxy-host-name](#)
- [proxy-password](#)
- [proxy-port \(ecs\)](#)
- [proxy-username](#)
- [rdn-corrections.<i>.rep](#)
- [rdn-corrections.<i>.rep-with](#)
- [user-name](#)

api-key

The API key for consuming the ECS CA services. See the CA Gateway guide for how to obtain this key.

Mandatory: Yes.

ca.cert

The DER and Base64 encoding of the ECS issuing CA certificate. CA Gateway returns the selected certificate when querying the following resource with `$field` set to `ca.cert`.

```
GET /v1/certificate-authorities
```

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: Yes.

ca.certchain.<i>

The DER and Base64 encoding of the certificate in the `<i>` position of the ECS CA certificate chain. For example, the certificate specified with the `ca.certchain.0` parameter is the certificate of the CA that issued the certificate specified with the `ca.cert` parameter.

CA Gateway returns the selected certificate when querying the following resource with `$field` set to `ca.chain`.

```
GET /v1/certificate-authorities
```

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: Yes.

client-id-domains

The client identifier defined in ECS for all domain operations sent to the ECS API.

i You must statically configure this setting because the ECS public API does not yet allow querying certificates from the CA.

Mandatory: No. This optional parameter defaults to 1.

ecs-url

Set this parameter to:

```
https://api.entrust.net/enterprise/v2
```

Mandatory: Yes.

enrollment-agent-p12

The SSL PKCS#12, as a file path or a Base64 encoding.

Mandatory: Yes.

enrollment-agent-p12-password

The password of the SSL PKCS#12 selected with the [enrollment-agent-p12](#) parameter.

Mandatory: Yes.

proxy-host-name

The hostname of the proxy for accessing the ECS CA server.

Mandatory: Only when traffic to the ECS CA server passes through a proxy.

proxy-password

The password for authenticating in the proxy server selected with the [proxy-host-name](#) parameter.

Mandatory: Only when the proxy requires authentication.

proxy-port

The port for accessing the proxy server selected with the [proxy-host-name](#) parameter.

Mandatory: Only when traffic to the ECS CA server passes through a proxy.

proxy-username

The username for authenticating in the proxy server selected with the [proxy-host-name](#) parameter.

Mandatory: Only when the proxy requires authentication.

rdn-corrections.<i>.rep

A distinguished name (DN) attribute you want to rename using the [rdn-corrections.<i>.rep-with](#) parameter.

Specifically, some Entrust Certificate Services profiles may include legacy attribute names in the subject of the issued certificates. However, these attribute names may not be compatible with the industry-standard names used by some client applications.

Entrust Certificate Services legacy attribute name	Industry-accepted attribute name
jurisdictionOfIncorporationStateOrProvinceName	jurisdictionStateOrProv
jurisdictionOfIncorporationCountryName	jurisdictionCountryName

In this case, add the following lines to the CA Gateway configuration.

```
rdn-corrections.0.rep: jurisdictionCountryName
rdn-corrections.0.rep-with: jurisdictionOfIncorporationCountryName
rdn-corrections.1.rep: jurisdictionStateOrProvinceName
rdn-corrections.1.rep-with: jurisdictionOfIncorporationStateOrProvinceName
```

Before sending certificate renewal requests to Entrust Certificate Services, CA Gateway will apply this configuration and replace industry-compliant subject attributes with legacy ones.

Example of subject name with industry-compliant attribute names

```
CN=test.com, serialNumber=705421, businessCategory=Private Organization, O=Entrust Corporation, jurisdictionStateOrProv=Delaware, jurisdictionCountryName=US, L=Shakopee, ST=Minnesota
```

Example of subject name with Entrust Certificate Services legacy attribute names

```
CN=test.com, serialNumber=705421, businessCategory=Private Organization, O=Entrust Corporation, jurisdictionOfIncorporationStateOrProvinceName=Delaware, jurisdictionOfIncorporationCountryName=US, L=Shakopee, ST=Minnesota
```

Mandatory: Only when renewing certificates with Entrust Certificate Services.

rdn-corrections.<i>.rep-with

A new name for the distinguished name (DN) attribute you selected with the [rdn-corrections.<i>.rep](#) parameter.

 See the [rdn-corrections.<i>.rep](#) parameter reference for an example of how to use both parameters.

Mandatory: Only when renewing certificates with Entrust Certificate Services.

user-name

The API username for consuming the ECS CA services. See the CA Gateway guide for how to obtain this username.

Mandatory: Yes.

com.entrust.MicrosoftCA

To integrate a Microsoft certificate authority, select this connector and configure the following settings.

- CA Proxy URL
- CA Host
- CA Name
- LDAP Port
- LDAPS Port
- LDAP Host
- Key Recovery Agent PKCS#12
- Key Recovery Agent PKCS#12 Password
- Client Certificate Key Alias
- Client Certificate Keystore Type
- Client Certificate Keystore File
- Client Certificate Keystore Password
- SSL Truststore Type
- SSL Truststore File
- SSL Truststore Password
- Additional Microsoft CA Properties

CA Proxy URL

The URL of the Entrust Proxy for Microsoft CA, in the following format:

```
https://<server>:8443/MSCAProxy
```

Mandatory: Yes.

CA Host

The CA hostname, as either:

- An IP
- A hostname
- A FQDN

As long as it resolves from the DNS.

Mandatory: Yes.

CA Name

The CA name – for example:

```
abc-issuing
```

Mandatory: Yes.

LDAP Port

The port number for LDAP connections with Microsoft Active Directory (for LDAPS connections, configure [LDAPS Port](#) instead).

i The port is anonymously bound. The Microsoft CA proxy connects to Active Directory to get certificate template information.

This value is typically 389, the well-known port for LDAP.

Mandatory: When not configuring [LDAPS Port](#).

LDAPS Port

The port number for LDAPS connections with Microsoft Active Directory (for LDAP connections, configure [LDAP Port](#) instead).

i The port is anonymously bound. The Microsoft CA proxy connects to Active Directory to get certificate template information.

This value is typically 636, the well-known port for LDAPS.

Mandatory: When not configuring [LDAP Port](#).

LDAP Host

The Microsoft Active Directory, as an IP, a hostname, or an FQDN (as long as it resolves from the DNS). The host must be in the [CA Host](#) domain because:

- CA Gateway only talks to the Entrust Proxy for Microsoft CA.
- The Entrust Proxy for Microsoft CA is on the CA's same domain and talks to the CA.

Mandatory: Yes.

Key Recovery Agent PKCS#12

The path of the key PKCS#12 generated when creating the RA recovery agents (if any). Where `<i>` is an integer greater than or equal to 0.

Mandatory: Only when creating the RA recovery agents.

Key Recovery Agent PKCS#12 Password

The password of the key recovery agent PKCS#12.

Mandatory: Only when creating the RA recovery agents.

Client Certificate Key Alias

The alias of the CA Gateway client key.

Mandatory: Yes.

Client Certificate Keystore Type

Set this parameter to:

JKS

Mandatory: Yes.

Client Certificate Keystore File

The filename of the CA Gateway client JKS.

Mandatory: Yes.

Client Certificate Keystore Password

The password of the CA Gateway client JKS.

Mandatory: Yes.

SSL Truststore Type

The type of CA Gateway trust store. Supported values are:

- JKS
- PKCS12

Mandatory: Yes.

SSL Truststore File

The CA Gateway truststore. Click **Select Files** to import this truststore from file.

SSL Truststore Password

The password of the CA Gateway trust store.

Mandatory: Yes.

Additional Microsoft CA Properties

Click + **Microsoft CA Properties** to add the following settings.

- [ca-host](#)
- [ca-name](#)
- [ca-proxy-url](#)
- [key-recovery-agent-p12-<i>](#)
- [key-recovery-agent-p12-password-<i>](#)
- [ldap-host](#)
- [ldap-port](#)
- [ldaps-port](#)
- [proxy-host-name](#)
- [proxy-password](#)
- [proxy-port](#)
- [proxy-ssl](#)
- [proxy-username](#)

ca-host

The CA hostname, as either:

- An IP
- A hostname
- A FQDN

As long as it resolves from the DNS.

Mandatory: Yes.

ca-name

The CA name – for example:

```
abc-issuing
```

Mandatory: Yes.

ca-proxy-url

The URL of the Entrust Proxy for Microsoft CA, in the following format:

```
https://<server>:8443/MSCAProxy
```

Mandatory: Yes.

key-recovery-agent-p12-<i>

The path of the key PKCS#12 generated when creating the RA recovery agents (if any). Where `<i>` is an integer greater than or equal to 0.

Mandatory: Only when creating the RA recovery agents.

key-recovery-agent-p12-password-<i>

The password of the key recovery agent PKCS#12.

Mandatory: Only when creating the RA recovery agents.

ldap-host


The Microsoft Active Directory, as an IP, a hostname, or an FQDN (as long as it resolves from the DNS). The host must be in the `ca-host` domain because:

- CA Gateway only talks to the Entrust Proxy for Microsoft CA.
- The Entrust Proxy for Microsoft CA is on the CA's same domain and talks to the CA.

Mandatory: Yes.

ldap-port

The port number for LDAP connections with Microsoft Active Directory (for LDAPS connections, configure `ldaps-port` instead).


 The port is anonymously bound. The Microsoft CA proxy connects to Active Directory to get certificate template information.

This value is typically 389, the well-known port for LDAP.

Mandatory: When not configuring `ldaps-port`.

ldaps-port

The port number for LDAPS connections with Microsoft Active Directory (for LDAP connections, configure `ldap-port` instead).


 The port is anonymously bound. The Microsoft CA proxy connects to Active Directory to get certificate template information.

This value is typically 636, the well-known port for LDAPS.

Mandatory: When not configuring [ldap-port](#).

proxy-host-name


The hostname of the proxy for accessing the Microsoft CA server.

 The proxy configured using this parameter is part of your corporate infrastructure. Do not confuse it with the Entrust Proxy for Microsoft CA, which is selected using the [CA Proxy URL](#) parameter.

Mandatory: Only when traffic to the Microsoft CA Proxy passes through a proxy.

proxy-password


The password for authenticating in the proxy selected with the [proxy-host-name](#) parameter.

 The proxy configured using this parameter is part of your corporate infrastructure. Do not confuse it with the Entrust Proxy for Microsoft CA, which is selected using the [CA Proxy URL](#) parameter.

Mandatory: Only when the proxy requires authentication.

proxy-port

The port for accessing the proxy selected with the [proxy-host-name](#) parameter.

 The proxy configured using this parameter is part of your corporate infrastructure. Do not confuse it with the Entrust Proxy for Microsoft CA, which is selected using the [CA Proxy URL](#) parameter.

Mandatory: Only when traffic to the Microsoft Proxy passes through a proxy.

proxy-ssl

Under this section, configure the following authentication settings.

- [client-cert-key-alias](#)
- [client-cert-key-store](#)
- [client-cert-key-store-password](#)
- [client-cert-key-store-type](#)
- [ssl-trust-store](#)
- [ssl-trust-store-password](#)
- [ssl-trust-store-type](#)

client-cert-key-alias

Under this section, configure the following authentication settings.

client-cert-key-store

The filename of the CA Gateway client JKS.

Mandatory: Yes.

client-cert-key-store-password

The password of the CA Gateway client JKS.

Mandatory: Yes.

client-cert-key-store-type

Set this parameter to:

JKS

Mandatory: Yes.

ssl-trust-store

The path of the CA Gateway trust store.

Mandatory: Yes.

ssl-trust-store-password

The password of the CA Gateway trust store.

Mandatory: Yes.

ssl-trust-store-type


The type of CA Gateway trust store. Supported values are:

- JKS
- PKCS12

Mandatory: Yes.

proxy-username

The username for authenticating in the proxy selected with the [proxy-host-name](#) parameter.

 The proxy configured using this parameter is part of your corporate infrastructure. Do not confuse it with the Entrust Proxy for Microsoft CA, which is selected using the [CA Proxy URL](#) parameter.

Mandatory: Only when the proxy requires authentication.

com.entrust.SecurityManager

To integrate Entrust Authority Security Manager, select this connector and configure the following settings.

- [Security Manager Host](#)
- [PKIX Port](#)
- [LDAP Host \(sm\)](#)
- [LDAP Port \(sm\)](#)
- [LDAPS Port \(sm\)](#)
- [LDAP Principal](#)
- [LDAP Credential](#)
- [XAP Port](#)
- [Admin EPF file](#)
- [Admin EPF Password](#)
- [Initial XAP Connections](#)
- [Max XAP Connections](#)
- [XAP Connection Idle Timer \(seconds\)](#)
- [XAP Connection Socket Timer \(seconds\)](#)

- [XAP Logging](#)
- [XAP Logs Level](#)
- [P11 APF File](#)
- [P11 Library](#)
- [P11 Slot](#)
- [P11 Password](#)
- [Enable niche certificate types](#)
- [Allow 100% PKUP](#)
- [Enable CA Profile Sync](#)

Security Manager Host

The hostname of the Security Manager instance.

Mandatory: Yes

PKIX Port

The PKIX-CMP port number of the Security Manager instance

Mandatory: Yes

LDAP Host

The hostname of the directory instance.

Mandatory: Yes.

LDAP Port

The port number for LDAP connections with the Security Manager directory (for LDAPS connections, configure [LDAPS Port](#) instead).

 This value is typically 389, the well-known port for LDAP.

Mandatory: When using an LDAP connection.

LDAPS Port

The port number for LDAPS connections with the Entrust Security Manager CA (for LDAP connections, configure [LDAP Port](#) instead).

 This value is typically 636, the well-known port for LDAPS.

Mandatory: When using an LDAPS connection.

LDAP Principal

The name of the LDAP user for logging in to the directory. Save this property in secure storage such as Vault rather than directly in a configuration file.

Mandatory: Yes

LDAP Credential

The password of the LDAP user. Save this property in secure storage such as Vault rather than directly in a configuration file.

Mandatory: Yes

XAP Port

The XAP port number of the Security Manager instance.

Mandatory: Yes.

Admin EPF file

The administrator's Entrust Profile File (EPF) for connecting to the Security Manager instance. Click **Select Files** to import this file.

Mandatory: When saving the user settings in an Entrust Profile File (EPF).

Admin EPF Password

The password for decrypting the administrator's Entrust Profile File (EPF).

Mandatory: When saving the administrator's settings in an EPF.

Initial XAP Connections

The initial number of XAP connections to the Security Manager.

Mandatory: No. This optional parameter defaults to 4 connections.

Max XAP Connections

The maximum number of XAP connections to the Security Manager.

Mandatory: No. This optional parameter defaults to 20 connections.

XAP Connection Idle Timer (seconds)

The idle timeout of the Security Manager XAP connection, in seconds.

Mandatory: No. This optional parameter defaults to 30 seconds.

XAP Connection Socket Timer (seconds)

The socket timeout of the Security Manager XAP connection, in seconds.

Mandatory: No. This optional parameter defaults to 60 seconds.

XAP Logging

`true` for logging the XAP debugging to file; `false` otherwise.

Mandatory: No. This optional parameter defaults to false.

XAP Logs Level

The XAP debug log level, from 0 (no logging) to 7 (maximum logging).

Mandatory: No. This optional parameter defaults to 0.

P11 APF File

The APF (Auxiliary Profile File). Click **Select Files** to import this file.

Mandatory: When saving the user settings in a PKCS #11 hardware security module (HSM) and archiving old private keys locally (to make them available for other purposes).

P11 Library

The full path of the PKCS#11 native library.

Mandatory: When saving the user settings in a PKCS #11 hardware security module (HSM).

P11 Slot

The slot number of the PKCS#11 slot.

Mandatory: When saving the user settings in a PKCS #11 hardware security module (HSM).

P11 Password

The PKCS#11 user PIN to log in to the PKCS#11 slot.

Mandatory: When saving the user settings in a PKCS #11 hardware security module (HSM).

Enable niche certificate types

`true` to expose certificate types relating to ePassport applications and legacy software, `false` otherwise.

Mandatory: No. This optional parameter defaults to `false`.

Allow 100% PKUP

The value of the `PrivateKeyUsagePeriod` extension in certificates issued by Security Manager when the request:

- Includes the `optionalCertificateRequestDetails.validityPeriod` field, and
- Does not include the `optionalCertificateRequestDetails.privateKeyUsagePercentage` field.

See below for the values supported by this setting.

apply-full-pkup	PrivateKeyUsagePeriod
true	The 100% of the <code>optionalCertificateRequestDetails.validityPeriod</code> value.
false	Set by the CA.

i As explained in [RFC2459](#), the `PrivateKeyUsagePeriod` extension "allows the certificate issuer to specify a different validity period for the private key than the certificate".

Mandatory: No. This optional value defaults to `true`.

Enable CA Profile Sync

`true` to enable profile synchronization with the Security Manager CA, `false` otherwise. When set to `true`, CA Gateway:

- Mirrors any eligible certificate types and definitions defined in the Security Manager CA as basic CA Gateway certificate profiles without the need to define them in the CA Gateway configuration explicitly.
- Suppresses niche certificate types relating to ePassport applications and legacy software. To expose these types, enable the [Enable niche certificate types](#) parameter.

Mandatory: No.

Profiles

In the **Profiles** tab of the **Configuration** page, configure the following settings.

- [Choose a key name](#)
- [Name](#)
- [Copy CN in SubjectDN to SAN](#)
- [Subject Variable Requirements](#)
- [Subject Builder Configuration](#)
- [SAN Requirements](#)
- [Minimum keysize](#)
- [ECS Profile Properties](#)

Choose a key name

Type a unique identifier for the profile.

Mandatory: Yes.

Name

A readable name that describes the profile.

Mandatory: Yes.

Copy CN in SubjectDN to SAN

`true` to enable copying the CN of the Subject DN as Subject Alternative Name, `false` otherwise.

Mandatory: No. This optional parameter defaults to `false`.

Subject Variable Requirements

Under this section, define the subject variables for an enrollment operation with the certificate profile. When CA Gateway clients query the certificate profile, these variables inform the subject variable to supply when enrolling for a certificate using the profile.

- [Name](#)
- [Description](#)
- [Required](#)

i CA Gateway will only enforce subject variables when of [Name](#) is `com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder`.

Name

The name of the variable.

Mandatory : Yes.

Description

A friendly description of the variable.

Mandatory : Yes.

Required

`true` if the variable is required, `false` if the variable is optional.

Mandatory : Yes.

Subject Builder Configuration

Under this section, define the Subject Builders for constructing the subject DN for the enrollment request.

- [Name](#)
- [Properties](#)

Name

The class name of the subject builder.

- [com.entrust.adminservices.cagw.common.subjects.BasicSubjectBuilder](#)
- [com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder](#)
- [com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder](#)

Mandatory: No. This optional parameter defaults to [com.entrust.adminservices.cagw.common.subjects.BasicSubjectBuilder](#).

`com.entrust.adminservices.cagw.common.subjects.BasicSubjectBuilder`

Select this subject builder to append all the supplied subject variables together in the order of arrival.

Sample BasicSubjectBuilder

```
- name: "Use BasicSubjectBuilder"
  unique-id: "CA-1003-PROF-1001"
  subject-builder-config:
    subject-builder-name:
      "com.entrust.adminservices.cagw.common.subjects.BasicSubjectBuilder"
```

Sample subject variables

```
"subjectVariables" : [
  {
    "type" : "cn",
```

```
    "value" : "test"
  },
  {
    "type" : "o",
    "value" : "pki"
  }
]
```

Subject DN generated by the sample builder when parsing the sample variables

```
cn=test,o=pki
```

com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder

Select this subject builder to construct the Subject DN from the Subject Alternative Name provided in the request or CSR. Specifically, this builder:

1. Pulls out the SAN as per the order of SAN types preference provided in [SAN type order](#) property.
2. Uses the first SAN as the subject by filling the provided template. This SAN type can have only one value.
3. Gives priority to SAN from the request over the SAN provided in CSR.

This subject builder is useful when subject is not provided in both the request and CSR.

Sample SubAltNameSubjectBuilder

```
- name: "Use SubAltNameSubjectBuilder"
  unique-id: "CA-1003-PROF-1003"
  subject-variable-requirements:
    - name: SAN
      description: "Subject Alternative Name"
      required: true
  subject-builder-config:
    subject-builder-name:
      "com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder"
  properties:
    template: "cn=<SAN>,ou=CA01,o=pki,test,dc=com"
    san-type-order:
      dnsName, ipAddress, registeredID, rfc822Name, uniformResourceIdentifier
```

Sample subject variables

```
"subjectVariables" : [
  {
    "type" : "Subject Alternative Name",
    "value" : "SAN"
  }
]
```



```
],  
  "subjectAltNames" : [ {  
    "type" : "dNSName",  
    "value" : "cagw.test"  
  } ]
```

Subject DN generated by the sample builder when parsing the sample variables

```
cn=cagw.test,ou=CA01,o=pki,test,dc=com
```

com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder

Select this subject builder to replace DN (Distinguished Name) variables in a template with variables from the CSR (Certificate Signing Request).

- [Example: building the Common Name from Subject Variables](#)
- [Example: building the Common Name when no Subject Variables are provided](#)

Example: building the Common Name from Subject Variables

To build the final DN, the following template expects an enrollment request with subject variables for "First Name" and "Last Name".

```
- name: "Use TemplateSubjectBuilder"  
  unique-id: "CA-1003-PROF-1002"  
  subject-variable-requirements:  
    - name: First Name  
      description: "First Name"  
      required: true  
    - name: Last Name  
      description: "Last Name"  
      required: true  
  subject-builder-config:  
    subject-builder-name:  
      "com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"  
    properties:  
      template: "cn=<First Name> <Last Name>, ou=CA01,o=pki,dc=test,dc=com"
```

For example, when receiving the following request values.

```
"subjectVariables" : [  
  {  
    "type" : "First Name",  
    "value" : "PKI"  
  },  
  {  
    "type" : "Last Name",  
    "value" : "Test"  
  }  
]
```

```
]
```

The template builds the following Distinguished Name.

```
cn=PKI Test,ou=CA01,o=pki,dc=test,dc=com
```

Example: building the Common Name when no Subject Variables are provided

To build the final DN when no Subject Variables are provided, the following template parses the CSR for common name.

 This configuration will not process fields other than CN and UID.

```
- name: "Use TemplateSubjectBuilder"
  unique-id: "CA-1003-PROF-1002"
  subject-builder-config:
    subject-builder-name:
      "com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
    properties:
      template: "cn=<cn>,ou=CA01,o=pki,dc=test,dc=com"
```

Parsing a CSR with multiple common names requires indexing the template output, starting with `cn.1`. For example;

```
template: "cn=<cn.1>, cn=<cn.2>, cn=<cn.3>, ou=CA01,o=pki,dc=test,dc=com"
```

 The use of `<CN>` or `<cn>` should be consistent.

Properties

Under this section, configure the following Subject Builder properties.

- [Template](#)
- [SAN type order](#)

Template

The DN template to use for constructing the subject. For example:

```
subject-builder-config:
  subject-builder-name:
    "com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
  properties:
    template: "cn=<CN>,ou=CA01,o=pki,dc=hooli,dc=com"
```

Mandatory: When the value of [Name](#) is `com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder`.

SAN type order

The SAN types to be used as the subject, in order of preference. Supported SAN types are:

- `dNSName`
- `iPAddress`
- `registeredID`
- `rfc822Name`
- `uniformResourceIdentifier`

Mandatory: When the value of `Name` is `com.entrust.adminservices.cagw.common.subjects.SubAltNameSubjectBuilder`.

SAN Requirements

Under this section, define the requirements of the Subject Alternative Name (`SubjectAltName`) expected during enrollment requests.

- `type`
- `required`

type

The type for the Subject Alternative Name, as defined by [RFC 5280](#). For example:

- `rfc822Name`
- `dNSName`

Mandatory: Yes.

required

`true` if the Subject Alternative Name is required, `false` if the Subject Alternative Name is optional.

Mandatory: Yes.

Minimum keysize

Configure the minimum sizes supported by the keys.

- [Specify whether to disable, log or reject key size violations](#)
- [Minimum RSA Keysize](#)
- [Minimum ECC Keysize](#)

Specify whether to disable, log or reject key size violations

The performed action when a key does not meet the [Minimum RSA Keysize](#) or [Minimum ECC Keysize](#) size requirements.

Value	Action
off	Nothing
log	Log the key size
block	Reject the key

Minimum RSA Keysize

The minimum size for RSA keys, as a number of bytes.

Minimum ECC Keysize

The minimum size for ECC (elliptic curve) keys, as a number of bytes.

ECS Profile Properties

Under this section, add the following profile settings for each ECS (Entrust Certificate Services) authority.

- [Certificate Type](#)
- [Certificate lifetime](#)
- [Client ID](#)

Certificate Type

The certificate types supported by ECS. For example:

```
STANDARD_SSL, ADVANTAGE_SSL, EV_SSL, UC_SSL, QWAC_SSL, PSD2_SSL, WILDCARD_SSL, SMIME_ENT
```

Mandatory: Yes.

Certificate lifetime

The certificate validity period in ISO 8601 format:

```
P<y>Y<m>M<d>D
```

For example, `P1Y6M10D` means one year, six months, and ten days. Certificate types such as `SMIME_ENT` restrict allowed values.

Mandatory: Yes.

Client ID

The identifier of the client requesting the certificates.

Mandatory: No. By default, the ECS REST API sets this value to 1.

Microsoft CA Profile Properties

Under this section, add the following profile settings for each Microsoft CA authority.

- [Certificate Template](#)
- [RA Enroll Key Store Provider Config](#)
- [RA Enroll Key Store Provider](#)
- [RA Enroll Key Store](#)
- [RA Enroll Key Store Type](#)
- [RA Enroll Key Store Password](#)
- [RA Enroll Key Alias](#)
- [RA Enroll Key Password](#)
- [Client Key Generation mode](#)

Certificate Template

The Microsoft Certificate name. No spaces.

Mandatory: Yes.

RA Enroll Key Store Provider Config

The SunPKCS11 configuration file described in the Thales Luna integration guide. Click **Select Files** to import this file.

Mandatory: Yes.

RA Enroll Key Store Provider

The security provider of the key store.

- [Supported security providers for credentials in Key Store file](#)
- [Supported security providers for credentials in PKCS#11 HSM](#)

CA Gateway tries loading the key store with any available security provider when this value is omitted or incorrect.

Mandatory: Yes.

Supported security providers for credentials in Key Store file

When creating RA enrollment agent credentials in a Key Store file, supported values are the following.

Value	Security provider
SunJSSE	PKCS#12 and PFX
SUN	JKS
SunJCE	JCEKS

Supported security providers for credentials in PKCS#11 HSM

When creating RA enrollment agent credentials in PKCS#11 HSM, supported values are the following.

Value	Security provider
SunPKCS11	nCipher
LunaProvider	Luna

RA Enroll Key Store

The file generated when creating RA enrollment agent credentials in a Key Store file. Supported extensions for this file are:

- p12
- pfx
- Jks

- jceks

Click **Select Files** to import this file.

Mandatory: Yes.

RA Enroll Key Store Type

The type of key store. Supported values are:


- pkcs12
- pfx
- Jks
- jceks

Mandatory: Yes.

RA Enroll Key Store Password

The password of the key store containing the enrollment agent credential. Where the key store is either:

- A key store file.
- An HSM slot.

 We recommend creating the enrollment agent credentials in a PKCS#11 HSM.

Mandatory: Yes.

RA Enroll Key Alias

The alias for accessing the enrollment agent's key in either:

- A key store file.
- An HSM slot. In this case, you can usually omit this value because most HSMs do not protect the slot objects with an additional password.

Mandatory: Yes.

RA Enroll Key Password

The password for accessing the enrollment agent's key in either:

- A key store file.
- An HSM slot. In this case, you can usually omit this value because most HSMs do not protect the slot objects with an additional password.

Mandatory: Yes.

Client Key Generation mode

The client key generation mode.

Value	Key generation mode
true	The client generates the key and provides a CSR for CA Gateway to return an X.509 certificate.

Value	Key generation mode
false	CA Gateway returns a PKCS#12 containing the client's key and certificate.

Mandatory: No. This optional parameter defaults to `true`.

Security Manager Profile Properties

Under this section, add the following profile settings for each Entrust Security Manager authority.

- [Certificate Type \(sm\)](#)
- [Certificate Definition](#)
- [LDAP entry creation mode](#)
- [LDAP directory mode](#)
- [User Role](#)
- [User Type](#)

Certificate Type

The Security Manager certificate type to use when processing an enrollment request under the certificate profile. For example:

- ent_twokeypair
- ent_default

The administrator EPF for the Managed CA must have permission to administer this certificate type.

Mandatory: Yes.

Certificate Definition

The certificate definition for processing enrollment requests under the certificate profile. For example:

- Verification
- Dual usage
- Encryption

This certificate definition must have an assigned certificate definition policy. Otherwise, enrollments will fail.

Mandatory: Yes.

LDAP entry creation mode

The LDAP entry creation mode.

Value	Action
true	CA Gateway will create the LDAP entry for the user. CA Gateway will connect to the directory using the LDAP credentials specified for the Managed CA.
false	The Security Manager will create an LDAP entry for the user depending on the <code>managed-cas.profiles.directory-mode</code> value.

Mandatory: No. This optional parameter defaults to `true`.

LDAP directory mode

When the [LDAP entry creation mode](#) option is disabled, this setting controls whether the Security Manager creates an LDAP entry for the user.

Value	Action
DO_OP_FAIL_IF_NOT_NEEDED	Perform the repository operation when needed, and fail if not needed.
DO_OP_SUCCEED_IF_NOT_NEEDED	Perform the repository operation when needed, and return success if not needed.
NO_OP	Omit the repository operation and do not check if the operation is needed.
NO_OP_FAIL_IF_NEEDED	Omit the repository operation, but fail if the operation is needed.

In the CA profile, certificate types as `vpn_nodir` include the following `master.certspec` advanced setting under `[Extension Definitions]`.

```
noUserInDirectory=1
```

Mandatory: Yes.

User Role


The Security Manager role for processing enrollment requests under the certificate profile (for example, "End User"). The administrator EPF for the Managed CA must have permission to administer this role.

Mandatory: No.

User Type

The Security Manager user type to use when processing an enrollment request under the certificate profile. For example:

- Person
- Web Server

 The administrator EPF for the Managed CA must have permission to administer this user type.

Mandatory: No. The user type is not required when the [LDAP entry creation mode](#) option is disabled.

Tenants

Set the following parameters in the **Tenants** tab of the **Configuration** page.

- [Tenants](#)
- [Integrators](#)

Tenants

For each tenant, set the following field values under **Tenants**.

- [Name](#)
- [Unique-Id](#)
- [Certificate Authority Id](#)

Name

A friendly name for the tenant.

Mandatory: Yes.


Unique-Id

A unique identifier for the tenant. When configuring integrators, you will specify this identifier for mapping an integrator to a tenant.

Mandatory: Yes.

Certificate Authority Id

The CA unique identifier in CA Gateway.

 Map each tenant to a different managed CA. Errors will occur if you map multiple tenants to the same managed CA.

Mandatory: Yes.

Integrators

Set the following fields under **Integrators**.

- [Name](#)
- [Unique-Id](#)
- [Tenant IDs](#)

Name

A friendly name for the integrator.

Mandatory: Yes.


Unique-Id

A unique identifier for the integrator. When creating certificates for clients, you can specify this integrator ID to map a client to an integrator.

Mandatory: Yes.

Tenant IDs

One or more of the tenant identifiers defined when configuring tenants. This setting maps an integrator to one or more tenants.

 Do not map multiple integrators to the same tenant. Errors will occur if you map more than one integrator to the same tenant.

Mandatory: Yes.

Clients

For each client, set the following field values in the **Clients** tab of the **Configuration** page.

- [Subject DN](#)
- [Tenant ID](#)
- [Integrator ID](#)
- [Role](#)

Subject DN


The subject DN of the client.

 You must issue the client a digital certificate with this subject DN.

Mandatory: Yes.

Tenant ID


One of the tenant identifiers listed under [Tenants](#)

 This value is mapped with the client and is mutually exclusive with [Integrator ID](#).

Mandatory: Yes.

Integrator ID

The integrator identifier.

 This value is mapped with the client and is mutually exclusive with [Tenant ID](#).

Mandatory: Yes.

Role

One of the following roles.

Role identifier	Role main permissions	Granted by default
integrator	Access to multiple CAs. For example, as an organization providing services or capabilities to customers, such as Identity Management service providers like Microsoft Intune.	Default role for clients mapped to an integrator.
policy-constrained-tenant	View a single CA. For example, as a consumer of the services provided by the Integrator.	Default role for clients mapped to a tenant.

Role identifier	Role main permissions	Granted by default
policy-override-tenant	Control the naming information in the certificates requested to a Security Manager CA. The CA policy of the requested certificate profile determines all other certificate content.	—
read-only-integrator	Access multiple CAs and perform <code>get</code> operations on any of them.	—
read-only-tenant	Access one CA and perform <code>get</code> operations.	—

See the following table for a more detailed description of the permissions assigned to each predefined role.

Permission	integrator	policy-override-tenant	policy-constrained-tenant	read-only-integrator	read-only-tenant
Access multiple CAs	✓	Single CA only	Single Security Manager CA only	✓	Single CA only
Request explicit extensions	✓	✓	✗	✗	✗
Request private key usage period	✓	✓	✗	✗	✗
External public keys (no CSR)	✓	✓	✗	✗	✗
Override Proof of Possession	✓	✓	✗	✗	✗
Request explicit validity dates	✓	✓	Can shorten the lifetime in CSR enrollments (relative to the CA policy).	✗	✗
CSR	✓	✓	✓	✗	✗
PKCS#12	✓	✓	✓	✗	✗

Permission	integrator	policy-override-tenant	policy-constrained-tenant	read-only-integrator	read-only-tenant
Subject DN Naming Info (including subjectDn and previousSubjectDn optional parameters)	✓	✓	✓	✗	✗
Subject Alternative Names	✓	✓	✓	✗	✗
Manage certificates (revoke, suspend, unsuspend)	✓	✓	✓	✗	✗
Search in the certificate inventory	✓	✓	✓	✓	✓
Certificate events	✓	✓	✓	✓	✓

Authorized users can request certificates with the following contents.

- Certificate Lifetimes.
- Certificate naming information: Subject DN (subject to CA DIT constraints), Subject Alternative Names.
- Key Usage
- Private Key Usage Percentage
- Required Certificate Extensions

No client role can request the following extensions from a Security Manager CA.

- authorityKeyIdentifier (2.5.29.35)
- basicConstraints (2.5.29.19)
- cRLDistributionPoints (2.5.29.31)
- cRLNumber (2.5.29.20)
- entrustVersInfo (1.3.0040.113533.7.65.0)
- invalidityDate (1.2.5.29.24)
- issuingDistributionPoint (2.5.29.28)
- netscapeRevocationUrl (2.16.840.1.113730.1.3)
- reasonCode (2.5.29.21)
- subjectKeyIdentifier (2.5.29.14)

✗ CA Gateway will ignore these extensions when included in a CSR sent from a client.

Each role can access any of the REST APIs. However, based on the role, the requested action is scoped to the allowed set of managed CAs.

Mandatory: No. This optional parameter defaults to the lowest privileged role.

Cmpv2

Set the following parameters in the **Cmpv2** tab of the **Configuration** page.

- [Truststore](#)
- [Alias](#)
- [Customization](#)
- [Shared Secret](#)
- [Caching of in-progress CMPv2 transactions](#)

Truststore

Configure the trust-store that contains root CA certificates for verifying CMP messages.

- [Upload file](#)
- [Password](#)
- [Type](#)

Upload file

The truststore that contains the CA certificates. Click **Select Files** to import this truststore from file.

Mandatory: Yes

Password

The password of the truststore that contains the CA certificates.

Mandatory: Yes.

Type

The type of truststore containing the CA certificates.

Type	Description
jks	Java truststore
pkcs12	PKCS #12 truststore

Mandatory: Yes.

Alias

An optional list of trusted root CA certificate aliases. For each alias, define the following settings.

- [Alias](#)
- [DN](#)

Alias

The alias assigned to the certificate when stored in the trust store

Mandatory: Yes.

DN

The DN (Distinguished Name) of the certificate.

Mandatory: Yes.

Customization

An optional list of rules for extending or modifying the specifications. Add the following setting for each custom rule.

- [Choose a key name](#)
- [Minimal RSA public key length](#)
- [Minimal elliptic curve \(EC\) public key length](#)
- [Digest algorithm](#)
- [MAC algorithm](#)
- [Signature class](#)
- [Signing algorithm](#)
- [EC public key algorithm](#)
- [Excluded test](#)

Choose a key name

Write a name for the new rule.

Mandatory: Yes.

Minimal RSA public key length

The minimal key length allowed for RSA public keys.

Mandatory: No. This optional value defaults to 2048 bits.

Minimal elliptic curve (EC) public key length

The minimal key length allowed for EC (Elliptic-curve) public keys.

Mandatory: No. This optional value defaults to 256 bits.

Digest algorithm

The list of supported one-way digest algorithms. Supported list items are:

- SHA-256
- SHA-384

Mandatory: No. When omitting this optional value, both SHA-256 and SHA-384 are supported.

MAC algorithm

The list of supported MAC (Message Authentication Code) algorithms.

Mandatory: No. When omitting this value, a default list is built from the [Digest algorithm](#) value.

Signature class

The list of supported signature algorithm classes. Supported list items are:

- rsa
- ecdsa

Mandatory: No. When omitting this optional value, both `rsa` and `ecdsa` are supported.

Signing algorithm

The list of supported signing algorithms.

Mandatory: No. When omitting this value, a default list is built from the [Digest algorithm](#) and [Signature-class](#) values.

EC public key algorithm

The list of algorithms of supported EC public keys. Supported list items are:

- secp256r1
- secp384r1

Mandatory: No. When omitting this optional value, both `secp256r1` and `secp384r1` are supported.

Excluded test

The list of specific tests to be excluded during validation of the message.

Mandatory: No.

Shared Secret

The settings for each connection between the CMP enrollment server and the potential request transmitters.

- [DN of the node sending the message](#)
- [Passcode](#)

Mandatory: Yes.

DN of the node sending the message

The subject's distinguished name of the certificate the enrollment server will use to sign the issued certificates.

 The certificate must be included in the Trust Store selected with the [Truststore](#) field

Mandatory: Yes.

Passcode

A list of request signing keys. Under this field, define each transmitter key with the following settings.

- [KID provided in message header](#)
- [Passcode](#)

Mandatory: Define at least one signing key.

KID provided in message header

The identifier of the key in the trust-store selected with the [Truststore](#) parameter.

Mandatory: Yes.

Passcode

The password for accessing the key in the trust-store selected with the [Truststore](#) parameter.

Mandatory: Yes.

Caching of in-progress CMPv2 transactions

The cache settings for the CMPv2 transactions.

- [Maximum Cache Size](#)
- [Initial Cache Capacity](#)
- [Expire After Value](#)
- [Expire After Time Unit](#)

Maximum Cache Size

The maximum number of entries supported by the API cache. Setting this parameter to 0 disables the cache -for example:

```
maximumSize: 0
initialCapacity: 0
```

Mandatory: Yes

Initial Cache Capacity

The initial number of entries in the API cache.

Mandatory: Yes.

Expire After Value

The number of minutes before removing an entry from the API cache. For example, to remove the entries after 5 minutes:

```
expireAfterAccess: true
expireAfterValue: 5
```

Mandatory: No. This optional parameter defaults to 60.

Expire After Time Unit

Time unit for the expiry period. This parameter supports the following Java TimeUnit enum constants:

- SECONDS
- MINUTES
- HOURS

Mandatory : Yes

TLS CRL-settings

Configure the following TLS authentication settings under **enable**.

- [To enable TLSV1 on EDM Deployments](#)
- [To enable TLSV1.1 on EDM Deployments](#)
- [To enable CRL Check on EDM Deployments](#)

To enable TLSV1 on EDM Deployments

`true` to enable TLS 1.0 in Entrust PKI Hub installations, `false` otherwise. For example, to support Security Manager 8.2 or earlier you must enable TLS 1.0.

Mandatory: No. This optional parameter defaults to `false`.

To enable TLSV1.1 on EDM Deployments

`true` to enable TLS 1.1 in Entrust PKI Hub installations, `false` otherwise. For example:

Mandatory: No. This optional parameter defaults to `false`.

To enable CRL Check on EDM Deployments

`true` to check CRL distribution points for certificate revocation, `false` otherwise.

Mandatory: No. This optional parameter defaults to `false`.

Issuing public trust certificates with CA Gateway

CA Gateway supports issuing certificates intended to be publicly trusted. See the following sections for how to use this feature with filter lists.

- [CA Authorization](#)
- [Certificate Transparency](#)

 In the current release, only the CA described in [Integrating a Security Manager CA](#) supports this feature.

CA Authorization

With the configured CAA filter, CA Gateway lookups CAA records for the domain and each parent domain. For example, CA Gateway performs the following lookups for `www.acme.com`.

- `www.acme.com`
- `acme.com`
- `com`

CA Gateway traverses up the tree in search of CAA records. This CAA check passes if:

- The issuer in a CAA record matches the issuer defined in the `issuer-string` filter setting.
- No CAA record defines an issuer or specifies "Any CA". In this case, the domain owner is not asserting a particular issuing CA.
- No CAA record is found. In this case, the domain owner is not asserting a particular issuing CA.

The above applies to each domain requested in the CA Gateway enrollment request. For example, domains inside the CSR, subject to the following flag if applied.

```
optionalCertificateRequestDetails/useSANFromCSR
```

Domains are requested in the separate `subjectAltNames`, or in the following fields externally to the CSR.

```
optionalCertificateRequestDetails/extensions
```

CA Gateway will check CAA records for wildcard domains under RFC8659 .

Defining Multiple DNS Servers

When defining multiple DNS servers, the DNS lookups run in parallel. The check for a domain stops when reaching the number of positive responses defined in the `dns.response-threshold` configuration parameter. Thus, this parameter provides additional assurance by forcing consultation of multiple separate DNS responders while allowing some contingency if a DNS server fails to respond quickly.

For example, when using three DNS servers, setting `dns.response-threshold` to "2" ensures at least two positive DNS checks against two distinct responders while allowing for the unavailability of one of the three responders.

DNS Infrastructure Guidance

Before using the CAA check feature of CA Gateway, read RFC8659 with particular attention to section 5 covering security considerations. This RFC provides rules and advice for CAA checking. Deploying the DNS infrastructure is the responsibility of the customer.

The DNS responders referenced in the CA Gateway configuration are under the CA and CA Gateway responsibility (not under the control of a third-party cache such as Google or CloudFlare). All records received by CA Gateway come from authoritative nameservers. Caching of these records at the responder is allowed.

DNSSEC

As stated by RFC8659 , DNSSEC allows CA Gateway to ensure that an empty resource record (potentially containing the domain owner's stated issuer) is legitimately empty or not empty after a record suppression. CA Gateway will validate DNSSEC if present but still proceed if no DNSSEC applies for the domain.



CA Gateway does not archive the DNSSEC proof for future audits.

Certificate Transparency

CA Gateway can collect a set of signed CT log server responses and ask the underlying CA if the certificates for public trust include these responses in an SCT List extension. The certificate transparency filter:

1. Sends parallel requests to all of the configured log servers.
2. Waits for sufficient log server responses to arrive. In the filter configuration, a certificate transparency policy states the type and the minimum of required responses.
3. Requests the final certificate to the CA.

This approach allows defining a surplus of log servers to guard against slow or offline servers.

Administrating CA Gateway

Once deployed, you can administrate CA Gateway as explained below.

- [Checking CA Gateway error codes](#)
- [Checking the CA Gateway health](#)
- [Checking the health of a CA](#)

Checking CA Gateway error codes

For a description of each error code recorded in the CA Gateway logs, see the CA Gateway API documentation at:

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/docs
```

Where `<HOST>` and `<CAGW_HOST_PORT>` are the hostname and port of the CA Gateway service. CA Gateway logs can also include the following warning message.

```
Version <= 1.4 profile configuration detected for CA <ca>. This configuration syntax is deprecated. Please update.
```

CA Gateway records this warning message when the YAML configuration includes a deprecated profile syntax under:

```
cagw.authorities.managed-cas.<CA>
```

To avoid this message, configure the CA profiles as described in the configuration guide.

Checking the CA Gateway health

As explained in the [Other CA Gateway endpoints](#), CA Gateway provides the following endpoints to check the health of the CA Gateway server.

- [health](#)
- [health/{group}/diskSpace](#)
- [health/{group}/ping](#)
- [prometheus](#)

Checking the health of a CA

As explained in the [Other CA Gateway endpoints](#), CA Gateway provides the `v1/certificate-authorities/{caId}/status` endpoint to check the health of a Certificate Authority.

CA Gateway health endpoints

To enable health endpoints, you must configure the following parameter in the `application.yml` file.

```
management.endpoints.web.exposure.include
```

CA Gateway will expose the following endpoints to check the CA application health.

```
https://<HOST>:<MONITOR_HOST_PORT>/<management.endpoints.web.base-path>/<ENDPOINT>
```

Where:

- `<HOST>` is the hostname or IP address of the CA Gateway host server.
- `<MONITOR_HOST_PORT>` is the 9444 port (not configurable in Entrust PKI Hub installations).
- `<management.endpoints.web.base-path>` is the value of the `management.endpoints.web.base-path` parameter in the `application.yml` configuration file.
- `<ENDPOINT>` is the identifier of one of the endpoints described below.
 - [health](#)
 - [health/{group}/diskSpace](#)

- [health/{group}/ping](#)
- [prometheus](#)

health

The following endpoint returns information on the CA Gateway server health.

```
https://<HOST>:<MONITOR_HOST_PORT>/<management.endpoints.web.base-path>/health
```

For example:

```
{"status":"UP","groups":["custom"]}
```

See below for a description of each value.

- [status](#)
- [groups](#)

status

The ping status of the CA Gateway server.

groups

The list of user groups configured in the CA Gateway server.

health/{group}/diskSpace

The following endpoint returns the disk space of the CA Gateway server for a group.

```
https://<HOST>:<MONITOR_HOST_PORT>/<management.endpoints.web.base-path>/health/{group}/diskSpace
```

Where `group` is one of the groups listed by the [health](#) endpoint. For example, to check the disk space for the `custom` group.

```
https://localhost:9444/cagw/management/actuator/health/custom/diskSpace
```

If the server is up, this endpoint will return a response like the following.

```
{"status":"UP","details":  
{"total":1013309239296,"free":765931622400,"threshold":10485760,"exists":true}}
```

health/{group}/ping

The following endpoint returns the ping status of the CA Gateway server for a group.

```
https://<HOST>:<MONITOR_HOST_PORT>/<management.endpoints.web.base-path>/health/  
{group}/ping
```

Where `group` is one of the groups listed by the `health` endpoint. For example, to check the ping status for the `custom` group.

```
https://localhost:9444/cagw/management/actuator/health/custom/ping
```

If the server is up, this endpoint will return the following response.

```
{"status":"UP"}
```

prometheus

The following endpoint returns CA Gateway metrics in Prometheus-compliant format.

```
https://<HOST>:<MONITOR_HOST_PORT>/<management.endpoints.web.base-path>/prometheus
```

For example:

```
# HELP jvm_threads_live_threads The current number of live threads including both  
daemon and non-daemon threads  
# TYPE jvm_threads_live_threads gauge  
jvm_threads_live_threads 51.0  
# HELP spring_security_filterchains_AnonymousAuthenticationFilter_before_total  
# TYPE spring_security_filterchains_AnonymousAuthenticationFilter_before_total  
counter  
spring_security_filterchains_AnonymousAuthenticationFilter_before_total{security_secu  
rity_reached_filter_section="before",spring_security_filterchain_position="0",spring_  
security_filterchain_size="0",} 8.0  
# HELP jvm_gc_live_data_size_bytes Size of long-lived heap memory pool after  
reclamation  
# TYPE jvm_gc_live_data_size_bytes gauge  
jvm_gc_live_data_size_bytes 8.7626752E7  
# HELP executor_completed_tasks_total The approximate total number of tasks that have  
completed execution  
# TYPE executor_completed_tasks_total counter  
executor_completed_tasks_total{name="applicationTaskExecutor",} 0.0  
executor_completed_tasks_total{name="taskScheduler",} 2.0  
# HELP system_cpu_count The number of processors available to the Java virtual  
machine  
# TYPE system_cpu_count gauge  
system_cpu_count 8.0
```

Other CA Gateway endpoints

In addition to the [CA Gateway health endpoints](#), CA Gateway provides the following endpoints.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/<ENDPOINT>
```

Where:

- `<HOST>` is the hostname or IP address of the CA Gateway host server.
- `<CAGW_HOST_PORT>` is the 8444 port (not configurable in Entrust PKI Hub installations).
- `<server.servlet.context-path>` is the value of the `server.servlet.context-path` parameter in the `application.yml` configuration file.
- `<ENDPOINT>` is the identifier of one of the endpoints described below.
 - [docs](#)
 - [swagger-ui](#)
 - [v1](#)
 - [v1/certificate-authorities/{cald}/properties](#)
 - [v1/certificate-authorities/{cald}/status](#)

docs


The following endpoint provides documentation on using the CA Gateway API for certificate policy, certificate issuance, and certificate lifecycle management.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/doc
```

swagger-ui

The following endpoint provides a Swagger UI for visualizing and interacting with the CA Gateway REST API. CA Gateway includes this UI to assist developers in API integrations.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/swagger-ui
```

 This UI is not for, nor will it be supported, in the production uses of CA Gateway. It is not a substitute for an administrator UI. We recommend using Entrust's Certificate Hub or an equivalent interface provided by another product.

To test CA Gateway with Swagger

1. In the `application.yml` file, configure a tenant, or an integrator.
2. Install the tenant or integrator credential in the browser.
3. Make sure that the certification chain of the CA Gateway TLS certificate is trusted.
4. Navigate to the URL of the Swagger UI. For example:

```
https://localhost:8444/cagw/swagger-ui
```

5. When prompted by the browser, select the credential of the tenant or integrator.

6. Use the Swagger options to generate curl commands. For example, the following command lists the CAs visible to the tenant or integrator.

```
curl --cert-type P12 --cert tentant.p12:mypassword -X GET "https://cid-cagw.dev.entrust.local/cagw/v1/certificate-authorities" -H "accept: application/json"
```

✘ When running curl commands, some Linux versions do not support authenticating with a P12 file.

v1

The following endpoint returns version information on CA Gateway.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/v1
```

This endpoint is the main API entry point to invoke API capabilities.

v1/certificate-authorities/{caId}/properties

The following endpoint returns property values on Entrust CAs.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/v1/certificate-authorities/{caId}/properties?fields={properties}
```

Where `{caId}` is the Security Manager CA identifier and `{properties}` is a comma-separated list of the following property identifiers:

- defaultPolicyOIDs
- encryptionPolicyOIDs
- verificationPolicyOIDs

For example, the following request checks all these properties on the Security Manager CA with the `CA3` identifier.

```
GET https://localhost:8444/cagw/v1/certificate-authorities/CA3/status?fields=defaultPolicyOIDs,encryptionPolicyOIDs,verificationPolicyOIDs
```

The response looks like the following.

```
{
  "type": "CAPropertiesResponse",
  "CAPropertiesInformation": {
    "properties": {
      "defaultPolicyOIDs": [
        "1.1.1.1",
        "2.2.2.2"
      ],
      "encryptionPolicyOIDs": [
        "1.1.1.1"
      ]
    }
  }
}
```

```
    ],  
    "verificationPolicyOIDs": [  
      "2.2.2.2"  
    ]  
  }  
}
```

v1/certificate-authorities/{caId}/status

The following endpoint returns the up or down status of a Certificate Authority.

```
https://<HOST>:<CAGW_HOST_PORT>/<server.servlet.context-path>/v1/certificate-  
authorities/{caId}/status
```

Where `{caId}` is the Certificate Authority identifier. For example, the following request checks the status of a Certificate Authority with the `CA3` identifier.

```
GET https://localhost:8444/cagw/v1/certificate-authorities/CA3/status
```

The response looks like the following.

```
{  
  "type" : "CAStatusResponse",  
  "status" : "UP",  
}
```

CA Capabilities reference

The "Get CA Capabilities" endpoint of the CA Gateway API informs on the capabilities supported by each type of CA. The following sections give a complete reference of the returned values.

- [CA management capabilities](#)
- [Certificate enrollment capabilities](#)
- [Certificate management capabilities](#)
- [Certificate search capabilities](#)

CA management capabilities

The "Get CA Capabilities" endpoint returns the following values for each CA management capability.

- [CAStatus](#)
- [PermitsDefaultCA](#)
- [SupportsMultipleCAs](#)

CAStatus

Check whether the CA is up or down.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	False

PermitsDefaultCA

The CA can be set as the default CA of CA Gateway.

CA	Returned value
Entrust Security Manager	False
ECS	False
Microsoft ADCS	False

SupportsMultipleCAs

The CA type supports multiple CAs of the same type.

CA	Returned value
Entrust Security Manager	False
ECS	False
Microsoft ADCS	False

Certificate enrollment capabilities

The "Get CA Capabilities" endpoint returns the following values for each enrollment capability.

- [CAGeneratedKey](#)
- [CAGeneratedKeyBackup](#)
- [ClientGeneratedKeyBackup](#)
- [EnrollmentByCSR](#)
- [ExtensionInCSR](#)
- [ExtensionInRequest](#)
- [KeyInRequest](#)
- [PKCS12Response](#)
- [SANInCSR](#)

- [SANInRequest](#)
- [SubjectNameInRequest](#)
- [ValidateProofOfPossession](#)
- [ValidityPeriodInRequest](#)
- [X509CertificateResponse](#)

CAGeneratedKey

Generate the key in the server and return the generated key in PKCS#12 form). Individual profiles within the CA can disable this capability.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

CAGeneratedKeyBackup

Back up the server-generated key.

CA	Returned value
Entrust Security Manager	True
ECS	False
Microsoft ADCS	True

ClientGeneratedKeyBackup

Back up the key provided by the client during the request.

CA	Returned value
Entrust Security Manager	True
ECS	False
Microsoft ADCS	True

EnrollmentByCSR

Support certificate signing requests.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

ExtensionInCSR

Process the extension request in the CSR.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

ExtensionInRequest

Process the extension request in the enrollment request.

CA	Returned value
Entrust Security Manager	True
ECS	False
Microsoft ADCS	False

KeyInRequest

In the enrollment request, the client can add a key for the enrollment.

CA	Returned value
Entrust Security Manager	True
ECS	False

CA	Returned value
Microsoft ADCS	False

PKCS12Response

Return certificates and keys in PKCS#12 form.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

SANInCSR

Process the Subject Alternative Names in the CSR.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

SANInRequest

Process Subject Alternative Names in the enrollment request.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	False

SubjectNameInRequest

Use Subject Name parameters of the CSR to construct the subject's DN of the supplied order.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	False

ValidateProofOfPossession

Validate the proof of possession.

CA	Returned value
Entrust Security Manager	True
ECS	False
Microsoft ADCS	False

ValidityPeriodInRequest

Requests can supply a validity period.

CA	Returned value
Entrust Security Manager	True
ECS	False
Microsoft ADCS	False

X509CertificateResponse

Return certificates in X509 form.

CA	Returned value
Entrust Security Manager	True
ECS	True

CA	Returned value
Microsoft ADCS	True

Certificate management capabilities

The "Get CA Capabilities" endpoint returns the following values for each certificate management capability.

- [CertificateAction](#)
- [RevokeAction](#)
- [CertificateEvents](#)
- [Recover](#)
- [SubjectDNAction](#)

CertificateAction

List the lifecycle management actions supported by the issued certificates.

Action	SM	ECS	MS ADCS
HoldAction	✓		✓
UnholdAction	✓		✓
RevokeAction	✓	✓	✓
DeactivateAction		✓	
RenewAction		✓	
ReissueAction		✓	

RevokeAction

List the revocation reasons supported by the certificates.

Reason	SM	ECS	MS ADCS
unspecified	✓	✓	✓
keyCompromise	✓	✓	✓
cACompromise			✓

Reason	SM	ECS	MS ADCS
affiliationChanged	✓	✓	✓
superseded	✓	✓	✓
cessationOfOperation	✓	✓	✓
certificateHold	✓		✓
removeFromCRL (Unholds a certificate previously revoked with the certificateHold reason)			
privilegeWithdrawn			
cACompromise			

CertificateEvents

States if the CA supports the Certificates Events API.

CA	Returned value
Entrust Security Manager below 8.3.30	False
Entrust Security Manager 8.3.30 and above	True
ECS	True
Microsoft ADCS	True

Recover

States if the CA can recover certificates by DN.

CA	Returned value
Entrust Security Manager	Recover all certificates, recover the latest certificates.
ECS	True

CA	Returned value
Microsoft ADCS	Recover all certificates, recover the latest certificates.

SubjectDNAction

List the certificate actions by the subject's DN.

Actions	SM	ECS	MS ADCS
HoldAction	✓		✓
UnholdAction	✓		✓
RevokeAction	✓		✓
DeactivateAction	✓		
ReactivateAction	✓		

Certificate search capabilities

The "Get CA Capabilities" endpoint returns the following values for each search capability.

- [SearchBySerial](#)
- [SearchBySubjectDN](#)

SearchBySerial

Lookup certificates by serial number.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

SearchBySubjectDN

Lookup certificates by the subject's DN.

CA	Returned value
Entrust Security Manager	True
ECS	True
Microsoft ADCS	True

Starting up Certificate Enrollment Gateway

See below for starting up the Certificate Enrollment Gateway solution.


- [Certificate Enrollment Gateway overview](#)
- [Preparing to deploy Certificate Enrollment Gateway](#)
- [Issuing TLS certificates for Certificate Enrollment Gateway](#)
- [Configuring and deploying Certificate Enrollment Gateway](#)
- [Enrollment URLs for Certificate Enrollment Gateway](#)
- [Integrating Certificate Enrollment Gateway](#)

See [Browsing logs with Grafana](#) for how to browse Certificate Enrollment Gateway logs.

Certificate Enrollment Gateway overview

Entrust Certificate Enrollment Gateway provides automated certificate enrollment and renewal for the following protocols:

- WSTEP
- ACMEv2
- Intune-SCEP
- SCEP
- MDMWS
- MDM-SCEP

 Certificate Enrollment Gateway does not archive or back up private keys for data decryption.

The following topics provide more information about Certificate Enrollment Gateway:

- [Certificate Enrollment Gateway architecture](#)
- [Entrust PKI as a Service certificate profiles](#)

Certificate Enrollment Gateway architecture

The following diagram illustrates the Certificate Enrollment Gateway architecture.



The following topics describe each component of the architecture.

- [Enrollment endpoint](#)
- [Certificate Enrollment Gateway service](#)
- [Certificate issuer](#)

Enrollment endpoint

An enrollment endpoint is a user or device that requests a certificate issuance or renewal.

Certificate Enrollment Gateway service

The Certificate Enrollment Gateway service runs in Entrust PKI Hub 1.0. This microservices-based cluster provides:

- Easy install and uninstall.
- Centralized logging.
- Reporting and operational dashboards.

The Certificate Enrollment Gateway service supports an HTTP and HTTPS proxy for outbound connections.

Certificate issuer

A Certificate Issuer is a Certificate Authority (CA) that issues certificates to the enrollment endpoints.

Entrust PKI as a Service certificate profiles

For online architectures, Entrust PKI as a Service provides the certificate profiles for Certificate Enrollment Gateway. The following topics briefly describe the certificate profiles defined in Entrust PKI as a Service for Certificate Enrollment Gateway.

- [ACMEv2 certificate profiles in Entrust PKI as a Service](#)
- [Intune-SCEP certificate profiles in Entrust PKI as a Service](#)
- [MDM-SCEP certificate profiles in Entrust PKI as a Service](#)
- [MDMWS certificate profiles in Entrust PKI as a Service](#)
- [SCEP certificate profiles in Entrust PKI as a Service](#)
- [WSTEP certificate profiles in Entrust PKI as a Service](#)

ACMEv2 certificate profiles in Entrust PKI as a Service

Entrust PKI as a Service provides the following digital signature and key encipherment profiles for ACMEv2 enrollment.

Profile	Usages
privatessl-tls-client-server	Digital signature and key encipherment with the 1.3.6.1.5.5.7.3.1 extended key usage extension for TLS server authentication and 1.3.6.1.5.5.7.3.2 for TLS client authentication.
privatessl-tls-server	Digital signature and key encipherment with the 1.3.6.1.5.5.7.3.1 extended key usage extension for TLS server authentication.
privatessl-tls-client	Digital signature and key encipherment with the 1.3.6.1.5.5.7.3.2 extended key usage extension for TLS client authentication.

Unless specified in an ACMEv2 request, each of these profiles has a 3-year duration. These ACMEv2 certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ApplicationPolicies	1.3.6.1.4.1.311.21.10

Intune-SCEP certificate profiles in Entrust PKI as a Service

Entrust PKI as a Service provides the following certificate profiles for Intune-SCEP enrollment with Certificate Enrollment Gateway.

Profile	Usages
intune-digital-signature-key-encipherment	Digital signature and key encipherment.
intune-digital-signature	Digital signature.
intune-key-encipherment	Key encipherment.
intune-non-repudiation	Digital signature and non repudiation.

Unless specified in an Intune request, these SCEP certificate profiles have a 3-year duration. These Intune-SCEP certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ExtendedKeyUsage	2.5.29.37
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2

MDM-SCEP certificate profiles in Entrust PKI as a Service


Entrust PKI as a Service provides the following certificate profiles for MDM-SCEP enrollment with Certificate Enrollment Gateway.

Profile	Usages
mdmws-digital-signature-key-encipherment	Digital signature and key encipherment.
mdmws-digital-signature	Digital signature.
mdmws-key-encipherment	Key encipherment.
mdmws-non-repudiation	Digital signature and non repudiation.

Unless specified in an MDM-SCEP request, these MDM-SCEP certificate profiles have a 3-year duration. These MDM-SCEP certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ExtendedKeyUsage	2.5.29.37
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2

MDMWS certificate profiles in Entrust PKI as a Service

 The MDMWS certificate profiles will be available in Entrust Certificate Services release 13.4.

Entrust PKI as a Service provides the following certificate profiles for MDM Web Service (MDMWS) enrollment with Certificate Enrollment Gateway.

Profile	Usages
mdmws-p12-digital-signature-key-encipherment	Digital signature and key encipherment, with PKCS #12 generation enabled (RSA-2048 key).
mdmws-p12-digital-signature	Digital signature, with PKCS #12 generation enabled (RSA-2048 key).

Profile	Usages
mdmws-p12-key-encipherment	Key encipherment, with PKCS #12 generation enabled (RSA-2048 key).
mdmws-p12-non-repudiation	Digital signature and non repudiation, with PKCS #12 generation enabled (RSA-2048 key).

Unless specified in an MDMWS request, these MDMWS certificate profiles have a 3-year duration. These MDMWS certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ExtendedKeyUsage	2.5.29.37
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2

SCEP certificate profiles in Entrust PKI as a Service

Entrust PKI as a Service provides the following certificate profiles for SCEP enrollment with Certificate Enrollment Gateway.

Profile	Usages
scep-digital-signature-key-encipherment	Digital signature and key encipherment.
scep-digital-signature	Digital signature.
scep-key-encipherment	Key encipherment.
scep-non-repudiation	Digital signature and non repudiation.

Unless specified in a request, these SCEP certificate profiles have a 3-year duration. These SCEP certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ExtendedKeyUsage	2.5.29.37
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2

WSTEP certificate profiles in Entrust PKI as a Service

Entrust PKI as a Service provides the following certificate profiles for WSTEP enrollment with Certificate Enrollment Gateway.

Profile	Usages
wstep-digital-signature-key-encipherment	Digital signature and key encipherment
wstep-digital-signature	Digital signature
wstep-key-encipherment	Key encipherment
wstep-non-repudiation	Digital signature and non repudiation

The validity period for these WSTEP certificate profiles will match what is defined in the Windows Certificate Template, with a maximum validity period of 3 years. These WSTEP certificate profiles support the following extensions in the certificate requests.

Certificate request extension	OID
CertificatePolicies	2.5.29.32
ExtendedKeyUsage	2.5.29.37
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15

Certificate request extension	OID
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2


Preparing to deploy Certificate Enrollment Gateway

This section describes how to prepare for new deployments of Certificate Enrollment Gateway.

- [Verifying port access for Certificate Enrollment Gateway](#)
- [Configuring an on-premises Security Manager CA for Certificate Enrollment Gateway](#)
- [Deploying Entrust CA Gateway for an on-premises CA](#)

Verifying port access for Certificate Enrollment Gateway

In addition to the ports listed in [Required open ports](#), ensure no network restriction blocks access to port 1443.

 Certificate Enrollment Gateway deployment automatically opens this port in the firewall of the machines hosting Entrust PKI Hub.

Configuring an on-premises Security Manager CA for Certificate Enrollment Gateway

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Configuring an on-premises Security Manager CA for ACMEv2 enrollment](#)
- [Configuring an on-premises Security Manager CA for MDM-SCEP enrollment](#)
- [Configuring an on-premises Security Manager CA for MDMWS enrollment](#)
- [Configuring an on-premises Security Manager CA for SCEP or Intune-SCEP enrollment](#)
- [Configuring an on-premises Security Manager CA for WSTEP enrollment](#)

Configuring an on-premises Security Manager CA for ACMEv2 enrollment

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA for ACMEv2 enrollment, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Adding certificate types to Security Manager for ACMEv2 enrollment](#)
- [Mapping certificate definition policies to the ACMEv2 certificate types](#)

Adding certificate types to Security Manager for ACMEv2 enrollment

For ACMEv2 enrollment, you must add the following certificate types to the Security Manager CA: ACME V2 TLS Client, ACME V2 TLS Server, and ACME V2 TLS Client and Server.

To add ACMEv2 certificate types to Security Manager

1. Log in to Security Manager Administration.
2. Export the certificate specifications to a file by selecting **File > Certificate Specifications > Export**.
3. Open the certificate specifications file in a text editor.

4. Add the following lines to the `[Certificate Types]` section.

```
; -----  
; Certificate types to be used with ACME  
; -----  
acme_tls_client=enterprise,ACME V2 TLS Client,ACME V2 TLS Client Certificate  
acme_tls_server=enterprise,ACME V2 TLS Server,ACME V2 TLS Server Certificate  
acme_tls_client_srv=enterprise,ACME V2 TLS Client and Server,ACME V2 TLS Client  
and Server Certificate
```

5. Add the following lines to the `[Extension Definitions]` section.

```
; -----  
; Certificate definitions to be used with ACME Public protocol in CEG  
; -----  
  
[acme_tls_client Certificate Definitions]  
1=Dual Usage  
  
[acme_tls_client Dual Usage Extensions]  
; KeyUsage = DigitalSignature + KeyEncipherment  
keyusage=2.5.29.15,n,m,BitString,101  
; TLS Client Authentication  
extkeyusage=2.5.29.37,n,o,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.2  
  
[acme_tls_client Advanced]  
noUserInDirectory=1  
  
[acme_tls_server Certificate Definitions]  
1=Dual Usage  
  
[acme_tls_server Dual Usage Extensions]  
; KeyUsage = DigitalSignature + KeyEncipherment  
keyusage=2.5.29.15,n,m,BitString,101  
; TLS Server Authentication  
extkeyusage=2.5.29.37,n,o,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.1  
  
[acme_tls_server Advanced]  
noUserInDirectory=1  
  
[acme_tls_client_srv Certificate Definitions]  
1=Dual Usage  
  
[acme_tls_client_srv Dual Usage Extensions]  
; KeyUsage = DigitalSignature + KeyEncipherment  
keyusage=2.5.29.15,n,m,BitString,101  
; TLS Server Authentication + TLS Client Authentication  
extkeyusage=2.5.29.37,n,o,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.1 1.3.6.  
_continue_=1.5.5.7.3.2
```



```
[acme_tls_client_srv Advanced]
noUserInDirectory=1

; --- END ACME Certificate Definitions -----
```

6. Save and close the file.
7. Import the certificate specifications back into Security Manager. In Security Manager Administration, select **File > Certificate Specifications > Import**.

Mapping certificate definition policies to the ACMEv2 certificate types

The ACMEv2 certificate types you added to Security Manager have certificate definitions. You must map certificate definition policies to these certificate definitions as described in the following procedures.

To map a certificate definition policy to the ACME V2 TLS Client certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > ACME V2 TLS Client > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the ACME V2 TLS Server certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > ACME V2 TLS Server > Verification_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the ACME V2 TLS Client and Server certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > ACME V2 TLS Client and Server > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

Configuring an on-premises Security Manager CA for MDM-SCEP enrollment

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA for MDM-SCEP enrollment, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Configuring Security Manager to allow server-generated keys for MDM-SCEP enrollment](#)
- [Adding certificate types to Security Manager for MDM-SCEP enrollment](#)
- [Mapping certificate definition policies to the MDM-SCEP certificate types](#)

Configuring Security Manager to allow server-generated keys for MDM-SCEP enrollment

For MDM-SCEP enrollment with Certificate Enrollment Gateway, Security Manager must allow server-generated verification and nonrepudiation keys. To allow server-generated verification and nonrepudiation keys in Security Manager, configure the following `entmgr.ini` settings:

```
[policy]
allowServerGenVerCert=true
allowServerGenNonRepudCert=true
```

For information about changing these settings, see the Security Manager documentation.

Adding certificate types to Security Manager for MDM-SCEP enrollment

For MDM-SCEP enrollment, you must add the following certificate types to the Security Manager CA: signing, encryption, dual usage (signing and encryption), non-repudiation.

To add MDM-SCEP certificate types to Security Manager

1. Log in to Security Manager Administration.
2. Export the certificate specifications to a file by selecting **File > Certificate Specifications > Export**.
3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```
; -----
; Certificate types to be used with MDM for SCEP Enrollments
; -----
ent_mdm_scep_sig=enterprise,MDM-SCEP Signing,MDM-SCEP Signing Certificate
ent_mdm_scep_enc=enterprise,MDM-SCEP Encryption,MDM-SCEP Encryption Certificate
ent_mdm_scep_sig_enc=enterprise,MDM-SCEP Signing and Encryption,MDM-SCEP
Signing and Encryption Certificate
ent_mdm_scep_nonrep=enterprise,MDM-SCEP Signing and Nonrepudiation,MDM-SCEP
Signing and Nonrepudiation Certificate
; -----
```

5. Add the following lines to the `[Extension Definitions]` section.

```
; -----
; Certificate definitions to be used with MDM for SCEP Enrollments
; -----
[ent_mdm_scep_sig Certificate Definitions]
1=Verification_p10

[ent_mdm_scep_sig Verification_p10 Extensions]
keyusage=2.5.29.15,n,m,BitString,1

[ent_mdm_scep_sig Advanced]
noUserInDirectory=1

[ent_mdm_scep_enc Certificate Definitions]
1=Encryption_p10

[ent_mdm_scep_enc Encryption_p10 Extensions]
keyusage=2.5.29.15,n,m,BitString,001

[ent_mdm_scep_enc Advanced]
```

```
noUserInDirectory=1

[ent_mdm_scep_sig_enc Certificate Definitions]
1=Dual Usage

[ent_mdm_scep_sig_enc Dual Usage Extensions]
keyusage=2.5.29.15,n,m,BitString,101

[ent_mdm_scep_sig_enc Advanced]
noUserInDirectory=1

[ent_mdm_scep_nonrep Certificate Definitions]
1=Nonrepudiation

[ent_mdm_scep_nonrep Nonrepudiation Extensions]
keyusage=2.5.29.15,n,m,BitString,11

[ent_mdm_scep_nonrep Advanced]
noUserInDirectory=1
;-----
```

6. Save and close the file.
7. Import the certificate specifications back into Security Manager. In Security Manager Administration, select **File > Certificate Specifications > Import**.

Mapping certificate definition policies to the MDM-SCEP certificate types

The MDM-SCEP certificate types you added to Security Manager have certificate definitions. You must map certificate definition policies to these certificate definitions as described in the following procedures.

To map a certificate definition policy to the MDM SCEP Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM SCEP Encryption > Encryption_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Encryption_P10 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDM SCEP Signing certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM SCEP Signing > Verification_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Verification_p10 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDM SCEP Signing and Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM SCEP Signing and Encryption > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDM SCEP Signing and Nonrepudiation certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM SCEP Signing and Nonrepudiation > Nonrepudiation**.
3. In the **Certificate definition Policy** drop-down list, select **Nonrepudiation Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

Configuring an on-premises Security Manager CA for MDMWS enrollment

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA for MDM Web Service (MDMWS) enrollment, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Configuring Security Manager to allow server-generated keys for MDMWS enrollment](#)
- [Creating a client policy and role for MDMWS P12 enrollments](#)
- [Adding certificate types to Security Manager for MDMWS P12 enrollment](#)
- [Creating certificate definition policies for MDMWS P12 certificate types](#)
- [Mapping certificate definition policies to the MDMWS P12 certificate types](#)

Configuring Security Manager to allow server-generated keys for MDMWS enrollment

For MDMWS enrollment with Certificate Enrollment Gateway, Security Manager must allow server-generated verification and nonrepudiation keys. To allow server-generated verification and nonrepudiation keys in Security Manager, configure the following `entmgr.ini` settings:

```
[policy]
allowServerGenVerCert=true
allowServerGenNonRepudCert=true
```

For information about changing these settings, see the Security Manager documentation.

Creating a client policy and role for MDMWS P12 enrollments

In Certificate Enrollment Gateway, MDM Web Service (MDMWS) enrollment enrolls a user with a PKCS #12 digital ID. To allow PKCS #12 enrollment through the MDMWS protocol, the client policy assigned to user's role must allow PKCS #12 enrollment. To allow PKCS #12 enrollment, the user's client policy must have the **Allow PKCS#12 Export** and **All Exportable** policy attributes.

The following procedures describe how to create a new client policy and role that allows PKCS #12 export.

To create a client policy in Security Manager for MDMWS P12 enrollment

1. Log in to Security Manager Administration.
2. In the tree view, select **Security Policy > User Policies > End User Policy**.
3. Select **User Policies > Selected User Policy > Copy**. The Copy User Policy dialog box appears.
4. In the **Label** field, enter `End User P12 Policy`.
5. In the **Common name** field, enter `End User P12 Policy`.
6. Under **Policy Attributes**:
 - Select **Allow PKCS#12 Export**.
 - Deselect **All exportable**.
7. Click **Apply**.
8. If prompted, authorize the operation.

To create a role in Security Manager for MDMWS P12 enrollment

1. Log in to Security Manager Administration.
2. In the tree view, select **Security Policy > Roles > End User**.
3. Select **User Policies > Selected Role > Copy**. A copy of the role appears at the bottom of the list of roles in the tree view, and the new role's properties appear in the right pane.
4. In the **Unique name** field, enter **End User P12**.
5. In the **User Policy** drop-down list, select **End User P12 Policy**.
6. Click **Apply**.
7. If prompted, authorize the operation.

In CA Gateway, the CA profile that will be used by Certificate Enrollment Gateway for MDMWS enrollment must assign this role to end users. The XAP administrator profile used to manage the CA profile must also have permissions to administer this role.

Adding certificate types to Security Manager for MDMWS P12 enrollment

For MDMWS PKCS #12 (P12) enrollment, you must add the following certificate types to the Security Manager CA: signing, encryption, dual usage (signing and encryption), non-repudiation.

To add MDMWS P12 certificate types to Security Manager

1. Log in to Security Manager Administration.
2. Export the certificate specifications to a file by selecting **File > Certificate Specifications > Export**.
3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```

; -----
; Certificate types to be used with MDM for P12 Enrollments
; -----
ent_mdm_p12_sig=enterprise,MDM P12 Signing,MDM P12 Signing Certificate
ent_mdm_p12_enc=enterprise,MDM P12 Encryption,MDM P12 Encryption Certificate
ent_mdm_p12_sig_enc=enterprise,MDM P12 Signing and Encryption,MDM P12 Signing
and Encryption Certificate
ent_mdm_p12_nonrep=enterprise,MDM P12 Signing and Nonrepudiation,MDM P12
Signing and Nonrepudiation Certificate
; -----

```

5. Add the following lines to the `[Extension Definitions]` section.

```

; -----
; Certificate definitions to be used with MDM for P12 Enrollments
; -----
[ent_mdm_p12_sig Certificate Definitions]
1=Verification

[ent_mdm_p12_sig Verification Extensions]
keyusage=2.5.29.15,n,m,BitString,1

[ent_mdm_p12_sig Advanced]
noUserInDirectory=1

```

```
[ent_mdm_p12_enc Certificate Definitions]
1=Encryption

[ent_mdm_p12_enc Encryption Extensions]
keyusage=2.5.29.15,n,m,BitString,001

[ent_mdm_p12_enc Advanced]
noUserInDirectory=1

[ent_mdm_p12_sig_enc Certificate Definitions]
1=Dual Usage

[ent_mdm_p12_sig_enc Dual Usage Extensions]
keyusage=2.5.29.15,n,m,BitString,101

[ent_mdm_p12_sig_enc Advanced]
noUserInDirectory=1

[ent_mdm_p12_nonrep Certificate Definitions]
1=Nonrepudiation

[ent_mdm_p12_nonrep Nonrepudiation Extensions]
keyusage=2.5.29.15,n,m,BitString,11

[ent_mdm_p12_nonrep Advanced]
noUserInDirectory=1
;-----
```

6. Save and close the file.
7. Import the certificate specifications back into Security Manager. In Security Manager Administration, select **File > Certificate Specifications > Import**.

Creating certificate definition policies for MDMWS P12 certificate types

For MDMWS PKCS #12 (P12) enrollment, you created certificate types in Security Manager. You must create new certificate definition policies for these certificate types. These new certificate definitions will allow server-generated keys and private key backup. You will map these certificate definition policies to the certificate definitions later.

To create a Dual Usage P12 certificate definition policy in Security Manager

1. Log in to Security Manager Administration.
2. In the tree view, select **Security Policy > User Policies > Dual Usage Policy**.
3. Select **User Policies > Selected User Policy > Copy**. The **Copy User Policy** dialog box appears.
4. In the **Label** field, enter `Dual Usage P12 Policy`.
5. In the **Common name** field, enter `Dual Usage P12 Policy`.
6. Under **Policy Attributes**:
 - Select **Back up private key**.
 - Deselect **Generate key at client**.
7. Click **Apply**.
8. If prompted, authorize the operation.

To create an Encryption P12 certificate definition policy in Security Manager

1. Log in to Security Manager Administration.

2. In the tree view, select **Security Policy > User Policies > Encryption Policy**.
3. Select **User Policies > Selected User Policy > Copy**. The **Copy User Policy** dialog box appears.
4. In the **Label** field, enter `Encryption P12 Policy`.
5. In the **Common name** field, enter `Encryption P12 Policy`.
6. Under **Policy Attributes**:
 - Select **Back up private key**.
 - Deselect **Generate key at client**.
7. Click **Apply**.
8. If prompted, authorize the operation.

To create a Verification P12 certificate definition policy in Security Manager

1. Log in to Security Manager Administration.
2. In the tree view, select **Security Policy > User Policies > Verification Policy**.
3. Select **User Policies > Selected User Policy > Copy**. The **Copy User Policy** dialog box appears.
4. In the **Label** field, enter `Verification P12 Policy`.
5. In the **Common name** field, enter `Verification P12 Policy`.
6. Under **Policy Attributes**:
 - Select **Back up private key**.
 - Deselect **Generate key at client**.
7. Click **Apply**.
8. If prompted, authorize the operation.

To create a Nonrepudiation P12 certificate definition policy in Security Manager

1. Log in to Security Manager Administration.
2. In the tree view, select **Security Policy > User Policies > Encryption Policy**.
3. Select **User Policies > Selected User Policy > Copy**. The **Copy User Policy** dialog box appears.
4. In the **Label** field, enter `Nonrepudiation P12 Policy`.
5. In the Common name field, enter `Nonrepudiation P12 Policy`.
6. Under **Policy Attributes**:
 - Select **Back up private key**.
 - Deselect **Generate key at client**.
7. Click **Apply**.
8. If prompted, authorize the operation.

Mapping certificate definition policies to the MDMWS P12 certificate types

The MDMWS P12 certificate types you added to Security Manager have certificate definitions. You must map the certificate definition policies that you created earlier to these certificate definitions as described in the following procedures.

To map a certificate definition policy to the MDMWS P12 Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM P12 Encryption > Encryption**.
3. In the **Certificate definition Policy** drop-down list, select **Encryption P12 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDMWS P12 Signing certificate type

1. Log in to Security Manager Administration.

2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM P12 Signing > Verification**.
3. In the **Certificate definition Policy** drop-down list, select **Verification P12 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDMWS P12 Signing and Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM P12 Signing and Encryption > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage P12 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the MDMWS P12 Signing and Nonrepudiation certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > MDM P12 Signing and Nonrepudiation > Nonrepudiation**.
3. In the **Certificate definition Policy** drop-down list, select **Nonrepudiation P12 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

Configuring an on-premises Security Manager CA for SCEP or Intune-SCEP enrollment

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA for SCEP or Intune-SCEP enrollment, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Adding certificate types to Security Manager for SCEP and Intune-SCEP enrollment](#)
- [Mapping certificate definition policies to the SCEP certificate types](#)

Adding certificate types to Security Manager for SCEP and Intune-SCEP enrollment

For SCEP and Intune-SCEP enrollment, you must add the following certificate types to the Security Manager CA: signing, encryption, dual usage (signing and encryption), non-repudiation.

To add SCEP certificate types to Security Manager

1. Log in to Security Manager Administration.
2. Export the certificate specifications to a file by selecting **File > Certificate Specifications > Export**.
3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```
; -----  
; Certificate types to be used with SCEP  
; -----  
ent_scep_sig=enterprise,SCEP Signing,SCEP Signing Certificate  
ent_scep_enc=enterprise,SCEP Encryption,SCEP Encryption Certificate  
ent_scep_sig_enc=enterprise,SCEP Signing and Encryption,SCEP Signing and  
Encryption Certificate  
ent_scep_sig_nonrep=enterprise,SCEP Signing and Nonrepudiation,SCEP Signing and  
Nonrepudiation Certificate  
; -----
```


5. Add the following lines to the [Extension Definitions] section.

```
-----  
; Certificate definitions to be used with SCEP  
-----  
[ent_scep_sig Certificate Definitions]  
1=Verification_p10  
  
[ent_scep_sig Verification_p10 Extensions]  
keyusage=2.5.29.15,n,m,BitString,1  
  
[ent_scep_sig Advanced]  
noUserInDirectory=1  
  
[ent_scep_enc Certificate Definitions]  
1=Encryption_p10  
  
[ent_scep_enc Encryption_p10 Extensions]  
keyusage=2.5.29.15,n,m,BitString,001  
  
[ent_scep_enc Advanced]  
noUserInDirectory=1  
  
[ent_scep_sig_enc Certificate Definitions]  
1=Dual Usage  
  
[ent_scep_sig_enc Dual Usage Extensions]  
keyusage=2.5.29.15,n,m,BitString,101  
  
[ent_scep_sig_enc Advanced]  
noUserInDirectory=1  
  
[ent_scep_sig_nonrep Certificate Definitions]  
1=Nonrepudiation  
  
[ent_scep_sig_nonrep Nonrepudiation Extensions]  
keyusage=2.5.29.15,n,m,BitString,11  
  
[ent_scep_sig_nonrep Advanced]  
noUserInDirectory=1  
-----
```

6. Save and close the file.
7. Import the certificate specifications back into Security Manager. In Security Manager Administration, select **File > Certificate Specifications > Import**.

Mapping certificate definition policies to the SCEP certificate types

The SCEP certificate types you added to Security Manager have certificate definitions. You must map certificate definition policies to these certificate definitions as described in the following procedures.

To map a certificate definition policy to the SCEP Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > SCEP Encryption > Encryption_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Encryption_P10 policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the SCEP Signing certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > SCEP Signing > Verification_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Verification_p10 Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the SCEP Signing and Encryption certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > SCEP Signing and Encryption > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the SCEP Signing and Nonrepudiation certificate type

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > SCEP Signing and Nonrepudiation > Nonrepudiation**.
3. In the **Certificate definition Policy** drop-down list, select **Nonrepudiation Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

Configuring an on-premises Security Manager CA for WSTEP enrollment

If you will use Certificate Enrollment Gateway with an on-premises Security Manager CA for WSTEP enrollment, you must configure Security Manager as described in the following sections. For detailed information about configuring Security Manager, see the Security Manager documentation.

- [Configuring certificates issued by Security Manager for WSTEP enrollment](#)
- [Adding certificate types to Security Manager for WSTEP enrollment](#)
- [Mapping certificate definition policies to the WSTEP certificate types](#)

Configuring certificates issued by Security Manager for WSTEP enrollment

When using secure LDAP (LDAPS) for WSTEP integration, all TLS certificates issued by the Security Manager must include a valid HTTP CDP (CRL Distribution Point). For information about specifying CRL distribution points in Security Manager, see the Security Manager documentation.

Adding certificate types to Security Manager for WSTEP enrollment

For WSTEP enrollment, you must add the following certificate types to the Security Manager CA: signing, encryption, dual usage (signing and encryption), non-repudiation.

To add WSTEP certificate types to Security Manager

1. Log in to Security Manager Administration.
2. Export the certificate specifications to a file by selecting **File > Certificate Specifications > Export**.

3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```
; -----  
; Certificate types to be used with WSTEP  
; -----  
ent_wstep_sig=enterprise,WSTEP Signing,WSTEP Signing Certificate  
ent_wstep_enc=enterprise,WSTEP Encryption,WSTEP Encryption Certificate  
ent_wstep_sig_enc=enterprise,WSTEP Signing and Encryption,WSTEP Signing and  
Encryption Certificate  
ent_wstep_sig_nonrep=enterprise,WSTEP Signing and Nonrepudiation,WSTEP Signing  
and Nonrepudiation Certificate  
; -----
```

5. Add the following lines to the `[Extension Definitions]` section.

```
; -----  
; Certificate definitions to be used with WSTEP  
; -----  
  
[ent_wstep_sig Certificate Definitions]  
1=Verification  
  
[ent_wstep_sig Verification Extensions]  
keyusage=2.5.29.15,n,m,BitString,1  
  
[ent_wstep_sig Advanced]  
noUserInDirectory=1  
  
[ent_wstep_enc Certificate Definitions]  
1=Encryption  
  
[ent_wstep_enc Encryption Extensions]  
keyusage=2.5.29.15,n,m,BitString,001  
  
[ent_wstep_enc Advanced]  
noUserInDirectory=1  
  
[ent_wstep_sig_enc Certificate Definitions]  
1=Dual Usage  
  
[ent_wstep_sig_enc Dual Usage Extensions]  
keyusage=2.5.29.15,n,m,BitString,101  
  
[ent_wstep_sig_enc Advanced]  
noUserInDirectory=1  
  
[ent_wstep_sig_nonrep Certificate Definitions]  
1=Nonrepudiation
```

```
[ent_wstep_sig_nonrep Nonrepudiation Extensions]
keyusage=2.5.29.15,n,m,BitString,11
```

```
[ent_wstep_sig_nonrep Advanced]
noUserInDirectory=1
```

6. Save and close the file.
7. Import the certificate specifications back into Security Manager. In Security Manager Administration, select **File > Certificate Specifications > Import**.

Mapping certificate definition policies to the WSTEP certificate types

The WSTEP certificate types you added to Security Manager have certificate definitions. You must map certificate definition policies to these certificate definitions as described in the following procedures.

To map a certificate definition policy to the WSTEP Encryption certificate type:

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > WSTEP Encryption (WSTEP Encryption Certificate) > Encryption_p10**.
3. In the **Certificate definition Policy** drop-down list, select **Encryption_P10 policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the WSTEP Signing certificate type:

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > WSTEP Signing (WSTEP Signing Certificate) > Verification_P10**.
3. In the **Certificate definition Policy** drop-down list, select **Verification_P10 policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.


To map a certificate definition policy to the WSTEP Signing and Encryption certificate type:

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > WSTEP Signing and Encryption (WSTEP Signing and Encryption Certificate) > Dual Usage**.
3. In the **Certificate definition Policy** drop-down list, select **Dual Usage Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

To map a certificate definition policy to the WSTEP Signing and Nonrepudiation certificate type:

1. Log in to Security Manager Administration.
2. In the tree view, expand **Security Policy > Certificate Categories > Enterprise > WSTEP Signing and Nonrepudiation (WSTEP Signing and Nonrepudiation Certificate) > Nonrepudiation**.
3. In the **Certificate definition Policy** drop-down list, select **Nonrepudiation Policy**.
4. Click **Apply**.
5. If prompted, authorize the operation.

Deploying Entrust CA Gateway for an on-premises CA

 For Entrust PKI as a Service deployments, CA Gateway is installed and managed by Entrust.

When deploying Certificate Enrollment Gateway with an on-premises CA, you must also deploy Entrust CA Gateway on-premises.

Entrust CA Gateway enables full certificate lifecycle management and operational management across all your Entrust-supported Certification Authorities (CAs). Each Entrust CA Gateway client can access one or several CAs. Certificate Enrollment Gateway will send certificate requests to Entrust CA Gateway. Entrust CA Gateway will forward the request to the appropriate Managed CA, and send the generated certificate back to Certificate Enrollment Gateway.

For detailed information about installing and configuring Entrust CA Gateway, see the Entrust CA Gateway documentation.

- [Issuing a client credential for Certificate Enrollment Gateway](#)
- [Generating a file containing the CA certificate chain for the CA Gateway server certificate](#)
- [Defining profiles in CA Gateway for issuing RA certificates](#)
- [Defining a profile in CA Gateway for TLS bootstrapping](#)
- [Configuring CA Gateway for ACMEv2 enrollment](#)
- [Configuring CA Gateway for MDM-SCEP enrollment](#)
- [Configuring CA Gateway for MDMWS P12 enrollment](#)
- [Configuring CA Gateway for SCEP and Intune-SCEP enrollment](#)
- [Configuring CA Gateway for WSTEP enrollment](#)

Issuing a client credential for Certificate Enrollment Gateway

i In Certificate Enrollment Gateway, the client credential is called the CA Gateway Keystore. The CA Gateway Keystore can contain multiple private keys (multiple PrivateKeyEntry entries) and certificates. You can specify the alias of the private key to use for the client credential when you configure Certificate Enrollment Gateway.

Certificate Enrollment Gateway requires a client credential issued from Entrust CA Gateway. Certificate Enrollment Gateway uses this client credential to access and authenticate to Entrust CA Gateway. The client credential must be a PKCS #12 (P12) file that contains a private key and client certificate issued by a Managed CA in Entrust CA Gateway.

To issue a client credential to Certificate Enrollment Gateway, you must configure Certificate Enrollment Gateway as a client in Entrust CA Gateway. In Entrust CA Gateway, you must assign the Certificate Enrollment Gateway client either the integrator or policy-override-tenant role.

For information about configuring clients in Entrust CA Gateway, see the Entrust CA Gateway documentation.

Generating a file containing the CA certificate chain for the CA Gateway server certificate

Certificate Enrollment Gateway requires the CA certificate chain of Entrust CA Gateway's server certificate. When connecting to Entrust CA Gateway, Certificate Enrollment Gateway will use the CA certificate chain to validate Entrust CA Gateway's server certificate.

The CA certificate chain must be stored in one of the following files:

File	Description
CA Gateway Truststore	<p>This file must be a PKCS #12 (P12) file. The file must contain at least one Trusted CA Certificate entry (TrustedCertEntry).</p> <p>You can re-use the CA Gateway Keystore if it contains the CA certificate chain.</p>

File	Description
CA Certificates File	The file must be a PEM-formatted file. The file must contain at least one PEM-formatted CA certificate. Each CA certificate must include any BEGIN CERTIFICATE and END CERTIFICATE lines if present.

To generate a CA Gateway Truststore (P12 file) using the Java keytool utility

1. Obtain the certificate chain for CA Gateway's server certificates, from the server certificate to the root CA certificate.
2. Log in to a computer that has Java installed.
3. For each certificate, enter the following command to generate the CA Gateway Truststore file and import certificates into the truststore:

```
keytool -import -alias <alias> -trustcacerts -file <cert-file> -keystore <truststore>
```

Where:

- <alias> is an alias for the certificate. Use a different alias for each certificate you will import.
- <cert-file> is the path and file name of the certificate.
- <truststore> is the path and file name of the CA Gateway Truststore file. For example, cagwtruststore.p12 . The utility will create the file if it does not exist.

For example:

```
keytool -import -alias cagw-root -trustcacerts -file /tmp/root.cer -keystore /home/user/cagwtruststore.p12
```

4. When prompted, enter a password for the truststore.

To generate a CA Certificates File (PEM file)

1. Obtain the certificate chain for CA Gateway's server certificate, from the server certificate to the root CA certificate.
 2. Open a text editor.
 3. Create a new file.
 4. Paste the contents of each CA certificate file into the new file, from the server certificate to the root CA certificate. Each CA certificate must include any BEGIN CERTIFICATE and END CERTIFICATE lines if present.
- For example:

```
-----BEGIN CERTIFICATE-----
<TLS server certificate in Base64 encoding>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Issuing CA certificate in Base64 encoding>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Root CA certificate in Base64 encoding>
-----END CERTIFICATE-----
```

5. The text file should look similar to the following:

```
-----BEGIN CERTIFICATE-----
MIIDqQYJKoZIhvcNAQcCoIIDmjCCA5YCAQExADALBgkqhkiG9w0BBwGggN...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIDejCCAmKgAwIBAgIQ8e7ock59Y21Mtcy7rGJUDANBgkqhkiG9w0BAQs...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIQ0EgRW50cnkwHhcNMjMwMjA4MTUxNzEwWhcNMzMwMjA4MTU0NzEwWjAyM...
-----END CERTIFICATE-----
```

6. Save and close the file.

Defining profiles in CA Gateway for issuing RA certificates

For all SCEP-related protocols (SCEP, MDM-SCEP, and Intune-SCEP), Certificate Enrollment Gateway uses RA certificates to sign and encrypt SCEP PKI messages. In Entrust CA Gateway, for each Managed CA that will issue certificates for all SCEP-related protocols, you must create a profile for issuing RA certificates.

All profiles used for RA certificates must allow for Dual Usage (both Digital Signature and Key Encipherment). It is recommended that you use a Dual Usage certificate type that you created earlier for a SCEP-related protocol. For example, for the SCEP and Intune-SCEP protocols, you can use the SCEP Signing and Encryption (`ent_scep_sig_enc`) certificate type you created earlier for the SCEP and Intune-SCEP protocols in [Adding certificate types to Security Manager for SCEP and Intune-SCEP enrollment](#).

When adding a profile to CA Gateway for issuing RA certificates:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.

For example:

```
- name: "SCEP RA"
  unique_id: ent_scep_ra
  properties:
    cert_type: ent_scep_sig_enc
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
```

Defining a profile in CA Gateway for TLS bootstrapping

In Certificate Enrollment Gateway, the CEG Service requires TLS certificates to operate. Certificate Enrollment Gateway includes a TLS bootstrapping feature that can request and obtain TLS certificates from a Managed CA through Entrust CA Gateway (see [Managing the TLS certificates of the Certificate Enrollment Gateway Service](#)).

Using the TLS bootstrapping feature is optional for issuing TLS certificates for the CEG Service. If desired, you can use another method or tool to issue TLS certificates for your deployment.

To use the TLS bootstrapping feature, you must define a profile in Entrust CA Gateway for the Managed CA that will issue the required TLS certificates for the CEG Service. For information about defining profiles in CA Gateway, see the CA Gateway documentation.

The profile used for TLS bootstrapping must allow for Dual Usage (both Digital Signature and Key Encipherment). It is recommended that you use a Dual Usage certificate type that you created earlier for an enrollment protocol. For example, for the SCEP protocol, you can use the SCEP Signing and Encryption (ent_scep_sig_enc) certificate type you created earlier for the SCEP and Intune-SCEP protocols in [Adding certificate types to Security Manager for SCEP and Intune-SCEP enrollment](#).

When adding the profile to CA Gateway:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.

For example:

```
- name: "CEG TLS Certificate"
  unique_id: ent_ceg_sig_enc
  properties:
    cert_type: ent_ceg_sig_enc
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
```

Configuring CA Gateway for ACMEv2 enrollment

In Entrust CA Gateway, you must create profiles for each Managed CA that will issue certificates for ACMEv2 enrollment. Each profile must issue one of the ACMEv2 certificate types you added earlier to Security Manager.

When adding these profiles to CA Gateway:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.

The following example shows multiple Managed CA profiles configured in CA Gateway for ACMEv2 enrollment, one profile for each ACMEv2 certificate type you created earlier in Security Manager.

```
- name: "ACME TLS Client"
  unique_id: acme_tls_client
  properties:
    cert_type: acme_tls_client
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
- name: "ACME TLS Server"
```



```
unique_id: acme_tls_server
properties:
  cert_type: acme_tls_server
  cert_definition: Dual Usage
  user_type: Web Server
  create_ldap_entry: false
- name: " ACME TLS Client Server"
  unique_id: acme_tls_client_srv
  properties:
    cert_type: acme_tls_client_srv
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
```

Configuring CA Gateway for MDM-SCEP enrollment

In Entrust CA Gateway, you must create profiles for each Managed CA that will issue certificates for MDM-SCEP enrollment. Each profile must issue one of the MDM-SCEP certificate types you added earlier to Security Manager.

When adding these profiles to CA Gateway:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.

The following example shows multiple Managed CA profiles configured in CA Gateway for MDM-SCEP enrollment, one profile for each MDM-SCEP certificate type you created earlier in Security Manager.

```
- name: "MDM-SCEP Verification"
  unique_id: ent_mdm_scep_sig
  properties:
    cert_type: ent_mdm_scep_sig
    cert_definition: Verification_p10
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-SCEP Encryption"
  unique_id: ent_mdm_scep_enc
  properties:
    cert_type: ent_mdm_scep_enc
    cert_definition: Encryption_p10
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-SCEP Dual Usage"
  unique_id: ent_mdm_scep_sig_enc
  properties:
    cert_type: ent_mdm_scep_sig_enc
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-SCEP Nonrepudiation"
```

```
unique_id: ent_mdm_scep_sig_enc
properties:
  cert_type: ent_mdm_scep_sig_enc
  cert_definition: Nonrepudiation
  user_type: Web Server
  create_ldap_entry: false
```

Configuring CA Gateway for MDMWS P12 enrollment

In Entrust CA Gateway, you must create profiles for each Managed CA that will issue certificate for MDM Web Service enrollment (PKCS #12 enrollment over the MDMWS protocol). Each profile must issue one of the MDMWS P12 certificate types you added earlier to Security Manager.

When adding these profiles to CA Gateway:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of `user_role` must match a role that allows PKCS #12 export. You may have created a role that allows PKCS #12 export named **End User P12**.
- The value of the `create_ldap_entry` parameter must be `false`.

The following example shows multiple Managed CA profiles configured in CA Gateway for MDMWS P12 enrollment, one profile for each MDMWS P12 certificate type you created earlier in Security Manager.

```
- name: "MDM-P12 Verification"
  unique_id: ent_mdm_p12_sig
  properties:
    cert_type: ent_mdm_p12_sig
    cert_definition: Verification
    user_role: End User P12
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-P12 Encryption"
  unique_id: ent_mdm_p12_enc
  properties:
    cert_type: ent_mdm_p12_enc
    cert_definition: Encryption
    user_role: End User P12
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-P12 Dual Usage"
  unique_id: ent_mdm_p12_sig_enc
  properties:
    cert_type: ent_mdm_p12_sig_enc
    cert_definition: Dual Usage
    user_role: End User P12
    user_type: Web Server
    create_ldap_entry: false
- name: "MDM-P12 Nonrepudiation"
```

```
unique_id: ent_mdm_p12_nonrep
properties:
  cert_type: ent_mdm_p12_nonrep
  cert_definition: Nonrepudiation
  user_role: End User P12
  user_type: Web Server
  create_ldap_entry: false
```

Configuring CA Gateway for SCEP and Intune-SCEP enrollment

In Entrust CA Gateway, you must create profiles for each Managed CA that will issue certificates for SCEP or Intune-SCEP enrollment. Each profile must issue one of the SCEP certificate types you added earlier to Security Manager.

When adding these profiles to CA Gateway:

- The `subject_builder_config` field is not supported.
- The `subject-variable-requirements` field is not supported.
- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.

The following example shows multiple Managed CA profiles configured in CA Gateway for SCEP and Intune-SCEP enrollment, one profile for each SCEP certificate type you created earlier in Security Manager.

```
- name: "SCEP Signing"
  unique_id: ent_scep_sig
  properties:
    cert_type: ent_scep_sig
    cert_definition: Verification_p10
    user_type: Web Server
    create_ldap_entry: false
- name: "SCEP Encryption"
  unique_id: ent_scep_enc
  properties:
    cert_type: ent_scep_enc
    cert_definition: Encryption_p10
    user_type: Web Server
    create_ldap_entry: false
- name: "SCEP Dual Usage"
  unique_id: ent_scep_sig_enc
  properties:
    cert_type: ent_scep_sig_enc
    cert_definition: Dual Usage
    user_type: Web Server
    create_ldap_entry: false
- name: "SCEP Nonrepudiation"
  unique_id: ent_scep_sig_nonrep
  properties:
    cert_type: ent_scep_sig_nonrep
    cert_definition: Nonrepudiation
    user_type: Web Server
    create_ldap_entry: false
```

Configuring CA Gateway for WSTEP enrollment

In Entrust CA Gateway, you must create profiles for each Managed CA that will issue certificates for WSTEP enrollment. Each profile must issue one of the WSTEP certificate types you added earlier to Security Manager.

When adding these profiles to CA Gateway:

- The values of the `cert_type` (certificate type) and `cert_definition` (certificate definition) parameters must match the values specified in Security Manager.
- The value of the `create_ldap_entry` parameter must be `false`.
- The value of the `directory_mode` parameter must be `NO_OP` (the value is case sensitive).
- The `subject-variable-requirements` field is supported when the `subject_builder_config` field is used.
- The `subject_builder_config` field is supported when Certificate Enrollment Gateway has mapped a Windows certificate template to the Profile ID. WSTEP requests to Certificate Enrollment Gateway will include Windows certificate template information. In the Certificate Enrollment Gateway `config.yml` file, the `certificate-templates` setting can map Windows certificate templates to Profile IDs in CA Gateway.
 - If the certificate template is not mapped to a Profile ID, the `subject_build_config` field is ignored.
 - If the certificate template is mapped to the Profile ID and the `subject_build_config` field exists:

- The `subject_builder_name` setting must be set to `com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder`.
- For machines, the **Subject name** in the certificate template must be **Common name** or **DNS**. For information about configuring the **Subject name** in the Windows certificate template, see the *Certificate Enrollment Gateway WSTEP Integration Guide*.
- For users, the **Subject name** in the certificate template must be **Common name**. For information about configuring the **Subject name** in the Windows certificate template, see the *Certificate Enrollment Gateway WSTEP Integration Guide*.
- If the certificate template is mapped to the Profile ID and the `subject_build_config` field is absent:
 - For machines, the subject of the issued certificate will be either `CN=<Common Name>` or `CN=<DNS name>`.
 - For users, the subject of the issued certificate will be `CN=<Common Name>`.

The following example shows multiple Managed CA profiles configured in CA Gateway for WSTEP enrollment, one profile for each WSTEP certificate type you created earlier in Security Manager.

```
- name: "WSTEP Signing"
  unique_id: ent_wstep_sig
  properties:
    cert_type: ent_wstep_sig
    cert_definition: Verification_p10
    user_type: Web Server
    create_ldap_entry: false
  subject-variable-requirements:
  - description: common name
    name: CN
    required: true
  subject_builder_config:
    subject_builder_name:
"com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
  properties:
    template: "cn=<CN>,cn=Users,dc=example,dc=com"
- name: "Encryption_p10"
  unique_id: ent_wstep_enc
  properties:
    cert_type: ent_wstep_enc
    cert_definition: Encryption_p10
    user_type: Web Server
    create_ldap_entry: false
  subject-variable-requirements:
  - description: common name
    name: CN
    required: true
  subject_builder_config:
    subject_builder_name:
"com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
  properties:
    template: "cn=<CN>,cn=Users,dc=example,dc=com"
- name: "Dual Usage"
```

```
unique_id: ent_wstep_sig_enc
properties:
  cert_type: ent_wstep_sig_enc
  cert_definition: Dual Usage
  user_type: Web Server
  create_ldap_entry: false
subject-variable-requirements:
- description: common name
  name: CN
  required: true
subject_builder_config:
  subject_builder_name:
"com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
  properties:
    template: "cn=<CN>,cn=Users,dc=example,dc=com"
- name: "Nonrepudiation"
  unique_id: ent_wstep_sig_nonrep
  properties:
    cert_type: ent_wstep_sig_nonrep
    cert_definition: Nonrepudiation
    user_type: Web Server
    create_ldap_entry: false
  subject-variable-requirements:
- description: common name
  name: CN
  required: true
  subject_builder_config:
    subject_builder_name:
"com.entrust.adminservices.cagw.common.subjects.TemplateSubjectBuilder"
  properties:
    template: "cn=<CN>,cn=Users,dc=example,dc=com"
```

Issuing TLS certificates for Certificate Enrollment Gateway

Certificate Enrollment Gateway requires a TLS certificate to secure incoming connections over HTTPS. This TLS certificate must be issued and installed into Entrust PKI Hub 1.0 before Certificate Enrollment Gateway can accept any enrollment requests over HTTPS.

You must issue the TLS certificate when deploying Certificate Enrollment Gateway for the first time. You must also renew the certificate before it expires so Certificate Enrollment Gateway can continue accepting enrollment requests.

- [Creating a CSR for the Certificate Enrollment Gateway certificate](#)
- [Issuing TLS certificates with Entrust PKI as a Service](#)
- [Issuing TLS certificates with an on-premises CA](#)
- [Building a TLS certificate chain for the Certificate Enrollment Gateway certificate](#)
- [Installing the Certificate Enrollment Gateway certificate chain into Entrust PKI Hub 1.0](#)

Creating a CSR for the Certificate Enrollment Gateway certificate

Certificate Enrollment Gateway requires a TLS certificate to secure incoming connections over HTTPS. This TLS certificate must be issued and installed into Entrust PKI Hub 1.0 before Certificate Enrollment Gateway can accept any enrollment requests over HTTPS.

The following procedure describes how to create a private key and certificate signing request (CSR) using OpenSSL. A CSR contains information that the issuing CA will use to create the certificate. You will need the private key later when installing the certificate into Entrust PKI Hub 1.0. Entrust PKI as a Service or an on-premises CA can process the CSR and issue the certificate.

To generate a private key and CSR for the Certificate Enrollment Gateway certificate using OpenSSL

1. Log in to any node in the Entrust PKI Hub 1.0 cluster as the user account that owns Entrust PKI Hub 1.0.
2. Enter the following command to check if OpenSSL is installed:

```
openssl version
```

If OpenSSL is installed, the currently-installed version of OpenSSL is displayed.

3. If OpenSSL is not installed, install OpenSSL by entering the following command:

```
sudo dnf install openssl
```

4. Enter the following command to create a CSR and private key for the Certificate Enrollment Gateway certificate:

```
openssl req -nodes -newkey rsa:2048 -keyout <private key> -out <csr> -subj "<subject>"
```

The following table describes the command parameters.

Parameter	Description
-nodes	This parameter will prevent the private key from being encrypted. Entrust PKI Hub 1.0 does not support encrypted private keys.
-newkey rsa:2048	This parameter will create a new certificate request and a new private key. The private key will be generated using RSA-2048.
-keyout <private key>	This parameter specifies a path and file name for the private key. Do not delete this file. You will need this file later to install the certificate in Entrust PKI Hub 1.0.
-out <csr>	This parameter specifies a path and file name for the CSR.
-subj "<subject>"	This parameter specifies a subject for the CSR.

For example:

```
openssl req -nodes -newkey rsa:2048 -keyout /home/user/ceg/private.key -out /home/user/ceg/csr.txt -subj "/CN=example.com"
```

Issuing TLS certificates with Entrust PKI as a Service

If you are using Certificate Enrollment Gateway with Entrust PKI as a Service, you can use native Linux tools to create a CSR (certificate signing request) for the Certificate Enrollment Gateway certificate, then process the CSR using Entrust PKI as a Service to create a certificate.

- [Processing the CSR with Entrust PKI as a Service](#)
- [Downloading the CA certificate chain from Entrust PKI as a Service](#)

Processing the CSR with Entrust PKI as a Service

After creating the certificate signing request (CSR) for the Certificate Enrollment Gateway certificate, you can submit the CSR to an Issuing CA in Entrust PKI as a Service. The Issuing CA will process the CSR and generate the certificate.

To submit the CSR to Entrust PKI as a Service and obtain the TLS certificate

1. Log in the Entrust Certificate Services interface.
2. Select **Create > PKIaaS**.
The Select **Certificate Authority** pane appears.
3. From the **Certificate Authority** drop-down list, select the CA you want to issue the TLS certificate.
4. From the **Certificate Profile** drop-down list, select the certificate profile you want to use for the TLS certificate. The certificate profile must include Digital Signature for TLS certificates.
5. Click **Next**.
The **Certificate Details** pane appears.
6. In the **Subject DN** field, enter a value for the certificate's subject DN. The value should be the DNS name of the server hosting Entrust PKI Hub 1.0. For example, `cn=example.com`.
7. For **Certificate Expiry**, provide an expiry date for TLS certificate. It is recommended that the TLS certificate be valid for 1 year or less.
8. Under **Subject Alternative Names**, add one or more DNS Name components to the Subject Alternative Name (subjectAltName) extension in the certificate. The subjectAltName extension must have a DNS Name component for each DNS name that may be used by the Entrust PKI Hub 1.0 cluster.
To add a DNS Name component the Subject Alternative Name extension:
 - a. For **SAN type**, select **DNS Name**.
 - b. In the **Value** field, enter a DNS name that may be used by the server.
 - c. Click **Add** to add the DNS Name component to the Subject Alternative Name extension.
The component is added to the list of components in the Subject Alternative Name extension
 - d. To remove a component from the Subject Alternative Name extension, click **Remove** next to the extension that you want to remove.
9. Copy the contents of the CSR you generated earlier, and paste the contents into the **Certificate Signing Request (CSR)** text box.
10. Click **Submit**.
If the certificate is generated successfully, a success message appears.
11. Click **Download the newly created certificate** to download the TLS certificate.

After processing the CSR, proceed to [Downloading the CA certificate chain from Entrust PKI as a Service](#).

Downloading the CA certificate chain from Entrust PKI as a Service

Entrust PKI Hub 1.0 requires the full TLS certificate chain for the Certificate Enrollment Gateway certificate, from the TLS certificate up to the root CA. Download all CA certificates in the CA certificate chain from Entrust PKI as a Service, from the Issuing CA to the root CA.

To download CA certificates from Entrust PKI as a Service

1. Log in the Entrust Certificate Services interface.
2. Select **Administration > PKIaaS Management**.
A list of private CAs appear.
3. For each CA in the TLS certificate chain, from the Issuing CA to the Root CA:
 - a. Select the CA.
 - b. Click **Download certificate**.

After downloading the CA certificate chain, proceed to [Building a TLS certificate chain for the Certificate Enrollment Gateway certificate](#).

Issuing TLS certificates with an on-premises CA

If you are using Certificate Enrollment Gateway with an on-premises CA, you can use native Linux tools to create a CSR (certificate signing request) for the Certificate Enrollment Gateway certificate, then use your existing CA tools to process the CSR and create the certificate.

- [Creating or recovering a user account in an on-premises CA](#)
- [Processing the CSR with an on-premises CA](#)
- [Obtaining the CA certificate chain](#)

Creating or recovering a user account in an on-premises CA

To issue a certificate for Certificate Enrollment Gateway, a user account for the certificate must exist in your on-premises CA. You must create a user account to issue the initial Certificate Enrollment Gateway certificate. You must recover (reset) the user account to renew the Certificate Enrollment Gateway certificate.

To manually create or recover (reset) a user account, you can use an administration application such as Entrust Authority Security Manager Administration or the User Management Service (Entrust Administration Services). When creating a new user account:

- It is recommended that you configure the user's name (using the directory naming attributes) to be the fully qualified domain name of the server hosting Entrust PKI Hub 1.0. For example, `example.com`.
- Select a 1-key-pair certificate type with a Dual Usage certificate definition that includes an Extended Key Usage extension with server authentication and client authentication. The certificate definition should also be assigned a certificate definition policy. For example, the Enterprise Machine (ent_machine) certificate type.
- For the Subject Alternative Name (SubjectAltName) extension, add a DNS Name component for each DNS name that may be used by the Entrust PKI Hub 1.0 cluster.

For information about creating or recovering user accounts, see the documentation for the client application.

Processing the CSR with an on-premises CA

You can process the CSR using the Profile Creation Utility. The Profile Creation Utility is a command line utility that can create and manage Entrust profiles for an on-premises Security Manager CA. You can use the Profile Creation Utility to process Certificate Signing Requests (CSRs) and generate certificates. The Profile Creation Utility is available as a separate software download for Entrust CA Gateway.

- i** When processing a CSR, the Profile Creation Utility will prompt you for the certificate definition required for the certificate. In Security Manager, that certificate definition for the user's certificate type must be assigned a certificate definition policy (user policy). If no certificate definition policy is assigned to the certificate definition you specify, an error will occur and the Profile Creation Utility will fail to process the CSR.

To download and install the Profile Creation Utility

1. Install a Java Development Kit (JDK) and set the `JAVA_HOME` environment variable.
2. Log in to Entrust TrustedCare (<http://trustedcare.entrust.com>).
3. Go to **PKI > Authority > CA Gateway** and click the latest version of the product.
4. Under software downloads, download the Profile Creation Utility for your preferred operating system:
 - `cagw-profilecreationutility-linux64-version.zip` for Linux 64-bit.
 - `cagw-profilecreationutility-win64-version.zip` for Windows 64-bit.
5. Extract the file contents of the ZIP file to a location on the computer.

To process the CSR using the Process Creation Utility

1. Obtain the CSR file along with the reference number and authorization code associated with the Security Manager user account.
When you create a user in Security Manager or set a user for key recovery, Security Manager generates a reference number and authorization code. You need these activation codes to process the CSR.
2. To process the CSR, the Profile Creation Utility requires an Entrust desktop profile (EPF file). the role associated with the profile requires the following permissions:
 - Under the **Certificates** permission category: permissions to administer the certificate category and certificate type of the certificate being issued.
 - Under the **Groups** permission category: **View** and permission to administer the group associated with the Security Manager user being issued the certificate.
 - Under the **Roles** permission category: **View** and permission to administer the role associated with the Security Manager user being issued the certificate.
 - Under the **Searchbase** permission category: **View** and permission to administer the searchbase associated with the Security Manager user being issued the certificate.
 - Under the **Users** permission category: **View** and **Perform PKIX** requests. Obtain the Entrust desktop profile (EPF file) from a Security Manager administrator.
3. Navigate to the directory containing the Profile Creation Utility.
4. Run the following command:
 - On Windows, run `pcu.bat` .
 - On Linux, run `pcu` .
5. The Profile Creation Utility main menu appears:

```
Main Menu
1. Exit
2. Help
3. Create Entrust profile
4. Recover Entrust profile
5. Inspect Entrust profile (read only)
6. Inspect and update Entrust profile (read/write)
7. Create Server Login credentials
8. Create PKCS #12 file (Security Manager)
9. Recover PKCS #12 file (Security Manager)
```

- 10. Create PKCS #12 file (3rdParty)
 - 11. Update PKCS #12 file (3rdParty)
 - 12. Process PKCS #10 Certificate Signing Request (CSR)
 - 13. Generate/Process Certificate Signing Request on HSM (3rdParty)
 - 14. Change password
- Select an operation [3]:

i To return to the main menu at any time, enter a period (.). For help about using the Profile Creating Utility, enter 2 in the main menu.

Enter 12 to process the CSR.

6. The following prompt appears:

Take settings from an existing entrust.ini file (y/n) [y]:

- To use Certificate Authority (CA) connection settings from an existing `entrust.ini` file, enter `y`.
- To provide CA connection settings manually, enter `n`.

7. If you chose to use an existing `entrust.ini` file, you are prompted to enter the full path to the `entrust.ini` file:

Enter full path to entrust.ini file:

Enter the full path and file name of the `entrust.ini` file.

8. If you chose to enter CA connections setting manually, the following prompts appear:
- a. You are prompted to provide the host name (or IP address) and port of the CA server:

Enter the CA hostname or IP address and port in the form `name:port`:

Enter the host name (or IPv4 address) and CMP port of the server hosting the CA in format of `<hostname>:<port>`. If you omit the port number, it defaults to 829.

- b. You are prompted to provide the host name (or IP address) and port of the directory server:

Enter the directory hostname or IP address and port in the form `name:port`:

Enter the host name (or IPv4 address) and LDAP port of the server hosting the directory in format of `<hostname>:<port>`. The name or address defaults to the same value that you entered for the CA address. If you omit the port number, it defaults to 389.

9. You are prompted for the full path to an administration profile:

Enter full path to administration profile:

Enter the full path and file name of an administration profile.

10. You are prompted to enter the profile password:

Enter profile password:

Enter the profile password.

11. You are asked if the CSR is authenticated:

Is the CSR authenticated? (y/n)? [n]:

Enter **n**. The CSR is not authenticated.

12. You are prompted for the full path to the CSR:

Enter full path to CSR:

Enter the full path and file name of the CSR.

13. You are prompted to enter the reference number for the CSR:

Enter reference number:

Enter the reference number you recorded earlier.

14. You are prompted to enter the authorization code for the CSR:

Enter authorization code:

Enter the authorization code you recorded earlier.

15. You are prompted to enter a file name for the certificate:

Enter certificate file to create:

Enter the full path and file name for the certificate file.

16. You are prompted to enter the certificate definition required for the certificate:

Enter certificate definition required [Verification]:

Enter the certificate definition required for the certificate, such as Verification or Dual Usage.

17. The Profile Creation Utility processes the certificate. If the operation is successful, Security Manager issues a certificate and the Profile Creation Utility writes the certificate to a file.

```
Requesting certificate from Security Manager...
Obtained new certificate with serial number 1340207625 from issuer
o=Example,c=US
Certificate written to c:\new_certificate.cer
```

After processing the CSR and obtaining the certificate, proceed to [Obtaining the CA certificate chain](#).

Obtaining the CA certificate chain

Entrust PKI Hub 1.0 requires the full TLS certificate chain for the Certificate Enrollment Gateway certificate, from the TLS certificate up to the root CA. Obtain all CA certificates in the CA certificate chain, from the issuing CA to the root CA. (In your environment, the issuing CA may be the root CA.) The CA certificates must be in PEM format.

After obtaining all CA certificates in the certificate chain, proceed to [Building a TLS certificate chain for the Certificate Enrollment Gateway certificate](#).

Building a TLS certificate chain for the Certificate Enrollment Gateway certificate

Entrust PKI Hub 1.0 requires the full TLS certificate chain for the Certificate Enrollment Gateway certificate, from the TLS certificate up to the root CA. You must combine all certificates in the TLS certificate chain into one file as described in the following procedure.

To combine the Certificate Enrollment Gateway certificate and CA certificates into a single file

1. Create a new text file.
2. Copy the contents of the Certificate Enrollment Gateway certificate (including BEGIN CERTIFICATE and END CERTIFICATE lines) into the new text file.
3. At the end of the new text file, copy the contents of each CA certificate in the chain (including BEGIN CERTIFICATE and END CERTIFICATE lines), in order from the Issuing CA certificate to the Root CA certificate. For example:

```
-----BEGIN CERTIFICATE-----  
<TLS server certificate in Base64 encoding>  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
<Issuing CA certificate in Base64 encoding>  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
<Root CA certificate in Base64 encoding>  
-----END CERTIFICATE-----
```

For Entrust PKI as a Service, the Issuing CA and Root CA are different CAs. For an on-premises CA, the Issuing CA may be the root CA. If the issuing CA is the root CA, the file would contain only the TLS certificate and the root CA.

4. The text file should look similar to the following:

```
-----BEGIN CERTIFICATE-----  
MIIDqQYJKoZIhvcNAQcCoIIDmjCCA5YCAQExADALBgkqhkiG9w0BBwGgggN...  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
MIIDejCCAmKgAwIBAgIQQ8e7ock59Y21Mtcy7rGJUDANBgkqhkiG9w0BAQs...  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
MIQ0EgrW50cnkwHhcNMjMwMjA4MTUxNzEwWWhcNMzMwMjA4MTU0NzEwWjAyM...  
-----END CERTIFICATE-----
```

5. Save the file. It is recommended that you save the file with a `.pem` or `.crt` extension. For example, `tlscertchain.pem`.

After building the TLS certificate chain, proceed to [Installing the Certificate Enrollment Gateway certificate chain into Entrust PKI Hub 1.0](#).

Installing the Certificate Enrollment Gateway certificate chain into Entrust PKI Hub 1.0

After building the TLS certificate chain for the Certificate Enrollment Gateway certificate, you can install the certificate into Entrust PKI Hub 1.0. To install the certificate, Entrust PKI Hub 1.0 requires the following:

- A single file containing the TLS certificate chain, from the TLS certificate to the root CA. You created this file earlier in [Building a TLS certificate chain for the Certificate Enrollment Gateway certificate](#).
- The private key for the certificate. The private key was generated when you created the CSR for the certificate.

i For more information about the `clusterctl certificate` command, see the Entrust PKI Hub 1.0 documentation.

To install the Certificate Enrollment Gateway certificate into Entrust PKI Hub 1.0

1. On the Entrust PKI Hub 1.0 node where the Certificate Enrollment Gateway certificate chain is located, log in as the user account that owns Entrust PKI Hub 1.0.
2. Navigate to the directory containing the Entrust PKI Hub 1.0 `clusterctl` command.
3. Enter the following command:

```
sudo clusterctl certificate --cert <tls certificate chain> --key <private key>
```

The following table describes the command parameters.

Parameter	Description
<code>--cert <tls certificate chain></code>	The path and file name of a PEM-formatted file containing the entire TLS certificate chain.
<code>--key <private key></code>	The path and file name of a PEM-formatted file containing the private key for TLS.

For example:

```
sudo clusterctl certificate --cert /home/user/ceg/corporate.example.com/tls.crt  
--key /home/user/ceg/corporate.example.com/tls.key
```

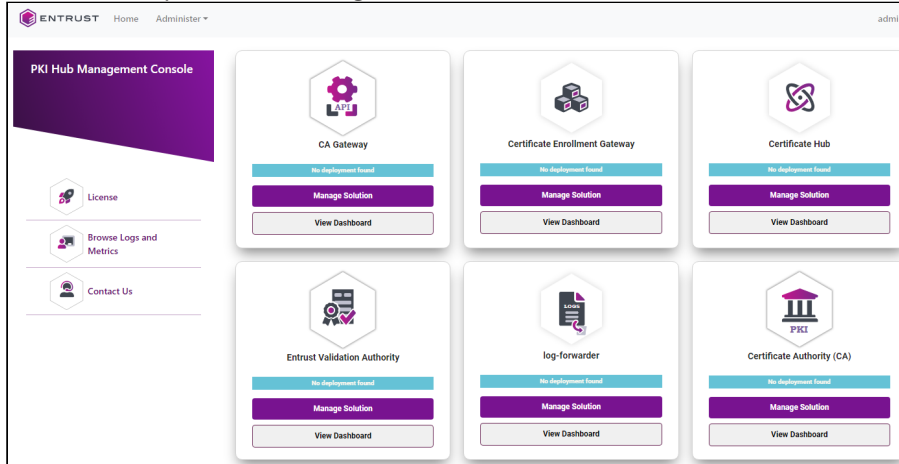
Configuring and deploying Certificate Enrollment Gateway

See below for configuring and deploying Certificate Enrollment Gateway with the Management Console.

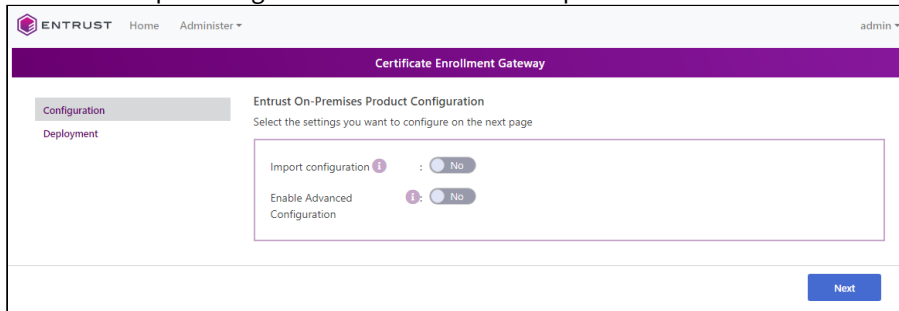
i Repeat the following steps each time a configuration update is required. Do not forget to click **Deploy** to make the changes effective.

To configure and deploy Certificate Enrollment Gateway with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).
2. In the content pane, click **Manage Solution** under **Certificate Enrollment Gateway**.



3. Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



4. Click **Next**.
5. Configure the solution settings described in the following sections.
 - [Tenants](#)
 - [CAGW](#)
 - [ACMEv2](#)
 - [MDMWS](#)
 - [Intune](#)
 - [SCEP](#)
 - [WSTEP](#)
6. Click **Validate** to validate the configured settings.
7. Correct any detected configuration error until the **Validate** option displays no warnings.
8. Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
9. Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
10. Click **Deploy**.

Tenants

Select the **Tenants** tab of the **Configuration** page to configure the following settings.

- [CEG Tenant Unique ID](#)
- [CEG Web Admin Username](#)
- [CEG Web Admin Password](#)

CEG Tenant Unique ID

A unique tenant identifier for the CEG Service. The selected identifier does not need to match any value defined in CA Gateway.

This value will be present in incoming enrollment request URLs. The value is case-sensitive and can include only the following characters:


- Dashes (-)
- Underscores (_)
- Tildes (~)
- Uppercase letters (A to Z)
- Lowercase letters (a to z)
- Numbers (0 to 9)

 Certificate Enrollment Gateway supports only one tenant.

Mandatory: Yes.

CEG Web Admin Username


The username to log in to the CEG Web Admin interface.

 The CEG Web Admin interface is for troubleshooting or modifying the log levels under Entrust Customer Support's guidance.

Mandatory: For security reasons, it is recommended to change the `WebAdmin` initial username.

CEG Web Admin Password

The password to log in to the CEG Web Admin interface.

 The CEG Web Admin interface is for troubleshooting or modifying the log levels under Entrust Customer Support's guidance.

Mandatory: For security reasons, it is recommended to change the `changeme` initial password.

CAGW

Select the **CAGW** tab of the **Configuration** page to configure the connection with an Entrust CA Gateway instance.

- [CA Gateway URL](#)
- [CAGW Keystore File \(P12\)](#)
- [CAGW Keystore Password](#)
- [CAGW Keystore Alias](#)
- [Trusted CA Certificates File Format](#)
- [RA Certificate Profile IDs](#)

CA Gateway URL

The URL of CA Gateway. This URL:

- Must not contain the API version – for example, it must not contain `"/api/v1"`.
- Must not end with a trailing slash `"/"`.

For example:


```
https://cagw.example.com/cagw
```

Mandatory: Yes.

CAGW Keystore File (P12)

A CA Gateway keystore file. This file must be a P12 file containing a private key and client certificate for Certificate Enrollment Gateway.

Mandatory: Yes.

CAGW Keystore Password

The password of the CA Gateway Keystore file.

Mandatory: Yes.

CAGW Keystore Alias

The alias of the private key entry (`PrivateKeyEntry`) in the CA Gateway Keystore. Run the following command to list all alias names in the `<file>` keystore.

```
keytool -v -list -keystore <file>
```

Mandatory: When the CA Gateway Keystore contains more than one private key.

Trusted CA Certificates File Format

The format of the file containing the CA certificate chain for the CA Gateway client credential. Select one of the following values.

- [Re-use the CAGW Keystore File](#)
- [P12](#)
- [PEM](#)

Mandatory: Yes.

Re-use the CAGW Keystore File

Select this value to re-use the [CAGW Keystore File \(P12\)](#).

 The deployment will fail if the [CAGW Keystore File \(P12\)](#) does not contain at least a Trusted CA Certificate entry.

P12

Select this value to:

- Import a P12 truststore file in the **CAGW Truststore File (P12)** field
- Enter a P12 truststore password in the **CAGW Truststore Password** field.

PEM

Select this value to import a PEM-formatted file in the **CA Certificates File (PEM)** field.

RA Certificate Profile IDs

One or more **RA Certificate Profile ID** mappings to issue an RA certificate. Define the following values for each mapping.

- [Key Name](#)
- [Value](#)

 Certificate Enrollment Gateway uses RA certificates to sign and encrypt SCEP PKI messages.


Mandatory: Add at least one mapping.

Key Name

The **CA ID** defined in the CA Gateway instance for the CA that will generate the RA certificate.

Value

The name of a **Profile ID** defined in the CA Gateway instance to issue the RA certificate.

 The selected **Profile ID** must allow Dual Usage (both Digital Signature and Key Encipherment).

ACMEv2

Select the **ACMEv2** tab of the **Configuration** page to configure ACMEv2 enrollment.

- [Enable ACMEv2](#)
- [ACMEv2 Order Expiry Interval](#)
- [Delete Expired Order Cron Job](#)
- [Delete Expired Authorizations Cron Job](#)
- [ACMEv2 DNS-01 Nameservers](#)
- [ACMEv2 DNS-01 Query Timeout](#)
- [ACMEv2 HTTP-01 Retry Count](#)
- [ACMEv2 HTTP-01 Retry Interval](#)
- [ACMEv2 HTTP-01 Redirect on POST](#)

Enable ACMEv2

Select **Yes** to enable the ACMEv2 protocol, **No** to disable the ACMEv2 protocol.

Mandatory: No.

ACMEv2 Order Expiry Interval

The period of time an ACMEv2 order can remain unprocessed by a client before the ACMEv2 server marks the order as "invalid". Enter a period in ISO-8601 duration format:

PnDTnHnMn.nS

Mandatory: No. This setting defaults to `p7D` (seven days).

Delete Expired Order Cron Job

The schedule for Certificate Enrollment Gateway to remove expired ACMEv2 orders from the internal database. The value must be a cron schedule expression in the following format:

```
<second> <minute> <hour> <day-of-month> <month> <day-of-week>
```

For example, to run the cron job every 1 hour:

```
0 0 * ? * *
```

Mandatory: No. If this setting is absent, Certificate Enrollment Gateway removes expired ACMEv2 orders every 1 hour.

Delete Expired Authorizations Cron Job

The schedule for Certificate Enrollment Gateway to remove expired ACMEv2 authorizations from the internal database. The value must be a cron schedule expression in the following format:

```
<second> <minute> <hour> <day-of-month> <month> <day-of-week>
```

For example, to run the cron job every 1 hour:

```
0 0 * ? * *
```

Mandatory: No. If this setting is absent, Certificate Enrollment Gateway removes expired ACMEv2 authorizations every 1 hour.

ACMEv2 DNS-01 Nameservers

The list of DNS nameservers for DNS-01 validation for ACMEv2. Use the following syntax to enter the IPv4 address and port (typically port 53) of each DNS nameserver.

```
<IP>:<PORT>
```

For example:

```
192.0.2.0:53
```

Mandatory: No. If this setting is absent, the ACMEv2 service will use the nameservers in the `resolv.conf` file.


ACMEv2 DNS-01 Query Timeout

The number of milliseconds to continue attempting DNS-01 Validation before timing out.

Mandatory: No. This value defaults to 10000 (10 seconds).

ACMEv2 HTTP-01 Retry Count

The maximum number of times the CEG ACMEv2 Enrollment Service will retry HTTP-01 Validation before timing out.

 HTTP-01 Validation attempts can fail when the HTTP-01 challenge server responds with the *503 Service Unavailable* HTTP code.

This setting supports a value range from 0 to unlimited.

Mandatory: No. This setting defaults to 4.

ACMEv2 HTTP-01 Retry Interval

The number of seconds to wait between HTTP-01 Validation attempts.

 HTTP-01 Validation attempts can fail when the HTTP-01 challenge server responds with the *503 Service Unavailable* HTTP code.

If set to 0, the ACMEv2 server will wait 1 second after the first connection failure and 2 seconds after each subsequent failure.

Mandatory: No. This defaults to 0.

ACMEv2 HTTP-01 Redirect on POST

Whether to enable redirects when the client responds with the 302, 307, or 308 HTTP code. Select:

- **Yes** to enable redirects and follow redirects up to 50 hops.
- **No** to disable redirects, mark the challenge as failed, and flag the associated client order as invalid.

Mandatory: No. This setting defaults to **No**.

MDMWS

Select the **MDMWS** tab of the **Configuration** page to configure MDMWS enrollment.

- [Enable MDMWS](#)
- [MDM-SCEP Token Expire Lifetime](#)
- [MDMWS Expired Token Clean-up Cron Job](#)
- [MDMWS Users](#)
- [MDMWS Enrollment Service Configuration](#)

Enable MDMWS

Select **Yes** to enable the MDM Web Service (MDMWS) and MDM-SCEP protocols; **No** to disable them.

Mandatory: No. This setting defaults to **No**.

MDM-SCEP Token Expire Lifetime

The lifetime of MDM challenges, in seconds. The minimum permitted value is 1 second.

Mandatory: Yes.

MDMWS Expired Token Clean-up Cron Job

The frequency for removing expired MDM challenges from the internal database. The value must be a cron schedule expression in the following format:

<second> <minute> <hour> <day-of-month> <month> <day-of-week>

Mandatory: Yes.

MDMWS Users

Configure a **Username** and **Password** for each user of the MDMWS protocol.

i The MDMWS protocol is protected with username and password authentication. Clients such as the Mobile Device Management (MDM) software must authenticate to Certificate Enrollment Gateway using valid username and password credentials.

MDMWS Enrollment Service Configuration

Configure one or more Digital ID Configurations for the MDM protocols.

- [Digital ID](#)
- [CAGW CA ID](#)
- [CAGW Profile ID](#)
- [Parent DN](#)
- [RDN Format](#)

i An **MDMWS Digital ID Configuration** is a template Certificate Enrollment Gateway uses to issue digital IDs for a mobile device with an MDM protocol.

Digital ID

A unique name for the Digital ID Configuration. This name:

- Must be at least four characters long,
- Must contain only letters, numbers, underscores, spaces, and hyphens.

Mandatory: Yes.

CAGW CA ID

The CA identifier (CA ID) in CA Gateway of the CA for certificate enrollments.

Mandatory: Yes.

CAGW Profile ID

The profile identifier (Profile ID) in CA Gateway of the CA for certificate enrollments.

Mandatory: Yes.

Parent DN

The parent DN (distinguished name) for certificates issued by Certificate Enrollment Gateway.

- For an on-premises CA, the parent DN must be a known searchbase defined in the CA.
- For Entrust PKI as a Service, the parent DN must be an absent or custom parent DN.

Examples:

```
ou=Devices,o=My Company,c=US
```

```
cn=Users,ou=North America,o=My Company,c=GB
```

Certificate Enrollment Gateway will build the rest of the client's DN when enrolling the client.

Mandatory: No.

RDN Format

This setting specifies the relative distinguished name (RDN) format that Certificate Enrollment Gateway uses to find and create users.

Enclose variables with angled brackets (< and >). The incoming MDMWS request must have these variables defined. Examples:

```
cn=<firstname> <lastname>  
cn=<igusername> <iggroup> <devicetype>
```

Mandatory: Yes.

Intune

Select the **Intune** tab of the **Configuration** page to configure Intune-SCEP enrollment.

- [Enable InTune-SCEP](#)
- [InTune Revocation Cron Job](#)
- [InTune-SCEP Enrollment Service Configurations](#)

Enable InTune-SCEP

Select **Yes** to enable the Intune-SCEP protocol, **No** to disable the Intune-SCEP protocol.

Mandatory: No. This setting defaults to **No**.

InTune Revocation Cron Job

The schedule for Certificate Enrollment Gateway to check Microsoft Intune for revocations. The value must be a cron schedule expression in the following format:

```
<second> <minute> <hour> <day-of-month> <month> <day-of-week>
```

For example, to run the cron job every 15 minutes:

```
0 0/15 * * * ?
```

Mandatory: No. This setting defaults to a period of 15 minutes.

InTune-SCEP Enrollment Service Configurations

Configure the following settings for each **InTune-SCEP Enrollment Service Configuration**.

- [CAGW CA ID](#)
- [Azure Application ID](#)
- [Azure Tenant](#)
- [Azure Authentication Method](#)
- [Override Default InTune Endpoints](#)


CAGW CA ID

The CA identifier (CA ID) for certificate enrollments defined in Entrust PKI as a Service or an on-premises CA Gateway.

Mandatory: Yes.

Azure Application ID

The Application ID of the Registered Azure Application, as viewed from Azure.

 The selected Registered Azure Application must have the proper API permissions.

Mandatory: Yes.

Azure Tenant

The tenant for connecting to the Intune instance. For example:

test.example.com

Mandatory: Yes.

Azure Authentication Method

If the registered Azure application authenticates with certificate-based (P12) credentials, select **App P12** and configure the following settings.

Setting	Value
Registered Azure Application Credential Key Store File (P12)	The P12 file containing the Azure application credential.
Registered Azure Application Key Store Password	The password of the P12 credential

If the application authenticates with application keys (client secrets), select **App Secret** and configure the following setting.

Setting	Value
Registered Azure Application Key (Client Secret)	The application key (client secret) for connecting to the Intune instance.

Mandatory: Yes.

Override Default InTune Endpoints

Select **Yes** to override some default setting values and work with the following non-default Intune endpoints.

Setting	Value
Override: Authentication Authority	The URL of the Microsoft authentication authority.
Override: Graph API Version	The version of the Microsoft Graph API.

Setting	Value
Override: Graph Resource URL	This URL of the Microsoft Graph Resource.
Override: InTune Application ID	The application ID of Microsoft Intune.
Override: InTune Resource URL	The URL of the Microsoft Intune Resource.

Select **No** to support only default Intune endpoints.

Mandatory: No. This setting defaults to **No**.

SCEP

Select the **SCEP** tab of the **Configuration** page to configure SCEP enrollment.

- [Enable SCEP](#)
- [SCEP Enrollment Service Configurations](#)

Enable SCEP

Select **Yes** to enable the SCEP protocol; **No** to disable the SCEP protocol.

Mandatory: No. This setting defaults to **No**.

SCEP Enrollment Service Configurations

Configure the following settings for each **SCEP Enrollment Service Configuration**.

- [CAGW CA ID](#)
- [SCEP Challenge Password](#)
- [Insecure SCEP \(Permit an empty challenge password\)](#)
- [Revoke Old Certificate on Renewal](#)


CAGW CA ID

The CA identifier (CA ID) of the CA for certificate enrollments, as defined in Entrust PKI as a Service or an on-premises CA Gateway.

Mandatory: Yes.

SCEP Challenge Password


A challenge password for SCEP clients.

 This setting only applies to requests sent through the SCEP Service. It does not apply to requests sent through the Intune-SCEP Service because Microsoft Intune validates these requests.

Mandatory: When the [Insecure SCEP \(Permit an empty challenge password\)](#) box is not checked.

Insecure SCEP (Permit an empty challenge password)


Check this box to allow an empty [SCEP Challenge Password](#); uncheck this box to make the challenge password mandatory.

 Deliberately configuring SCEP without a challenge password can be a security risk.

Mandatory: No. This box is unchecked by default.

Revoke Old Certificate on Renewal

Check this box to revoke the old certificate with the "Superseded" reason after issuing the new certificate. Uncheck this box for the old certificate to be valid after the new certificate is issued.

 This setting only applies to requests sent through the SCEP Service. It does not apply to requests sent through the Intune-SCEP Service because Microsoft Intune validates these requests.

Mandatory: No. By default, this box is not checked.

WSTEP

Select the **WSTEP** tab of the **Configuration** page to configure WSTEP enrollment.

- [Enable WSTEP](#)
- [WSTEP CAGW Settings](#)
- [Active Directory Domains](#)

Enable WSTEP

Select **Yes** to enable the WSTEP protocol; **No** to disable the WSTEP protocol.

Mandatory: No. This setting defaults to **No**.

WSTEP CAGW Settings

Configure the following settings for each enrollment connection with CA Gateway.

- [CAGW CA ID](#)
- [Parent DN](#)
- [CAGW Profile ID for Digital Signature](#)
- [CAGW Profile ID for Key Encipherment](#)
- [CAGW Profile ID for Digital Signature and Key Encipherment](#)
- [CAGW Profile ID for Digital Signature and Nonrepudiation](#)
- [Certificate Templates](#)

CAGW CA ID

The CA identifier (CA ID) in CA Gateway of the CA for WSTEP enrollment.

Mandatory: Yes.

Parent DN

The parent DN (distinguished name) for certificates issued by the CEG WSTEP service. The selected value is appended to incoming Subject DNs.

CA Type	Parent DN
Security Manager	A known searchbase defined in Security Manager.

CA Type	Parent DN
Entrust PKI as a Service	An absent parent DN, or a user-defined or custom parent DN.

Examples:

```
ou=Devices, o=My Company, c=US
```

```
cn=Users, ou=North America, o=My Company, c=GB
```

Mandatory: No.

CAGW Profile ID for Digital Signature

The unique ID defined in CA Gateway for the WSTEP signing certificate profile.

Mandatory: Yes.

CAGW Profile ID for Key Encipherment

The unique ID defined in CA Gateway for the WSTEP encryption certificate profile.

Mandatory: Yes.

CAGW Profile ID for Digital Signature and Key Encipherment

This unique ID defined in CA Gateway for the WSTEP signing and encryption certificate profiles.

Mandatory: Yes.

CAGW Profile ID for Digital Signature and Nonrepudiation

The unique ID defined in CA Gateway for the WSTEP signing and nonrepudiation certificate profile.

Mandatory: Yes.

Certificate Templates

The required mappings for each certificate template.

- For **Value**, enter the name of a Profile ID defined in CA Gateway for issuing the certificate.
- For **Value**, enter the name of a Profile ID defined in CA Gateway for issuing the certificate.

Note that:

- For machines, the **Subject name** in the certificate template must be **Common name** or **DNS name**.
- For users, the **Subject name** in the certificate template must be **Common name**.

See [Subject Name tab](#) for information on configuring the **Subject name** in the Windows certificate template.

Mandatory: No.

Active Directory Domains

Configure the following settings for each connection with Active Directory domains.

- [Domain Name](#)

- [Computer Name](#)
- [Enable WSTEP Kerberos Authentication for WSTEP Enrollment](#)
- [Authentication Type for LDAP and Global Catalog Connections](#)

Domain Name

The Active Directory domain name. The Active Directory domain name is the root domain naming context of the Active Directory. For example:

```
example.com
```

To retrieve the Active Directory domain name, run the following PowerShell command:

```
([ADSI]"LDAP://RootDSE").rootDomainNamingContext -replace '^DC=', '' -replace '.DC=', ''
```

You can specify multiple domain names by specifying multiple **Domain Name** settings. This setting contains child settings that configure the connection to the Active Directory domain.

Mandatory: Yes.

Computer Name

The fully qualified domain name (FQDN) of the Active Directory domain controller. For example:

```
activedirectory.example.com
```

If this setting is configured, Certificate Enrollment Gateway will use this FQDN for LDAP and Global Catalog connections.

If this setting is not configured, Certificate Enrollment Gateway will use DNS to find the Active Directory domain controller from the domain name specified in the WSTEP request.

Mandatory: No.

Enable WSTEP Kerberos Authentication for WSTEP Enrollment

Select **Yes** to enable Kerberos authentication for WSTEP enrollment and configure the following settings.

- [Principal](#)
- [Keytab File](#)
- [Permit Deprecated Algorithms \(3DES and RC4\)](#)

Select **No** to disable Kerberos authentication for WSTEP enrollment.

Mandatory: No. This setting defaults to **No**.

Principal

The Kerberos principal that the CEG Service will use to authenticate to each Active Directory forest for cross-forest WSTEP enrollment. You must use the same Kerberos principal to generate the keytab file used for Kerberos v5 LDAP referrals.

The value must be a string with the following syntax:

HTTP/<ceg-fqdn>

Mandatory: When cross-forest trust must be supported for WSTEP enrollment with Kerberos authentication.

Keytab File

The name of the Kerberos keytab file for the domain controller. The keytab file is used to authenticate incoming WSTEP requests.

Mandatory: When cross-forest trust must be supported for WSTEP enrollment with Kerberos authentication.

Permit Deprecated Algorithms (3DES and RC4)

Check this box to permit the 3DES and RC4 deprecated Kerberos algorithms. Uncheck this box to reject these algorithms.

Mandatory: No. This box is unchecked by default.

Authentication Type for LDAP and Global Catalog Connections

This authentication method for LDAP and Global Catalog connections to Active Directory. Select:

- **Username/Password** for username and password authentication. This option requires configuring the [LDAP Connection Settings](#).
- **Kerberos (Required for Cross-Forest Enrollments)** for Kerberos authentication. This option requires configuring the [LDAP Connection Settings](#) and the [Kerberos LDAP Referrals](#).

Mandatory: Yes.

LDAP Connection Settings

Configure the LDAP connection to Active Directory.

- [Enable LDAPS](#)
- [CA Certificates File Format \(P12 or PEM\)](#)
- [WSTEP LDAPS Trusted CA Certificates File \(PEM\)](#)
- [WSTEP LDAPS Truststore File \(P12\)](#)
- [LDAPS Truststore Password](#)
- [LDAP Port](#)
- [Global Catalog Port](#)
- [LDAP Connect Timeout](#)
- [LDAP Read Timeout](#)
- [Username](#)
- [Password](#)

Enable LDAPS

Select **Yes** to use LDAPS (secure LDAP) for all connections to Active Directory; select **No** to use LDAP for connections to Active Directory.

Mandatory: No. This setting defaults to **No**.

CA Certificates File Format (P12 or PEM)

The format of the file containing the CA certificate chain for the Active Directory server certificate. Select;

- **P12** if the file is a P12 truststore.
- **PEM** if the file is a PEM-formatted file.

Mandatory: When **Enable LDAPS** is **Yes**.

WSTEP LDAPS Trusted CA Certificates File (PEM)

A PEM-formatted file that contains the CA certificate chain for Active Directory's server certificate.

Mandatory: When **CA Certificates File Format (P12 or PEM)** is **PEM**.

WSTEP LDAPS Truststore File (P12)

An LDAPS Truststore file (P12 file) that contains the CA certificate chain for Active Directory's server certificate.

Mandatory: When **CA Certificates File Format (P12 or PEM)** is **P12**.

LDAPS Truststore Password

The password of the LDAPS Truststore file for LDAPS authentication to Active Directory.

Mandatory: When **CA Certificates File Format (P12 or PEM)** is **P12**.

LDAP Port

The LDAP or LDAPS port to connect to Active Directory.

Mandatory: No. This setting defaults to port 389 (LDAP) or 636 (LDAPS).

Global Catalog Port

The port for connecting with the global catalog in Active Directory.

Mandatory: No. This setting defaults to port 3268 (LDAP) or 3269 (LDAPS).

LDAP Connect Timeout

The number of milliseconds Certificate Enrollment Gateway will wait for Active Directory to establish a connection before aborting the connection attempt.

Mandatory: No. This setting defaults to 30000 (30 seconds).

LDAP Read Timeout

The number of milliseconds Certificate Enrollment Gateway will wait for Active Directory to respond to an LDAP request before aborting the read attempt.

Mandatory: No. This setting defaults to 30000 (30 seconds).

Username

The username for WSTEP to connect with the Active Directory domain. It can be the username of any Active Directory Domain user.

This domain user account must be a service logon account without any special permissions. This service account will be used for read-only access with LDAP and Global Catalog.

Mandatory: When [Authentication Type for LDAP and Global Catalog Connections](#) is **Username/Password**.

Password

The password for WSTEP to connect to Active Directory.

Mandatory: When [Authentication Type for LDAP and Global Catalog Connections](#) is **Username/Password**.

Kerberos LDAP Referrals

Configure the following settings when [Authentication Type for LDAP and Global Catalog Connections](#) is **Kerberos (Required for Cross-Forest Enrollments)**.

- [Principal](#)
- [Keytab File](#)
- [Kerberos Configuration File](#)
- [Maximum LDAP Referrals](#)

Principal

The Kerberos principal that the CEG Service will use to authenticate to each Active Directory forest for cross-forest WSTEP enrollment. You must use the same Kerberos principal to generate the keytab file used for Kerberos v5 LDAP referrals.

The value must be a string with the following syntax:

```
HTTP/<ceg-fqdn>
```

Where `<ceg-fqdn>` is the fully qualified domain name (FQDN) of the server hosting Certificate Enrollment Gateway. For example:

```
HTTP/cegserver1.example.com
```

Mandatory: Only when cross-forest trust must be supported for WSTEP enrollment with Kerberos authentication.

Keytab File

A keytab file for the domain controller. The keytab file is used to authenticate incoming WSTEP requests.

Mandatory: Only when cross-forest trust must be supported for WSTEP enrollment with Kerberos authentication.

Kerberos Configuration File

A Kerberos configuration file for Kerberos V5 LDAP Referrals.

Mandatory: Only when cross-forest trust must be supported for WSTEP enrollment with Kerberos authentication.

Maximum LDAP Referrals

The maximum number of Kerberos V5 LDAP Referrals to follow. The value must be an integer from 0 to 10.

Mandatory: Yes.

Enrollment URLs for Certificate Enrollment Gateway

When deploying or redeploying Certificate Enrollment Gateway into Entrust PKI Hub 1.0, Entrust PKI Hub 1.0 will display a list of local test commands and enrollment URLs for Certificate Enrollment Gateway.

- [ACMEv2 enrollment URL](#)
- [Intune-SCEP enrollment URL](#)
- [MDM-SCEP enrollment URL](#)
- [MDMWS enrollment URL](#)
- [SCEP enrollment URL](#)
- [WSTEP enrollment URL](#)

ACMEv2 enrollment URL

ACMEv2 clients must use the following URL to communicate with Certificate Enrollment Gateway:

```
https://<CEG-server>/acme/<tenant-ID>/<CA-ID>/<profile-ID>/directory
```

Where:


- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the ACMEv2 endpoint.
- `<profile-ID>` is the profile ID defined in CA Gateway that defines the certificate type issued to the ACMEv2 client. For Entrust PKI as a Service, the profile ID is one of the following
 - `privatessl-tls-client-server`
 - `privatessl-tls-server`
 - `privatessl-tls-client`

For example:

```
https://cegserver.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory
```

Intune-SCEP enrollment URL

Microsoft Intune must be configured to use one of the following URLs to communicate with Certificate Enrollment Gateway:

 The following Intune-SCEP enrollment URL requires the trailing forward slash (/). To support macOS (Apple) devices, the URL must start with `http` instead of `https`.

```
http://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/intune/  
https://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/intune/
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the SCEP client.
- `<profile-ID>` is the profile ID defined in CA Gateway that defines the certificate type issued to the SCEP client. For Entrust PKI as a Service, the profile ID is one of the following:
 - `intune-digital-signature-key-encipherment`
 - `intune-digital-signature`
 - `intune-key-encipherment`
 - `intune-non-repudiation`

For example:

```
http://cegserver.example.com/scep/tenant1/example_ca1/intune-digital-signature-key-encipherment/intune/  
https://cegserver.example.com/scep/tenant1/example_ca1/intune-digital-signature-key-encipherment/intune/
```

MDM-SCEP enrollment URL

MDM-SCEP clients must use one of the following URLs to communicate with Certificate Enrollment Gateway:

 To support macOS (Apple) devices, the URL must start with `http` instead of `https`.

```
http://<CEG-server>/scep/<tenant-ID>/<digitalid-config>/mdm  
https://<CEG-server>/scep/<tenant-ID>/<digitalid-config>/mdm
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<digitalid-config>` is a digital ID configuration defined in the CEG Service.

For example:

```
http://cegserver.example.com/scep/tenant1/digitalid-config1/mdm  
https://cegserver.example.com/scep/tenant1/digitalid-config1/mdm
```

MDMWS enrollment URL

Mobile Device Management products must use the following URL to communicate with Certificate Enrollment Gateway:

```
https://<CEG-server>/mdm/services/<tenant-ID>
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.

For example:

```
https://cegserver.example.com/mdm/services/tenant1
```


SCEP enrollment URL

SCEP clients must use one of the following URLs to communicate with Certificate Enrollment Gateway:

- i** The following SCEP enrollment URL requires the trailing forward slash (/). To support macOS (Apple) devices, the URL must start with `http` instead of `https`.

```
http://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/  
https://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the SCEP client.
- `<profile-ID>` is the profile ID defined in CA Gateway that defines the certificate type issued to the SCEP client. For Entrust PKI as a Service, the profile ID is one of the following:
 - `scep-digital-signature-key-encipherment`
 - `scep-digital-signature`
 - `scep-key-encipherment`
 - `scep-non-repudiation`

For example:

```
http://cegserver.example.com/scep/tenant1/example_ca1/scep-digital-signature/  
https://cegserver.example.com/scep/tenant1/example_ca1/scep-digital-signature/
```

Some SCEP clients will append an additional parameter to all SCEP URLs. For these clients, you must append `nop` to the SCEP URL. For example:

```
http://cegserver.example.com/scep/tenant1/example_ca1/scep-digital-signature/nop/  
https://cegserver.example.com/scep/tenant1/example_ca1/scep-digital-signature/nop/
```

WSTEP enrollment URL

For WSTEP enrollment, the enrollment service in Active Directory must use the following URL to communicate with Certificate Enrollment Gateway:

```
https://<CEG-server>:443/wstep/<auth>/services/<tenant-ID>/<CA-ID>
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<auth>` is the authentication method, either `usertoken` for user name and password authentication or `kerberos` for Kerberos (Windows integrated) authentication.

- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example, when authenticating with a user name and password:

```
https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/example-ca1
```

For example, when authenticating with Kerberos:

```
https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/example-ca1
```

Integrating Certificate Enrollment Gateway

See below for the main integration use cases of Certificate Enrollment Gateway.

- [Integrating ACMEv2 clients with Certificate Enrollment Gateway](#)
- [Integrating Microsoft Intune with Certificate Enrollment Gateway](#)
- [Integrating SCEP clients with Certificate Enrollment Gateway](#)
- [Integrating MDM and MDM-SCEP clients with Certificate Enrollment Gateway](#)
- [Integrating WSTEP clients with Certificate Enrollment Gateway](#)

Integrating ACMEv2 clients with Certificate Enrollment Gateway

The following topics explain how to integrate ACMEv2 clients with the Certificate Enrollment Gateway service.

- [Configuring Certificate Enrollment Gateway for ACMEv2 enrollment](#)
- [Configuring ACMEv2 clients for enrollment with Certificate Enrollment Gateway](#)
- [ACMEv2 client examples](#)

Configuring Certificate Enrollment Gateway for ACMEv2 enrollment

To configure Certificate Enrollment Gateway for ACMEv2 enrollment, you must configure the ACMEv2 enrollment settings in Certificate Enrollment Gateway. You can edit the ACMEv2 enrollment settings using the Management Console interface.

To configure Certificate Enrollment Gateway for ACMEv2 enrollment

1. Log in into the Management Console as explained in [Logging into the Management Console](#).
2. In the **Certificate Enrollment Gateway** pane, click **Manage Solution**.
A **Certificate Enrollment Gateway** page appears.
3. In the left navigation bar, click **Configuration**.
A **Product Configuration** pane appears.
4. Turn on **Enable Advanced Configuration**.
5. Click **Next**.
6. Click the **ACMEv2** tab and configure the following settings.
 - [Enable ACMEv2](#)
 - [ACMEv2 Order Expiry Interval](#)
 - [Delete Expired Order Cron Job](#)
 - [Delete Expired Authorizations Cron Job](#)
 - [ACMEv2 DNS-01 Nameservers](#)

- [ACMEv2 DNS-01 Query Timeout](#)
 - [ACMEv2 HTTP-01 Retry Count](#)
 - [ACMEv2 HTTP-01 Retry Interval](#)
 - [ACMEv2 HTTP-01 Redirect on POST](#)
7. Configure any other settings if required.
 8. After configuring the settings, click **Validate** to validate the settings.
If any configuration errors are detected, correct the errors then click **Validate** again.
 9. After validating the configuration settings, click **Next**.
Entrust PKI Hub uploads the configuration and any attached files, such as P12 credentials.
 10. In the **Product Deployment Status** pane, re-deploy Certificate Enrollment Gateway with the updated configuration file by clicking **Deploy**.
A dialog box appears, prompting you to confirm the operation. Click **Yes** to confirm the operation and deploy the Certificate Enrollment Gateway solution.

Configuring ACMEv2 clients for enrollment with Certificate Enrollment Gateway

This section describes the information required to configure ACMEv2 clients to enroll for a certificate using Certificate Enrollment Gateway. For information about using your ACMEv2 client, see the documentation for your ACMEv2 client.

- [About CSRs with an empty Subject DN](#)
- [Supported validation methods](#)
- [Adding the CA certificate chain to the ACMEv2 client](#)
- [Supported algorithms for CSRs](#)
- [Enrollment URL for ACMEv2 clients](#)

About CSRs with an empty Subject DN

Some ACMEv2 clients may send a CSR with an empty Subject DN. However, certificates issued by Security Manager CAs will have a non-empty Subject DN. If an ACMEv2 client sends a CSR with an empty Subject DN, Certificate Enrollment Gateway will use the first Subject Alternative Name value in the CSR as the Subject DN.

Certificate Enrollment Gateway will not modify the Subject DN in the CSR. Certificate Enrollment Gateway will send the CSR unaltered to CA Gateway for processing, and send the Subject DN separately as a CA Gateway request parameter.

Supported validation methods

During enrollment, ACMEv2 clients must pass one of the following validation methods:

Method	Required configuration
DNS-01	Certificate Enrollment Gateway and the ACMEv2 client must point to the same DNS server. Certificate Enrollment Gateway must be able to query for DNS TXT records generated by the ACMEv2 client.
HTTP-01	Certificate Enrollment Gateway must resolve the hostname of the FQDN in the CSR. The hostname must resolve to the IP address of the ACMEv2 client. The ACMEv2 client must listen on port 80 to use HTTP-01 validation.

Adding the CA certificate chain to the ACMEv2 client

ACMEv2 clients must trust the CA certificate chain for the Entrust PKI Hub cluster's TLS certificate. The cluster's TLS certificate secures Certificate Enrollment Gateway's TLS traffic. If ACMEv2 clients do not trust the CA certificate chain, the clients will fail to establish a secure TLS connection to Certificate Enrollment Gateway.

✘ While some ACMEv2 clients may allow insecure TLS connections, you should avoid these connections for security reasons.

See your ACMEv2 client documentation for instructions about adding certificates to the ACMEv2 client.

Supported algorithms for CSRs

When an ACMEv2 client requests a certificate, the CSR (certificate signing requests) must use an algorithm supported by Certificate Enrollment Gateway.

The ACMEv2 service of the Certificate Enrollment Gateway supports the following algorithms for CSRs (certificate signing requests):

- RSA-2048, RSA-3072, RSA-4096
- EC P-256, EC P-384, EC P-521

Enrollment URL for ACMEv2 clients

ACMEv2 clients must use the following URL to communicate with Certificate Enrollment Gateway:

```
https://<CEG-server>/acme/<tenant-ID>/<CA-ID>/<profile-ID>/directory
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the ACMEv2 endpoint.
- `<profile-ID>` is the profile ID defined in CA Gateway that defines the certificate type issued to the ACMEv2 client. For Entrust PKI as a Service, the profile ID is one of the following:
 - `privatessl-tls-client-server`
 - `privatessl-tls-server`
 - `privatessl-tls-client`

For example:

```
https://cegserver.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory
```

ACMEv2 client examples

The following topics provide some examples about how to request certificates from Entrust Certificate Enrollment Gateway using some common ACMEv2 clients.

- [Certbot example](#)
- [Win-acme example](#)

- [acme.sh example](#)
- [Cert-manager.io example](#)

For complete information about using these ACMEv2 clients, see the ACMEv2 client documentation.

Certbot example

Certbot is a free, open-source software tool for automatically using digital certificates on Web sites to enable HTTPS. You can use Certbot to request certificates from Certificate Enrollment Gateway using the ACMEv2 protocol.

- [Preparing to use Certbot](#)
- [Using Certbot to request a certificate](#)

Preparing to use Certbot

Before using Certbot, configure Certbot to trust your root CA certificate using the `REQUESTS_CA_BUNDLE` environment variable.

- For example, to set the `REQUESTS_CA_BUNDLE` environment variable on Windows:

```
set REQUESTS_CA_BUNDLE=<root-CA-cert-file>
```

Where `<root-CA-cert-file>` is the path and file name of the root CA certificate file. For example:

```
set REQUESTS_CA_BUNDLE= "C:\root_ca.crt"
```

- For example, to set the `REQUESTS_CA_BUNDLE` environment variable on Linux:

```
sudo REQUESTS_CA_BUNDLE=<root-CA-cert-file>
```

Where `<root-CA-cert-file>` is the path and file name of the root CA certificate file. For example:

```
sudo REQUESTS_CA_BUNDLE=/tmp/root_ca.crt
```

Using Certbot to request a certificate

To request a certificate using Certbot, enter the following command:

```
certbot certonly -d <domain> --<CEG-ACME-URL> --standalone --no-eff-email --agree-tos  
-m <email-address>
```

Where:

- `<domain>` is a domain to include in the certificate. You can specify multiple domains using multiple `-d <domain>` parameters. For example:

```
-d example.com -d www.example.com
```

- `<CEG-ACME-URL>` is the ACMEv2 enrollment URL used to request a certificate from Certificate Enrollment Gateway. For details, see [Configuring ACMEv2 clients for enrollment with Certificate Enrollment Gateway](#).
- `--standalone` requests a certificate if you do not want to use (or do not have) existing server software. Certbot will bind on port 80 to perform domain validation. Port 80 must be available and allowed through any configured firewalls. If another application such as a Web server is running and using port 80, disable the application.
- `--no-eff-email` forces Certbot to not share your e-mail address with the Electronic Frontier Foundation.
- `--agree-tos` will cause Certbot to automatically agree to the terms of service of the ACMEv2 server (Certificate Enrollment Gateway).
- `<email-address>` is the email address that Certbot uses when registering the ACME account with Certificate Enrollment Gateway. Certificate Enrollment Gateway will not send email messages to this email address.

For example:

```
certbot certonly -d example.com -d www.example.com --server https://
cegservers.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory --
standalone --no-eff-email --agree-tos -m notifications@example.com
```

Win-acme example

Win-acme is an ACMEv2 client for Windows operating systems. You can use win-acme to request certificates from Certificate Enrollment Gateway using the ACMEv2 protocol.

To request a certificate using Win-acme

1. Import your root CA certificate into the Windows trust store.
2. Open a Command Prompt. Select **Start > Windows System > Command Prompt**.
3. Navigate to the location where Win-acme is installed.
4. Enter the following command:

```
wacs.exe --baseuri <CEG-ACME-URL> --accepttos --target manual --host <hosts> --
force
```

Where:

- `<CEG-ACME-URL>` is the ACMEv2 enrollment URL used to request a certificate from Certificate Enrollment Gateway. For details, see [ACMEv2 enrollment URL](#).
- `--accepttos` will force Win-acme to automatically accept the ACMEv2 terms of service.
- `--source manual` will use the manual plugin, allowing you to provide the host names for a certificate.
- `<hosts>` is a comma-separated list of host names to include in the certificate. The first host name listed will become the common name of the certificate. Subsequent host names will be added to the certificate as subjectAltName extensions only.
- `--force` will force a renewal if a valid certificate already exists.

For example:

```
wacs.exe --baseuri https://cegserver.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory --accepttos --target manual --host www.example.com,example.com --force
```

5. Win-acme connects to Certificate Enrollment Gateway, and prompts you to select a task:

```
A simple Windows ACMEv2 client (WACS)
Software version 2.1.17.1065 (release, pluggable, standalone, 64-bit)
Connecting to https://cegserver.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory...
Scheduled task not configured yet
Please report issues at https://github.com/win-acme/win-acme

N: Create certificate (default settings)
M: Create certificate (full options)
R: Run renewals (0 currently due)
A: Manage renewals (0 total)
O: More options...
Q: Quit

Please choose from the menu
```

6. Enter **N** to create a new certificate with the default settings.
7. Win-acme prompts you to provide one or email addresses to receive notifications about problems and abuse:

```
Enter email(s) for notifications about problems and abuse (comma-separated):
```

8. Enter an email address. Certificate Enrollment Gateway will not send email messages to this email address.
9. Win-acme requests the certificate. If the certificate is successfully created, Win-acme installs the certificate and creates a scheduled task for renewing the certificate.

acme.sh example

Acme.sh is UNIX shell script that implements the ACMEv2 client protocol. You can use acme.sh to request certificates from Certificate Enrollment Gateway using the ACMEv2 protocol.

To request a certificate using Acme.sh, enter the following command:

```
acme.sh --issue --standalone [--httpport <port>] -d <domain> --server <CEG-ACME-URL> --ca-bundle <ca-file>
```

Parameters in square brackets are optional parameters. Where:

- `--standalone` requests a certificate if you do not want to use (or do not have) existing server software.
- `--httpport <port>` specifies the standalone HTTP listen port. If not specified, the port defaults to port 80. The port (such as port 80) must be available and allowed through any configured firewalls. If another application such as a Web server is running and using the port, disable the application.

- `<domain>` is a domain to include in the certificate. You can specify multiple domains using multiple `-d <domain>` parameters. For example:

```
-d example.com -d www.example.com
```

- `<CEG-ACME-URL>` is the ACMEv2 enrollment URL used to request a certificate from Certificate Enrollment Gateway. For details, see [ACMEv2 enrollment URL](#).
- `<ca-file>` is the path and file name of the CA certificate bundle, used to verify the Certificate Enrollment Gateway server certificate.

For example:

```
acme.sh --issue --standalone -d example.com -d www.example.com --server https://
cegservers.example.com/acme/tenant1/example_ca1/privatessl_tls_client/directory --ca-
bundle /tmp/root-ca.crt
```

Cert-manager.io example

Cert-manager.io adds certificates and certificate issuers as resource types in Kubernetes clusters, and can simplify the process of obtaining, renewing, and using those certificates. You can use Cert-manager.io to request certificates from Certificate Enrollment Gateway using the ACMEv2 protocol.

i The instructions in this example use Cert-manager.io installed in a Kubernetes cluster using Helm. Other methods of configuring and deploying Cert-manager.io are available, but are not documented in this guide.

- [Cert-manager.io prerequisites](#)
- [Preparing Linux for HTTPS \(optional\)](#)
- [Deploying Kubernetes and Cert-manager.io](#)
- [Configuring Cert-manager.io for Certificate Enrollment Gateway with ACMEv2](#)

Cert-manager.io prerequisites

To use Cert-manager.io with Certificate Enrollment Gateway:

- If you will use secure HTTPS with Cert-manager.io, copy Certificate Enrollment Gateway's TLS certificate chain to the server that will host Cert-manager.io.
- For HTTP-01 validation, the DNS server must resolve the requested DNS name to the IP address of the server hosting Cert-manager.io.
- For DNS-01 validation, nonsecure and secure dynamic updates must be enabled for the domain for which Cert-manager.io is requesting certificates.
- Obtain the ACMEv2 enrollment URL used to request a certificate from Certificate Enrollment Gateway. For details, see [ACMEv2 enrollment URL](#). You need this URL later when configuring Cert-manager.io.

Preparing Linux for HTTPS (optional)

i This section is required only if the Kubernetes cluster that will host Cert-manager.io will use a trusted HTTPS connection to connect to Certificate Enrollment. If you will not use a trusted HTTPS connection, you can skip this section. You must complete this step before deploying the Kubernetes cluster.

To configure Linux to trust a CA certificate chain, complete the following steps.

To configure Linux to trust a CA certificate chain

1. Transfer Certificate Enrollment Gateway's trust certificate chain (from the issuing CA certificate to the root CA certificate) to the Linux server that will host Cert-manager.io. The certificate files must be in PEM format.
2. Log in to the Linux server that will host Cert-manager.io.
3. Copy the certificates (trust chain) into the following directory:

```
/etc/pki/ca-trust/source/anchors
```

4. Enter the following command to update the file permissions for `ca-bundle.crt` so everyone can read the file:

```
sudo chmod +r ./ca-bundle.crt
```

5. Run the following command to update the `ca-bundle.crt` file at the operating system level:

```
sudo update-ca-trust extract
```

6. Verify that the certificates were added to the following file:

```
/etc/pki/tls/certs/ca-bundle.crt
```

Deploying Kubernetes and Cert-manager.io

This section describes how to deploy a single-node Kubernetes, and then how to deploy Cert-manager.io into it.

To deploy Kubernetes and Cert-manager.io

1. Install K3s Lightweight Kubernetes by running the following commands:

```
curl -sfL https://get.k3s.io | sh -  
export PATH=$PATH:/usr/local/bin
```

2. Enable the use of kubectl permanently, by ensuring that `/usr/local/bin` appears as part of the `PATH` environment variable in the `~/.bash_profile` file.
3. Install Helm by running the following commands:

```
curl https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3 |  
bash  
echo "export KUBECONFIG=/etc/rancher/k3s/k3s.yaml" >> ~/.bash_profile  
export KUBECONFIG=/etc/rancher/k3s/k3s.yaml
```

4. Add the Cert-manager.io repository to Helm by running the following commands:

```
helm repo add jetstack https://charts.jetstack.io  
helm repo update
```

5. Install Cert-manager.io using Helm by entering the following command:

```
helm upgrade -i -n cert-manager cert-manager jetstack/cert-manager --set
installCRDs=true --create-namespace --version v1.6.0-beta.0 --wait
```

6. (Secure HTTP Only) If you want to use Cert-manager.io with trusted HTTPS connections, then you must add the TLS CA certificate chain into the cluster. Create a `private-ca-bundle.pem` file.
 - a. This file contains a concatenation of all PEM certificates in the CA certificate chain, starting with the issuing CA first and ending with the root CA last. For example:

```
-----BEGIN CERTIFICATE-----
MIIF0TCCA7mgAwIBAgIQCy...
...
V8HU0ts=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFUDCCAzigAwIBAgIQRv...
...
XLy202FpMk40J031gqbnD0usrY8=
-----END CERTIFICATE-----
```

- b. Create a config-map from the `private-ca-bundle.pem` file by running the following command:

```
kubectl create configmap private-ca-bundle -n cert-manager --from-
file=private-ca-bundle.pem
```

- c. Update the Cert-manager.io deployment to use the config-map using Helm, by running the following command:

```
helm upgrade -i -n cert-manager cert-manager jetstack/cert-manager \
--version v1.6.0-beta.0 \
--set installCRDs=true \
--set volumes[0].name=ca-certs,volumes[0].configMap.name=private-ca-
bundle \
--set volumeMounts[0].name=ca-certs,volumeMounts[0].mountPath=/etc/ssl/
certs \
--wait --wait-for-jobs
```

You have now deployed a single-node Kubernetes cluster and installed Cert-manager.io. Certificate Enrollment Gateway's TLS certificate chain is also trusted at the Cert-manager.io namespace, cluster, and operating system levels. The certificate chain still needs to be configured at the pod level.

Configuring Cert-manager.io for Certificate Enrollment Gateway with ACMEv2

The following procedures configure Cert-manager.io to request and receive certificates using DNS-01 and HTTP-01 validation. Configuring Cert-manager.io requires configuring a series of YAML files, then applying those files to Cert-manager.io. After applying the files, Cert-manager.io will automatically request the files from Certificate Enrollment Gateway.

For HTTP-01 validation, the following example uses Cert-manager.io's ingress-shim features. In this example, you will create a dummy back-end service (echo), and then an Ingress. The Ingress routes traffic into the cluster, and requests TLS certificates for the services to which it is routing.

For this example, you will create the following files:

- `dns-issuer.yaml` , to define the DNS issuer for DNS-01 validation.
- `dns-cert.yaml` , to define the DNS certificate for DNS-01 validation.
- `http-issuer.yaml` , to define the HTTP issuer for HTTP-01 validation.
- `echo.yaml` , to define the echo (dummy back-end) service for HTTP-01 validation.

In this example, the echo service is a dummy back-end to show how to secure an existing service on the Kubernetes cluster.

- `http-ingress.yaml` , to define the Ingress for HTTP-01 validation.

To create the YAML files for Cert-manager.io

1. Create a new file named `dns-issuer.yaml` with the following contents. Read the comments and modify the content as required.

```
---
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: test-dns
  namespace: default
spec:
  dnsNames:
    #NOTE: This only works if the DNS ClusterIssuer has permission to update
    "example.com" records
    - dns.example.com
  secretName: test-dns
  issuerRef:
    name: ceg-issuer-dns
    kind: ClusterIssuer
```

2. Create a new file named `dns-cert.yaml` with the following contents. Read the comments and modify the content as required.

```
---
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: test-dns
  namespace: default
spec:
  dnsNames:
    #NOTE: This only works if the DNS ClusterIssuer has permission to update
    "example.com" records
    - dns.example.com
  secretName: test-dns
  issuerRef:
    name: ceg-issuer-dns
    kind: ClusterIssuer
```

3. Create a new file named `http-issuer.yaml` with the following contents. Read the comments and modify the content as required.

```
---
apiVersion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: ceg-issuer-http
  namespace: cert-manager
spec:
  acme:
    # Uncomment the following line to allow insecure TLS connections.
    #skipTLSVerify: true
    # The ACME server URL
    server: https://cegserver.example.com/acme/tenant1/example_ca1/
privatessl_tls_client/directory
    # Email address used for ACME registration
    email: certmanager@example.com
    # Name of a secret used to store the ACME account private key
    privateKeySecretRef:
      name: ceg-acme-account-key-http
    # Enable the HTTP-01 challenge provider
    solvers:
      - http01:
          ingress:
            class: traefik
```

4. Create a new file named `echo.yaml` with the following contents.

```
---
apiVersion: v1
kind: Service
metadata:
  name: echo
spec:
  ports:
    - port: 80
      targetPort: 5678
    selector:
      app: echo
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: echo
spec:
  selector:
    matchLabels:
      app: echo
  replicas: 1
```

```
template:
  metadata:
    labels:
      app: echo
  spec:
    containers:
      - name: echo
        image: hashicorp/http-echo
        args:
          - "-text=echo"
        ports:
          - containerPort: 5678
```

5. Create a new file named `http-ingress.yaml` with the following contents. Read the comments and modify the content as required.

```
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: echo1-traefik-ingress
  namespace: default
  annotations:
    kubernetes.io/ingress.class: traefik
    cert-manager.io/cluster-issuer: ceg-issuer-http
spec:
  tls:
    - hosts:
        # Change the hostname here to the one you want a TLS Certificate for.
        # NOTE: CEG's must resolve the following hostname to cert-manager.io's IP
        Address.
        - echo1.example.com
      secretName: echo-tls
  rules:
    # The following host must match the host in the "tls" section a few lines up.
    - host: echo1.example.com
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: echo
                port:
                  number: 80
```

To apply the YAML files to Kubernetes and request certificates

1. Apply the files with the following commands.

```
kubectl apply -f dns-issuer.yaml
kubectl apply -f dns-cert.yaml
kubectl apply -f http-issuer.yaml
kubectl apply -f http-echo.yaml
kubectl apply -f http-ingress.yaml
```

2. After applying the files, Cert-manager.io has been configured to request two different certificates over two different ClusterIssuers, one certificate for DNS-01 validation, and one certificate for HTTP-01 validation. The certificates will automatically be requested by Cert-manager.io.
3. To view the status of all cert-manger.io ACMEv2 objects, enter the following command:

```
kubectl get
Issuers,ClusterIssuers,Certificates,CertificateRequests,Orders,Challenges --
all-namespaces
```

Integrating Microsoft Intune with Certificate Enrollment Gateway

Microsoft Intune is a Microsoft cloud-based service that manages mobile devices and applications. It integrates with other Enterprise Mobility and Security components for identity and access control and data protection.

In Microsoft Intune, you can add non-Microsoft Certification Authorities (CAs) and have these CAs issue and validate certificates using the Simple Certificate Enrollment Protocol (SCEP). SCEP is a PKI communication protocol that allows administrators to automatically and securely issue certificates to mobile devices that support SCEP.

Certificate Enrollment Gateway can receive SCEP requests with a CSR (certificate signing request) from Intune-supported devices and send the CSR to Intune for validation.

This section describes how to integrate Entrust Certificate Enrollment Gateway with the Microsoft Intune service.

- [How Certificate Enrollment Gateway works with Microsoft Intune](#)
- [Configuring Microsoft Intune for Certificate Enrollment Gateway](#)
- [Configuring Certificate Enrollment Gateway for Microsoft Intune](#)
- [Updating the client secret \(application key\) used by the integration](#)

How Certificate Enrollment Gateway works with Microsoft Intune

Traditional SCEP enrollment uses a static password for authentication. This static password is vulnerable to brute force attacks. Entrust worked with Microsoft Intune to co-develop a secure authentication mechanism for SCEP enrollment.

Certificate Enrollment Gateway can receive SCEP requests with a CSR (certificate signing request) from Windows clients, and send the CSR to Intune for validation.

Certificate Enrollment Gateway works with Intune as follows:

1. Microsoft Intune pushes a certificate profile and SCEP challenge to a Windows client.
2. The Windows client sends a SCEP request with a CSR from the Intune system to the Certificate Enrollment Gateway.
3. During the validation process of the SCEP request, Certificate Enrollment Gateway sends the CSR to the Intune service to validate the CSR.
4. If the CSR is valid:
 - a. Certificate Enrollment Gateway sends the CSR to Entrust CA Gateway, which forwards the CSR to the Managed CA for processing. A Gateway can issue digital certificates for one or more Certification Authorities (CAs). Each of these CAs is called a Managed CA.

- b. The Managed CA processes the request, and issues a certificate for the device.
- c. The Managed CA sends the certificate back to CA Gateway, which forwards the certificate back to Certificate Enrollment Gateway.
5. Certificate Enrollment Gateway returns the certificate to the Windows client. The Windows client will import the certificate into the client's certificate store.
6. Upon success or failure, Certificate Enrollment Gateway calls the Intune system to relay the status information.
7. With the certificate, the Windows client can access protected resources.

For more information about how non-Microsoft CAs work with Microsoft Intune, see the Microsoft documentation.

Configuring Microsoft Intune for Certificate Enrollment Gateway

To work with Certificate Enrollment Gateway, you must register and configure an application for Certificate Enrollment Gateway, import the issuing CAs as trusted third-party CAs, and configure a SCEP certificate profile in Microsoft Intune. When configuring Microsoft Intune, you must obtain and record information that Certificate Enrollment Gateway requires to connect to your Microsoft Intune instance.

- [Registering an application for Certificate Enrollment Gateway](#)
- [Generating a client secret for password-based authentication with Certificate Enrollment Gateway](#)
- [Generating and importing a TLS certificate for certificate-based authentication with Certificate Enrollment Gateway](#)
- [Adding API permissions to the CEG Service application](#)
- [Adding CAs to Microsoft Intune as trusted third-party CAs](#)
- [Configuring identity protection profiles for Windows Hello for Business](#)
- [Configuring SCEP certificate profiles](#)
- [Obtaining information required to configure Certificate Enrollment Gateway for Microsoft Intune](#)

Registering an application for Certificate Enrollment Gateway

For Certificate Enrollment Gateway to run custom challenge validation with Intune, you must register a new application in Azure Active Directory. This application will give delegated rights to Intune to validate SCEP requests.

To register an application for the CEG Service

1. Log in to the Microsoft Azure portal.
2. Under **Azure services**, click **Azure Active Directory**.
3. Click **App Registrations**.
4. Click **Register an application**.
The **Register an application** page appears.
5. For **Name**, enter a unique application name. For example, `Entrust SCEP Service`.
6. For **Supported account types**, select **Accounts in any organizational directory**.
7. Do not provide any values for **Redirect URI**. Intune does not need to redirect back to Certificate Enrollment Gateway after issuing the certificate.
8. Click **Register**.
After registering the application, an **Overview** page appears for the application.
9. Record the **Application (client) ID** value. For example:

```
00000000-0000-0000-0000-000000000000
```

You need this value later to configure Certificate Enrollment Gateway for Microsoft Intune.

10. Record your Tenant ID. You need this value later to configure Certificate Enrollment Gateway for Microsoft Intune. The Tenant ID is the domain text after the @ sign in to your account. For example, if your account is `admin@test.example.com`, then your tenant ID is `test.example.com`.

Generating a client secret for password-based authentication with Certificate Enrollment Gateway

Certificate Enrollment Gateway can authenticate to Microsoft Intune using one of the following authentication methods:

- Password-based authentication: Certificate Enrollment Gateway authenticates to Microsoft Intune using an application key (also called a client secret) generated in Microsoft Intune.
- Certificate-based authentication: Certificate Enrollment Gateway authenticates to Microsoft Intune using a trusted certificate. The certificate must be imported into Microsoft Intune.

This section describes how to generate a client secret for the application you registered earlier in Microsoft Intune. Certificate Enrollment Gateway can then use this secret to authenticate to Intune.

To generate a client secret

1. Log in to the Microsoft Azure portal.
2. Under **Azure services**, click **Azure Active Directory**.
3. Click **App Registrations**.
4. Select the application you created earlier for the CEG Service.
5. Click **Certificates & secrets**.
6. Click **New client secret**.
The **Add a client secret** page appears.
7. For **Description**, enter a description of the client secret.
8. For **Expires**, select a lifetime for the client secret.
9. Click **Add**.
The client secret is displayed under the **Client secrets** pane.
10. Record the client secret. For example:

```
abcdefghijklmnopqrstuvwxyz123456
```

The client secret is also known as the Application Key. You need this value later to configure Certificate Enrollment Gateway for Microsoft Intune.

Generating and importing a TLS certificate for certificate-based authentication with Certificate Enrollment Gateway

Certificate Enrollment Gateway can authenticate to Microsoft Intune using one of the following authentication methods:

- Password-based authentication: Certificate Enrollment Gateway authenticates to Microsoft Intune using an application key (also called an authentication key or client secret) generated in Microsoft Intune.
- Certificate-based authentication: Certificate Enrollment Gateway authenticates to Microsoft Intune using a trusted certificate. The certificate must be imported into Microsoft Intune.

You cannot generate a TLS certificate using Microsoft Intune. You must generate a certificate using another tool, and then import the certificate into Microsoft Intune. Microsoft Intune and Certificate Enrollment Gateway must use the same certificate for authentication.

This section provides instructions about how you can use the TLS bootstrapping feature of Certificate Enrollment Gateway to generate a TLS certificate for certificate-based authentication. You can then import this certificate into Microsoft Intune.

To generate a TLS certificate file for certificate-based authentication using TLS bootstrapping

1. Log in to the server hosting the Certificate Enrollment Gateway.
2. Generate a TLS certificate (`tls.crt`) using the TLS bootstrapping feature of Certificate Enrollment Gateway. For instructions about using the TLS bootstrapping feature, see the *Certificate Enrollment Gateway Deployment Guide*. The value of the distinguished name (DN) does not need to be a fully qualified domain

name (FQDN).

You can now import the TLS certificate into Microsoft Intune as described in the following procedure. You must also complete the following steps to convert the TLS certificate (`tls.crt`) and associated private key (`tls.key`) into a PKCS #12 (P12) file. Certificate Enrollment Gateway requires a P12 file for certificate-based authentication to Microsoft Intune.

3. To convert the PEM-formatted TLS certificate and private into a P12 file (AppKey.p12) for Certificate Enrollment Gateway, enter the following command:

```
openssl pkcs12 -export -out AppKey.p12 -in tls.crt -inkey tls.key
```

Certificate Enrollment Gateway requires a P12 file for certificate-based authentication to Microsoft Intune.

4. When prompted, enter a password for the P12 file.
5. Copy the P12 file you just created (`AppKey.p12`) file into the Certificate Enrollment Gateway configuration directory, the same directory hosting the `config.yml` file.
6. Reload the Certificate Enrollment Gateway package to apply the changes.

To import the TLS certificate into Microsoft Intune

1. Log in to the Microsoft Azure portal.
2. Under **Azure services**, click **Azure Active Directory**.
3. Click **App Registrations**.
4. Select the application you created earlier for the CEG Service.
5. Click **Certificates & secrets**.
6. Click **Upload certificate**.
7. Select the TLS certificate
8. Click **Add**.

Information about the certificate is displayed under the **Certificates** pane.

Adding API permissions to the CEG Service application

After registering an application for the CEG Service, you must add the following API permissions to the application.

API Permission category	Permissions
Intune	scep_challenge_provider (SCEP challenge validation)
Microsoft Graph	Application.Read.All (Read all applications)

You must also grant administrative consent for these permissions to the application.

To add required API permissions to the CEG Service application


1. Log in to the Microsoft Azure portal.
2. Under Azure services, click Azure Active Directory.
3. Click App Registrations.
4. Select the application you created earlier for the CEG Service.
5. Click **API permissions**.
6. To add the required Intune API permissions:
 - a. Click **Add a permission**. The Request API permissions page appears.
 - b. Click **Microsoft APIs**.
 - c. Click **Intune**.
 - d. Select **Application permissions**.

- e. Select the following Intune application permissions:
 - Select **scep_challenge_provider** (SCEP challenge validation).
- f. Click **Add permissions**.
7. To add the required Microsoft Graph API permissions:
 - a. Click **Add a permission**. The Request API permissions page appears.
 - b. Click **Microsoft APIs**.
 - c. Click **Microsoft Graph**.
 - d. Select **Application permissions**.
 - e. Select the following permissions:
 - Select **Application.Read.All** (Read all applications).
 - f. Click **Add permissions**.
8. When prompted, click **Yes** to confirm consent.

Adding CAs to Microsoft Intune as trusted third-party CAs

CA Gateway can issue digital certificates for one or more Certification Authorities (CAs). Each of these CAs is called a Managed CA.

For each Managed CA in CA Gateway that will issue certificates, you must add the Managed CA to Microsoft Intune as a trusted third-party CA. To add a trusted CA to Microsoft Intune, you must create a trusted certificate profile in Microsoft Intune. When creating a trusted certificate profile, you will import the CA certificate of the Managed CA.

 If the CA is an intermediate CA (also called a subordinate CA) and not the root CA, you must add each CA certificate in the certificate chain as a trusted third-party CA.

Each Managed CA will also act as a root of trust for one or more SCEP certificate profiles (see [Configuring SCEP certificate profiles](#)).

To add a CA to Microsoft Intune as a trusted third-party CA

1. Obtain the CA certificate of the Managed CA.
 - If the Managed CA is an intermediate CA (also called a subordinate CA) and not the root CA, you must add each CA certificate in the certificate chain as a trusted third-party CA.
2. Log in to Intune.
3. Click **Devices**.
4. Click **Configuration profile**.
5. Click **Create Profile**.
 - The **Create profile** page appears.
6. For **Platform**, select a device platform that will use the trusted certificate.
7. For **Profile type**, select **Trusted certificate**.
8. For **Name**, enter a unique name to identify the trusted certificate profile.
9. For **Description**, enter a description for the trusted certificate profile.
10. In the **Trusted certificate** pane, select the CA certificate you obtained earlier, then click **OK**.
11. Click **Create** to create the certificate profile.
12. Click **Assignments**.
13. For **Include**, select the Azure Active Directory groups you want to include with the certificate profile.
14. For **Exclude**, select the Azure Active Directory groups you want to exclude from the certificate profile.
15. Click **Save**.

Configuring identity protection profiles for Windows Hello for Business

Windows Hello for Business is a method for signing in to Windows devices by replacing passwords, smart cards, and virtual smart cards. To support Windows Hello for Business with Microsoft Intune, you must create one or more

identity protection profiles. Each identity protection profile will enable Windows Hello for Business for devices and users, and configure various PIN and authentication settings.

To configure an identity protection profile for Windows Hello for Business

1. Log in to the Microsoft Azure portal.
2. Log in to Intune.
3. Click **Devices**.
4. Under **Policy**, click **Configuration profiles**.
5. Click **Create profile**.
The **Create profile** page appears.
6. For **Platform**, select **Windows 10 and later**.
7. For **Profile type**, select **Templates**.
8. Search or select **Identity protection**, then click **Create**.
9. For **Name**, enter a unique name to identify the identity protection profile.
10. For **Description**, enter a description for the identity protection profile.
11. Scroll down to the **Identity protection** pane.
12. Under **Configuration settings**, configure the following settings:
 - a. For **Configure Windows Hello for Business**, select **Enabled**.
 - b. For **Minimum PIN length**, enter the minimum PIN length.
 - c. For **Maximum PIN length**, enter the maximum PIN length.
 - d. For **Lowercase letters in PIN**, select whether lowercase letters are not allowed, allowed but not required, or required in a PIN.
 - e. For **Uppercase letters in PIN**, select whether uppercase letters are not allowed, allowed but not required, or required in a PIN.
 - f. For **Special characters in PIN**, select whether special characters (non-alphanumeric characters) are not allowed, allowed but not required, or required in a PIN
 - g. For **PIN expiration (days)**, select the number of days a PIN can be used before it expires. Users must change their PIN after the configured number of days.
 - h. For **Remember PIN history**, select how many previous PINs are remembered. When users change their PIN, they cannot reuse this number of previously-used PINs.
 - i. For **Enable PIN recovery**, select **Enable** to allow users to recover their PIN using the Windows Hello for Business PIN recovery service.
 - j. For **Use a Trusted Platform Module (TPM)**, select **Enable** to allow only devices with an accessible TPM to provision Windows Hello for Business.
 - k. For **Allow biometric authentication**, select **Enable** to allow Windows Hello for Business to authenticate using biometric authentication.
 - l. For **Use enhanced anti-spoofing, when available**, select **Enable** to use anti-spoofing features on the device when available.
 - m. For **Certificate for on-premise resources**, select **Enable** to allow Windows Hello for Business to use certificates for authentication to on-premises resources.
 - n. For **Use security keys for sign-on**, select **Enable** to allow users to sign in with Windows Hello security key.
13. Click **Next**.
14. Under **Assignments**:
 - a. For **Include**, select the Azure Active Directory groups you want to include with the identity protection profile.
 - b. For **Exclude**, select the Azure Active Directory groups you want to exclude from the identity protection profile.
15. Click **Next**.
16. Under **Applicability Rules**:
 - a. If required, configure any rules to work with your environment.
 - b. Click **Next**.
17. Under **Review + create**:

- a. Review the identity protection profile. Change any settings if required.
- b. Click **Create** to create the identity protection profile.

Configuring SCEP certificate profiles


To issue certificates with SCEP, you must create one or more SCEP certificate profiles in Microsoft Intune. A SCEP certificate profile defines various properties of a certificate issued to users or devices over SCEP, including the subject name format and subject alternative name extensions.

When configuring a SCEP certificate profile, you must provide the URL to the CEG Intune-SCEP Enrollment Service in Certificate Enrollment Gateway. Some other settings may also require specific values to work with work with Windows Hello for Business.

To configure a SCEP certificate profile

1. Log in to the Microsoft Azure portal.
2. Log in to Intune.
3. Click **Devices**.
4. Under **Policy**, click **Device configuration**.
5. Click **Create profile**.
The **Create profile** page appears.
6. For **Name**, enter a unique name to identify the SCEP certificate profile.
7. For **Description**, enter a description for the SCEP certificate profile.
8. For **Platform**, select a device platform that will use the trusted certificate. To work with Windows Hello for Business, select **Windows 10 and later**.
9. For **Profile type**, select **SCEP certificate**.
10. In the **SCEP Certificate** pane, provide the information that will be included in the CSR (certificate signing request):
 - a. For **Certificate type**, select the type of certificate that will be issued.
 - b. For **Subject name format**, enter the subject name format. To work with Windows Hello for Business, enter `CN={{UserPrincipalName}}`.
 - c. For **Subject alternative name**, select the subject alternative name extensions that will be included in the CSR.
 - d. For **Certificate validity period**, select a lifetime for the certificate—for example, 1 year.
The validity period must not exceed the maximum validity period that is permitted by the Managed CA. For Entrust PKI as a Service, the maximum validity period is 3 years. For an on-premises Security Manager CA, see the Security Manager documentation for information about viewing or configuring the validity period.
For iOS devices, the validity period will be defined by the Managed CA. iOS devices will ignore the validity period defined in the SCEP certificate profile. For Entrust PKI as a Service, the validity period for certificates issued to iOS devices will be 3 years.
 - e. For **Key storage provider (KSP)**, select a key storage provider. To work with Windows Hello for Business, select **Enroll to Windows Hello for Business, otherwise fail**.
 - f. For **Key usage**, select a key usage for the certificate.
Certificate Enrollment Gateway will ignore the key usage value set in the SCEP certificate profile. The certificate profile defined in CA Gateway controls the key usage of the certificate.
The profile ID of the certificate profile is part of the SCEP Server URL. Ensure that the key usage defined in the SCEP certificate profile is compatible with the certificate profile defined in CA Gateway.
 - g. For **Key size (bits)**, select the size of the key in bits—for example, 2048.
 - h. For **Root Certificate**, select a trusted root certificate profile that you created previously.
 - i. For **Hash Algorithm**, select SHA-2. Certificate Enrollment Gateway supports only SHA-2 as the hash algorithm.
 - j. For **Extended key usage**, add values for the certificate's intended purpose.

- In most cases, the certificate requires Client Authentication so that the user or device can authenticate to a server.
 - To work with Windows Hello for Business, add Smart Card Logon (OID 1.3.6.1.4.1.311.20.2.2).
- k. For **Renewal threshold (%)**, enter the percentage of the certificate lifetime that remains before the certificate should be renewed.
- l. For **SCEP Server URLs**, enter one of the following Certificate Enrollment Gateway URLs:

 The following Intune-SCEP enrollment URL requires a trailing forward slash (/). To support macOS devices, the URL must start with `http` instead of `https`.

```
https://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/intune/  
http://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/intune/
```

See below for a description of each field.

- `<CEG-server>`
- `<tenant-ID>`
- `<CA-ID>`
- `<profile-ID>`

For example:

```
https://cegserver.example.com/scep/tenant1/example-ca1/intune-digital-  
signature/intune/  
http://cegserver.example.com/scep/tenant1/example-ca1/intune-digital-  
signature/intune/
```

- m. Enter values for any other settings as required.
11. Click **OK**.
 12. Click **Create** to create the SCEP certificate profile.
 13. Click **Assignments**.
 14. For **Include**, select the Azure Active Directory groups you want to include with the certificate profile.
 15. For **Exclude**, select the Azure Active Directory groups you want to exclude from the certificate profile.
 16. Click **Save**.

`<CEG-server>`

The hostname or IP address of the Certificate Enrollment Gateway server.

`<tenant-ID>`

The unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.

`<CA-ID>`

The CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the SCEP client.

`<profile-ID>`

The profile ID defined in CA Gateway that defines the certificate type issued to the SCEP client. For Entrust PKI as a Service, the profile ID is one of the following:

- `intune-digital-signature-key-encipherment`
- `intune-digital-signature`
- `intune-key-encipherment`

- intune-non-repudiation

Obtaining information required to configure Certificate Enrollment Gateway for Microsoft Intune

To integrate with Microsoft Intune, Certificate Enrollment Gateway requires the following Microsoft Intune information.

- Application (client) ID. For example:

```
00000000-0000-0000-0000-000000000000
```

- Tenant ID.
The Tenant ID is the domain text after the @ sign in to your account. For example, if your account is `admin@test.example.com`, then your tenant ID is `test.example.com`.
- Application Key (Authentication Key), if you generated a client secret in Microsoft Intune. The application key is the client secret you generated earlier. For example:

```
abcdefghijklmnopqrstuvwxyz123456
```

If you will use certificate-based authentication, you do not need an application key.

You may have already obtained and recorded this information when you configured Microsoft Intune for Certificate Enrollment Gateway.

Configuring Certificate Enrollment Gateway for Microsoft Intune

To configure Certificate Enrollment Gateway for Microsoft Intune, you must configure the Intune connection settings in Certificate Enrollment Gateway. You can edit the Intune connection settings using the Management Console interface.

To configure Certificate Enrollment Gateway for Microsoft Intune

1. Log in into the Management Console as explained in [Logging into the Management Console](#).
2. In the **Certificate Enrollment Gateway** pane, click **Manage Solution**.
A **Certificate Enrollment Gateway** page appears.
3. In the left navigation bar, click **Configuration**.
A **Product Configuration** pane appears.
4. Turn on **Enable Advanced Configuration**.
5. Click **Next**.
6. For all SCEP-related protocols (SCEP, MDM-SCEP, and Intune-SCEP), Certificate Enrollment Gateway uses RA certificates to sign and encrypt SCEP PKI messages. For an on-premises CA, you must specify one or more profiles that are defined in CA Gateway used to issue RA certificates.
 - a. Click the [CAGW](#) tab.
 - b. Configure the [RA Certificate Profile IDs](#) setting.
7. Click the [Intune](#) tab and configure the following settings.
 - [Enable InTune-SCEP](#)
 - [InTune Revocation Cron Job](#)
 - [InTune-SCEP Enrollment Service Configurations](#)
8. Configure any other settings if required.
9. After configuring the settings, click **Validate** to validate the settings.
If any configuration errors are detected, correct the errors then click **Validate** again.
10. After validating the configuration settings, click **Next**.
Entrust PKI Hub uploads the configuration and any attached files, such as P12 credentials.

11. In the **Product Deployment Status** pane, re-deploy Certificate Enrollment Gateway with the updated configuration file by clicking **Deploy**.
A dialog box appears, prompting you to confirm the operation. Click **Yes** to confirm the operation and deploy the Certificate Enrollment Gateway solution.

Updating the client secret (application key) used by the integration

Certificate Enrollment Gateway authenticates to Microsoft Intune using a client secret (also called an application key). Client secrets are created in Intune, and will expire after a configurable amount of time, such as two years. If the client secret used by Certificate Enrollment Gateway expires, Certificate Enrollment Gateway cannot authenticate to Intune, and therefore can no longer issue certificates to SCEP clients.

Intune allows you to create multiple client secrets. Before a client secret used by Certificate Enrollment Gateway expires, you should create a new client secret in Intune, and then change the application key in Certificate Enrollment Gateway. Certificate Enrollment Gateway can then use the updated application key to authenticate to Intune.

To generate a new client secret

1. Log in to the Microsoft Azure portal.
2. Under **Azure services**, click **Azure Active Directory**.
3. Click **App Registrations**.
4. Select the application you created earlier for the CEG Service.
5. Click **Certificates & secrets**.
6. Click **New client secret**. The **Add a client secret** page appears.
7. For **Description**, enter a description of the client secret.
8. For **Expires**, select a lifetime for the client secret.
9. Click **Add**. The client secret is displayed under the **Client secrets** pane.
10. Record the client secret. For example:

```
abcdefghijklmnopqrstuvwxyz123456
```

The client secret is also known as the Application Key. You need this value to update the application key used by Certificate Enrollment Gateway to connect to Microsoft Intune.

To update Certificate Enrollment Gateway to use the new application key

1. Log in into the Management Console as explained in [Logging into the Management Console](#).
2. In the **Certificate Enrollment Gateway** pane, click **Manage Solution**.
A **Certificate Enrollment Gateway** page appears.
3. In the left navigation bar, click **Configuration**.
A **Product Configuration** pane appears.
4. Turn on **Enable Advanced Configuration**.
5. Click **Next**.
6. For the Intune settings (see [Intune](#)), update each **Registered Azure Application Key (Client Secret)** setting to use the new application key (client setting) value.
7. After configuring the settings, click **Validate** to validate the settings.
If any configuration errors are detected, correct the errors then click **Validate** again.
8. After validating the configuration settings, click **Next**.
Entrust PKI Hub uploads the configuration and any attached files, such as P12 credentials.
9. In the **Product Deployment Status** pane, re-deploy Certificate Enrollment Gateway with the updated configuration file by clicking **Deploy**.
A dialog box appears, prompting you to confirm the operation. Click **Yes** to confirm the operation and deploy the Certificate Enrollment Gateway solution.

Integrating SCEP clients with Certificate Enrollment Gateway

This section explains how to integrate SCEP clients with the Certificate Enrollment Gateway service.

- [Configuring Certificate Enrollment Gateway for SCEP enrollment](#)
- [Configuring SCEP clients for enrollment with Certificate Enrollment Gateway](#)
- [SCEP client examples](#)

Configuring Certificate Enrollment Gateway for SCEP enrollment


To configure Certificate Enrollment Gateway for SCEP enrollment, you must configure the SCEP enrollment settings in Certificate Enrollment Gateway. You can edit the SCEP enrollment settings using the Management Console interface.

To configure Certificate Enrollment Gateway for SCEP enrollment

1. Log in into the Management Console as explained in [Logging into the Management Console](#).
2. In the **Certificate Enrollment Gateway** pane, click **Manage Solution**.
A **Certificate Enrollment Gateway** page appears.
3. In the left navigation bar, click **Configuration**.
A **Product Configuration** pane appears.
4. Turn on **Enable Advanced Configuration**.
5. Click **Next**.
6. For all SCEP-related protocols (SCEP, MDM-SCEP, and Intune-SCEP), Certificate Enrollment Gateway uses RA certificates to sign and encrypt SCEP PKI messages. For an on-premises CA, you must specify one or more profiles that are defined in CA Gateway used to issue RA certificates.
 - a. Click the [CAGW](#) tab.
 - b. Configure the [RA Certificate Profile IDs](#) setting.
7. Click the [SCEP](#) tab and configure the following settings.
 - [Enable SCEP](#)
 - [SCEP Enrollment Service Configurations](#)
8. Configure any other settings if required.
9. After configuring the settings, click **Validate** to validate the settings.
If any configuration errors are detected, correct the errors then click **Validate** again.
10. After validating the configuration settings, click **Next**.
Entrust PKI Hub uploads the configuration and any attached files, such as P12 credentials.
11. In the **Product Deployment Status** pane, re-deploy Certificate Enrollment Gateway with the updated configuration file by clicking **Deploy**.
A dialog box appears, prompting you to confirm the operation. Click **Yes** to confirm the operation and deploy the Certificate Enrollment Gateway solution.

Configuring SCEP clients for enrollment with Certificate Enrollment Gateway

SCEP clients must use one of the following URLs to communicate with Certificate Enrollment Gateway:

-  The following SCEP enrollment URL requires a trailing forward-slash (/). To support macOS (Apple) devices, the URL must start with `http` instead of `https`.

```
http://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/
https://<CEG-server>/scep/<tenant-ID>/<CA-ID>/<profile-ID>/
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.

- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the SCEP client.
- `<profile-ID>` is the profile ID defined in CA Gateway that defines the certificate type issued to the SCEP client. For Entrust PKI as a Service, the profile ID is one of the following:
 - `scep-digital-signature-key-encipherment`
 - `scep-digital-signature`
 - `scep-key-encipherment`
 - `scep-non-repudiation`

For example:

```
http://cegservers.example.com/scep/tenant1/example_ca1/scep-digital-signature/  
https://cegservers.example.com/scep/tenant1/example_ca1/scep-digital-signature/
```

Some SCEP clients will append an additional parameter to all SCEP URLs. For these clients, you must append `nop/` to the SCEP URL. For example:

```
http://cegservers.example.com/scep/tenant1/example_ca1/scep-digital-signature/nop/  
https://cegservers.example.com/scep/tenant1/example_ca1/scep-digital-signature/nop/
```


SCEP client examples

The following topics provide some examples about how to configure some common clients for SCEP enrollment. For complete information about using these SCEP clients, see the documentation for the SCEP client.

- [Google ChromeOS example](#)

Google ChromeOS example

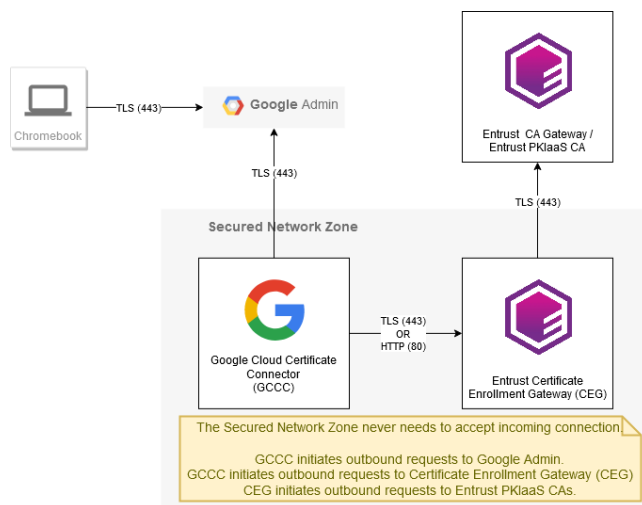
Google ChromeOS is a cloud-first operating system for Google Chromebooks and other Google devices. Google devices that use ChromeOS can request certificates from Certificate Enrollment Gateway using the SCEP protocol. SCEP enrollment for ChromeOS is controlled using Google Admin, a Web-based interface for managing users and groups for an organization.

 For issues related to Google applications and products, contact Google for support. For issues related to Certificate Enrollment Gateway and the SCEP service, contact Entrust Customer Support.

- [ChromeOS integration requirements](#)
- [Configuring Google Admin for SCEP enrollment](#)
- [Downloading and installing the Google Cloud Certificate Connector](#)
- [Testing SCEP enrollment with ChromeOS](#)
- [Troubleshooting SCEP enrollments with ChromeOS](#)

ChromeOS integration requirements

The following diagram illustrates the architecture and components of a Chromebook (with ChromeOS) that can enroll for a certificate over SCEP with Entrust Certificate Enrollment Gateway.



Google Admin requirements:

- Google Admin requires either the Chrome Enterprise Upgrade or the Chrome Education Upgrade.
- Google Admin requires the CA certificate chain (from the root CA certificate to the issuing CA certificate) for the on-premises Managed CA or Entrust PKI as a Service (PKIaaS).

Google Cloud Certificate Connector requirements:

- The Google Cloud Certificate Connector must be installed on Domain-joined Windows server.
- The Google Cloud Certificate Connector requires outbound network connectivity to Google Admin.
- The Google Cloud Certificate Connector requires outbound network connectivity to Entrust Certificate Enrollment Gateway.

Entrust Certificate Enrollment Gateway requirements:

i Static challenge passwords are not secure. To increase security, it is recommended that you configure the firewall on the Certificate Enrollment Gateway server to limit incoming traffic for the SCEP service to only the Google Cloud Certificate Connector.

- The SCEP service must be configured with a static challenge password.
- Certificate Enrollment Gateway requires inbound connectivity from the Google Cloud Certificate Connector.

ChromeOS requirements:

- ChromeOS must be enrolled with Google Admin using the **Enterprise enrollment** option. See the Google documentation for instructions about enrolling a device using the **Enterprise enrollment** option (<https://support.google.com/chrome/a/answer/1360534>).
- ChromeOS requires outbound connectivity to Google Admin without the interference of SSL decryption.

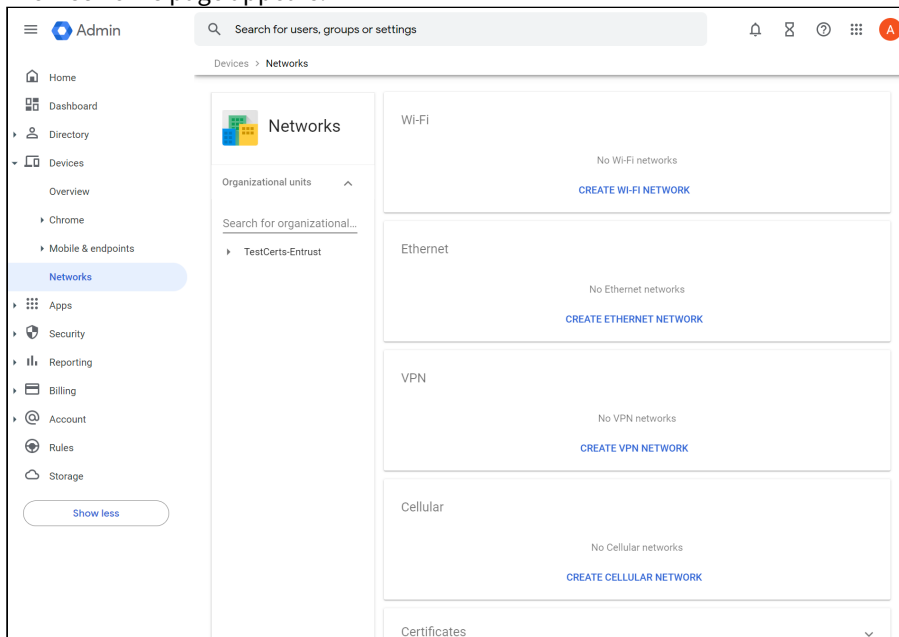
Configuring Google Admin for SCEP enrollment

SCEP enrollment for ChromeOS is controlled using Google Admin, a Web-based interface for managing users and groups for an organization.

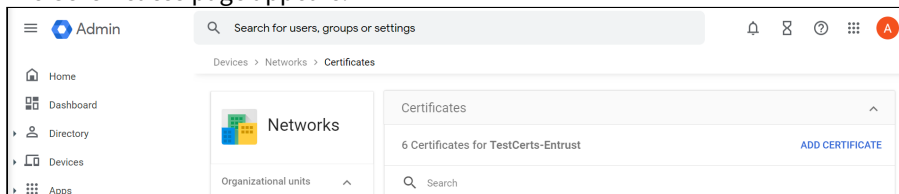
To configure Google Admin for SCEP enrollment

1. Log in to Google Admin (<https://admin.google.com>).

2. Navigate to **Devices > Networks**.
The **Networks** page appears.

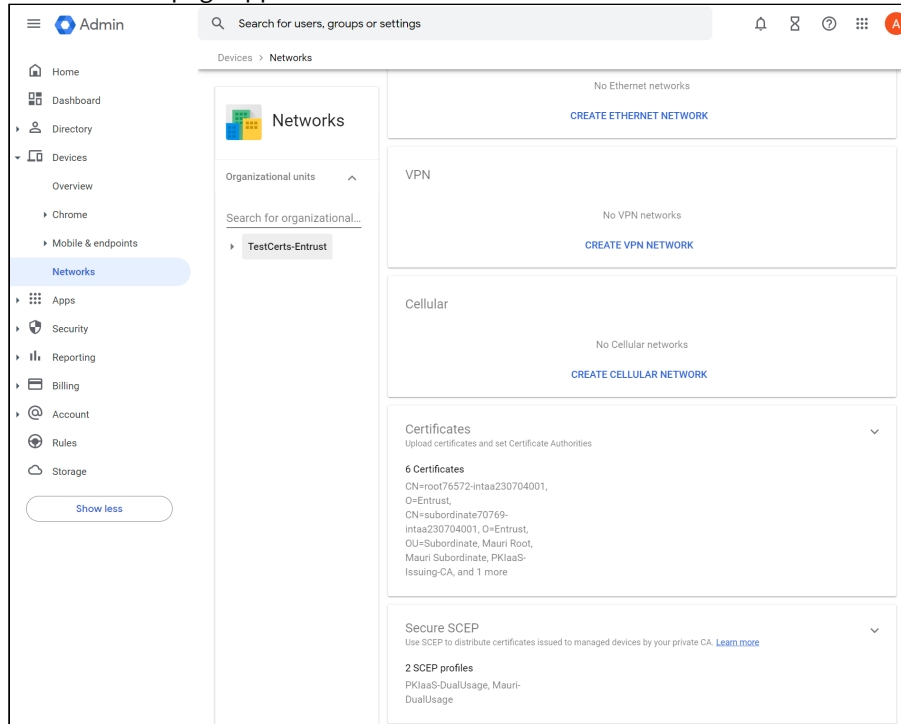


3. Click on the **Certificates** pane.
The **Certificates** page appears.

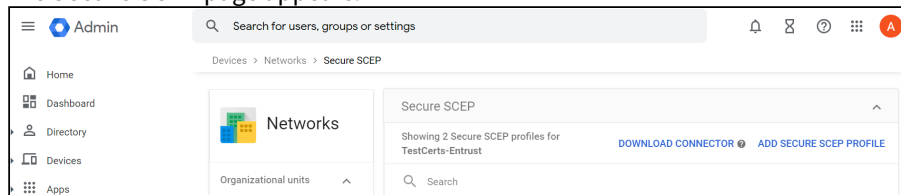


4. Add the entire CA certificate chain (from the root CA to the issuing CA) for the Managed CA. To add a CA certificate:
 - a. Click **Add Certificate**. The **Add Certificate** page appears.
 - b. In the **Name** field, enter a unique friendly name for the CA certificate,
 - c. Click **Upload** and then select the CA certificate you want to upload.
 - d. Select **Chromebook**.
 - e. Click **Add**.
5. Navigate to **Devices > Networks**.

6. The **Networks** page appears.



7. Click on the **Secure SCEP** pane.
The **Secure SCEP** page appears.



8. Create a SCEP profile:
- Click **ADD SECURE SCEP PROFILE**. The **Edit Secure SCEP** page appears.
 - For **Device platforms**, select the Chromebook platforms that will enroll for a certificate over SCEP:
 - Select **Chromebook (user)** for Chromebook users.
 - Select **Chromebook (device)** for Chromebook devices.
 - For **SCEP profile name**, enter a unique name for the SCEP profile.
 - For **Subject name format**, define the desired Subject Name format.

i The key usages you specify in the SCEP profile must match the certificate profile used in the SCEP server URL (SCEP enrollment URL). For example, if both **Key encipherment** and **Signing** are selected, then the certificate profile used in the SCEP server URL must include both encryption and signing key usages. For example in Entrust PKI as a Service (PKIaaS) deployments, if both **Key encipherment** and **Signing** are selected, then the certificate profile used in the SCEP server URL must be `scep-digital-signature-key-encipherment`.

- For **Key Usage**, select each key usage that will be included in the issued certificates.
- For **Key size (bits)**, select a key size for the issued certificates.
- For **Security**, select the security level (attestation requirement) for the issued certificates.
- In the **SCEP server URL** field, enter the CEG SCEP Service URL.

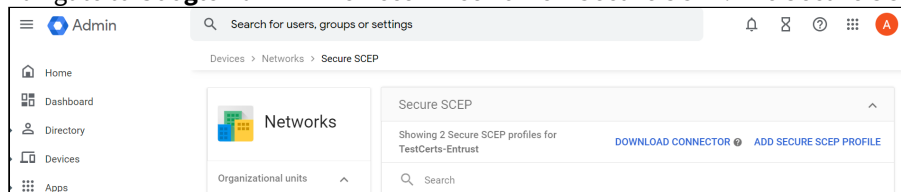
- i. For **Certificate validity period (years)**, enter a lifetime (in years).
The certificate validity period will be ignored for SCEP enrollment with Certificate Enrollment Gateway. The lifetime for issued certificates is controlled by the issuing CA. For Entrust PKI as a Service, the default certificate lifetime is 1 year.
- j. For **Renew within days**, enter the renewal period (in days) for certificates. The renewal period is the number of days before a certificate expires. Certificates that will expire within this period will be renewed.
- k. For **Extended key usage**, select the extended key usage extensions that will be included in the issued certificates.
- l. For **Challenge type**, select **Static** and then enter the challenge password defined in the CEG SCEP Service.
- m. For **Certificate Authority**, select the issuing CA certificate (Managed CA certificate) that you uploaded earlier.
- n. (Optional.) For **Network type**, select the network types that will use the SCEP profile.
- o. Click **SAVE**.

Downloading and installing the Google Cloud Certificate Connector

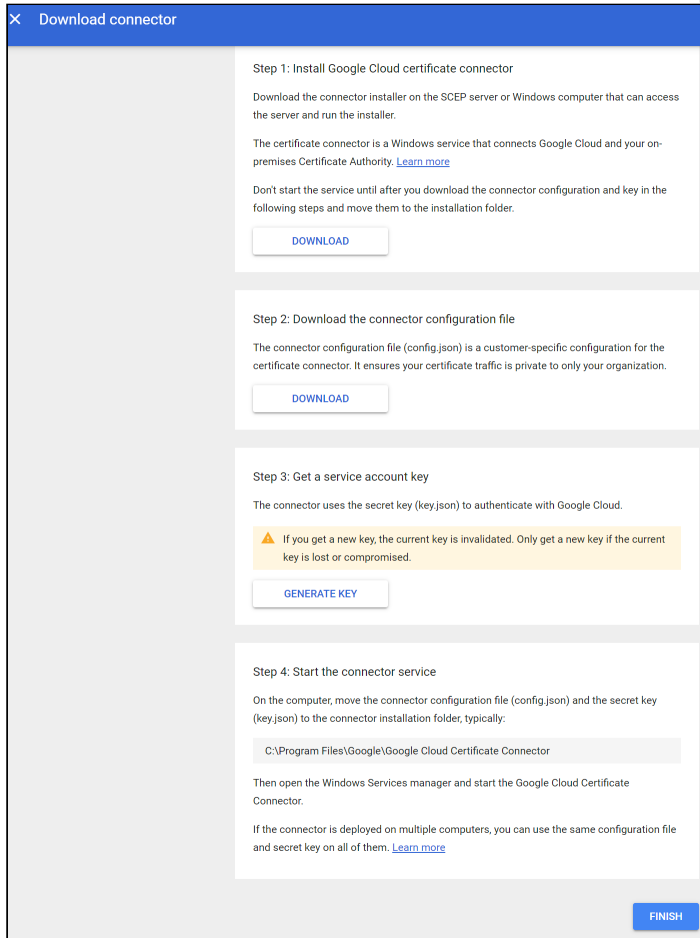
The Google Cloud Certificate Connector is a Windows service that securely distributes certificates and authentication keys from a SCEP server to users' devices. The Google Cloud Certificate Connector must be installed on Domain-joined Windows server.

To download and install the Google Cloud Certificate Connector

1. Log in to <https://admin.google.com>
2. Navigate to **Google Admin > Devices > Networks > Secure SCEP**. The **Secure SCEP** page appears.



3. Click on **DOWNLOAD CONNECTOR**. The **Download connector** page appears.



4. In the **Step 1** pane, click **DOWNLOAD** to download the installer for Google Cloud Certificate Connector.
5. In the **Step 2** pane, click **DOWNLOAD** to download the configuration file (`config.json`) for Google Cloud Certificate Connector.
6. In the **Step 3** pane, click **GENERATE KEY** to generate and download a key file (`key.json`) for Google Cloud Certificate Connector.
7. Copy the installer, configuration file (`config.json`), and key file (`key.json`) to a Domain-joined Windows server.
8. On the Domain-joined Windows server, run the installer to install Google Cloud Certificate Connector.
9. Copy the `config.json` and `key.json` files into the Google Cloud Certificate Connector installation folder.
10. Start Google Cloud Certificate Connector. Open the Services administrative tool (select **Start > Windows Administrative Tools > Services**), and start **Google Cloud Certificate Connector**.

Testing SCEP enrollment with ChromeOS

To test SCEP enrollment with ChromeOS:

1. Enroll a Chromebook into Google Admin using the Enterprise Enrollment option.
2. Log in to the Chromebook.
3. Open Google Chrome.
4. Browse to `chrome://policy`.

5. Ensure that a device or user policy for client certificates exists (either **RequiredClientCertificateForDevice** or **RequiredClientCertificateForUser**).
 - The **RequiredClientCertificateForDevice** policy corresponds to the **Chromebook (device)** device platform in the SCEP profile.
 - The **RequiredClientCertificateForUser** policy corresponds to the **Chromebook (user)** device platform in the SCEP profile.
6. Browse to <chrome://certificate-manager>.
7. The name of the SCEP certificate should appear in the list.
8. Click on the SCEP certificate to view details.
9. Validate that no errors are displayed for the certificate.

Troubleshooting SCEP enrollments with ChromeOS

If errors occur during SCEP enrollment with ChromeOS, check the Google Cloud Certificate Connector and Entrust Certificate Enrollment Gateway logs for more information.

For information about viewing the Certificate Enrollment Gateway logs, see the Entrust PKI Hub documentation.

To view Google Cloud Certificate Connector logs

1. On the Domain-joined Windows server hosting the Google Cloud Certificate Connector, open the Event Viewer (select **Start > Windows Administrative Tools > Event Viewer**).
2. Analyze events from **GoogleCloudCertificateConnector**.

Integrating MDM and MDM-SCEP clients with Certificate Enrollment Gateway

This section explains how to integrate Mobile Device Management (MDM) products and MDM-SCEP clients with the Certificate Enrollment Gateway service.

- [Configuring a Mobile Device Management product for enrollment with Certificate Enrollment Gateway](#)
- [Configuring MDM-SCEP clients for enrollment with Certificate Enrollment Gateway](#)
- [Configuring Certificate Enrollment Gateway for MDMWS and MDM-SCEP enrollment](#)

Configuring a Mobile Device Management product for enrollment with Certificate Enrollment Gateway

This section describes the information required to configure Mobile Device Management (MDM) products to enroll for a certificate using Certificate Enrollment Gateway. For information about using your MDM product, see your MDM product documentation.

- [Supported MDM authentication methods](#)
- [Adding the CA certificate chain to the MDM product](#)
- [Issuing a signing certificate to the MDM product](#)
- [Enrollment URL for MDMWS clients](#)

Supported MDM authentication methods

Certificate Enrollment Gateway supports username and password authentication to authenticate with MDM products. You must configure at least one username and password credential in the MDM product.

All username and password credentials that Certificate Enrollment Gateway will use to authenticate to the MDM product must be specified in the Certificate Enrollment Gateway MDMWS configuration settings (see [MDMWS](#)).

Adding the CA certificate chain to the MDM product

MDM products must trust the CA certificate chain for the Certificate Enrollment Gateway TLS certificate. If the MDM product does not trust the CA certificate chain, the MDM clients will fail to establish a secure TLS connection to

Certificate Enrollment Gateway. For instructions about adding certificates to your MDM product, see your MDM product documentation.

Issuing a signing certificate to the MDM product

Some MDM products may require a signing certificate to sign data being delivered to MDM devices. For instructions about adding certificates to your MDM product, see your MDM product documentation.

Enrollment URL for MDMWS clients

Mobile Device Management products must use the following URL to communicate with Certificate Enrollment Gateway:

```
https://<CEG-server>/mdm/services/<tenant-ID>
```

Where:


- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.

For example:

```
https://cegserver.example.com/mdm/services/tenant1
```

Configuring MDM-SCEP clients for enrollment with Certificate Enrollment Gateway

MDM-SCEP clients must use one of the following URLs to communicate with Certificate Enrollment Gateway.

 To support macOS (Apple) devices, the URL must start with `http` instead of `https`.

```
http://<CEG-server>/scep/<tenant-ID>/<digitalid-config>/mdm  
https://<CEG-server>/scep/<tenant-ID>/<digitalid-config>/mdm
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<digitalid-config>` is a digital ID configuration defined in the CEG Service.

For example:

```
http://cegserver.example.com/scep/tenant1/digitalid-config1/mdm  
https://cegserver.example.com/scep/tenant1/digitalid-config1/mdm
```


Configuring Certificate Enrollment Gateway for MDMWS and MDM-SCEP enrollment

To configure Certificate Enrollment Gateway for MDM Web Service (MDMWS) and MDM-SCEP enrollment, you must configure the MDMWS enrollment settings in Certificate Enrollment Gateway. You can edit the MDMWS enrollment settings using the Management Console interface.

To configure Certificate Enrollment Gateway for MDMWS and MDM-SCEP enrollment

1. Log in into the Management Console as explained in [Logging into the Management Console](#).
2. In the **Certificate Enrollment Gateway** pane, click **Manage Solution**.
A **Certificate Enrollment Gateway** page appears.
3. In the left navigation bar, click **Configuration**.
A **Product Configuration** pane appears.
4. Turn on **Enable Advanced Configuration**.
5. Click **Next**.
6. For all SCEP-related protocols (SCEP, MDM-SCEP, and Intune-SCEP), Certificate Enrollment Gateway uses RA certificates to sign and encrypt SCEP PKI messages. For an on-premises CA, you must specify one or more profiles that are defined in CA Gateway used to issue RA certificates.
 - a. Click the **CAGW** tab.
 - b. Configure the **RA Certificate Profile IDs** setting.
7. Click the **MDMWS** tab and configure the following settings.
 - [Enable MDMWS](#)
 - [MDM-SCEP Token Expire Lifetime](#)
 - [MDMWS Expired Token Clean-up Cron Job](#)
 - [MDMWS Users](#)
 - [MDMWS Enrollment Service Configuration](#)
8. Configure any other settings if required.
9. After configuring the settings, click **Validate** to validate the settings.
If any configuration errors are detected, correct the errors then click **Validate** again.
10. After validating the configuration settings, click **Next**.
Entrust PKI Hub uploads the configuration and any attached files, such as P12 credentials.
11. In the **Product Deployment Status** pane, re-deploy Certificate Enrollment Gateway with the updated configuration file by clicking **Deploy**.
A dialog box appears, prompting you to confirm the operation. Click **Yes** to confirm the operation and deploy the Certificate Enrollment Gateway solution.

Integrating WSTEP clients with Certificate Enrollment Gateway

The enrollment endpoints of the CEG Service can be:

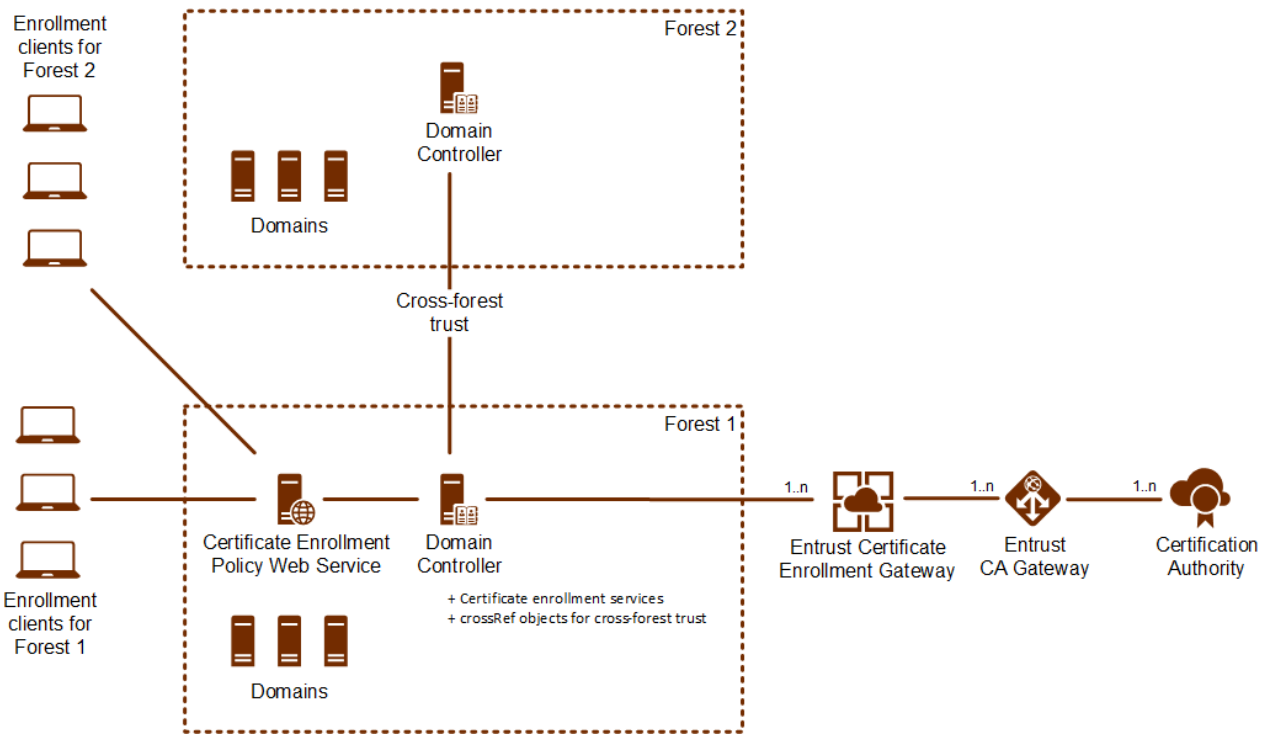
- An end-user or machine in a Windows domain.
- A user or machine connected to a Windows domain.

This section describes how to integrate these endpoints with Certificate Enrollment Gateway using WS-Trust X.509v3 Token Enrollment Extensions (WSTEP). Certificate Enrollment Gateway does not use NTLM authentication.

WSTEP integration architecture

In a WSTEP integration architecture, a Windows enrollment client can connect to a Domain Controller through the Certificate Enrollment Policy Web Service and request certificates from multiple Certification Authorities (CAs).

The following diagram shows a WSTEP integration architecture with Entrust Certificate Enrollment Gateway.



The following topics describe the components in the WSTEP integration architecture:

- [Enrollment clients](#)
- [Certificate Enrollment Policy Web Service](#)
- [Domain Controller](#)
- [Cross-forest trust](#)
- [Entrust Certificate Enrollment Gateway](#)
- [Entrust CA Gateway](#)
- [Certificate Authority](#)

Enrollment clients

For WSTEP enrollment, an enrollment client is a Windows user or a machine that requires a certificate. Enrollment clients are either in a Windows Domain or connected to a Windows Domain. With cross-forest trust (see [Cross-forest trust](#)), enrollment clients from one forest can request a certificate from a Domain Controller in another forest.

Certificate Enrollment Policy Web Service

The Certificate Enrollment Policy Web Service allows enrollment clients to retrieve certificate enrollment policies from a Certificate Authority (CA) when the clients are not permitted to access the Domain Controller. After receiving policy information from the Certificate Enrollment Policy Web Service, enrollment clients can then request a certificate from a certificate enrollment service.

The Windows server hosting the Certificate Enrollment Policy Web Service can be the Domain Controller or any other server in the domain. It is recommended that you install and configure the Certificate Enrollment Policy Web Service on a different server than the Domain Controller. The Certificate Enrollment Policy Web Service must be in same forest as the Domain Controller hosting the certificate templates and enrollment services.

Domain Controller

A Domain Controller is a server computer hosting Active Directory Domain Services that is responsible for allowing host access to domain resources. The Domain Controller authenticates users, stores user account information, and enforces security policy for a domain.

For WSTEP enrollment, a Domain Controller requires the following objects:

- A certificate enrollment service for each Certificate Authority (CA) that will issue certificates to enrollment clients. Each enrollment service will connect to a single CEG Service instance.
- For cross-forest deployments, a crossRef object for each cross-forest domain you must support.

Cross-forest trust

For WSTEP enrollment, the Domain Controller can use Kerberos authentication to authenticate Windows enrollment clients. Cross-forest trust is a Windows Server feature that allows multiple Active Directory forests to trust each other. With cross-forest trust, a Domain Controller for one forest can use Kerberos V5 LDAP referrals to locate and authenticate enrollment clients that exist in a different forest. For more information about referrals, see the Microsoft documentation.

Entrust Certificate Enrollment Gateway

Each Entrust Certificate Enrollment Gateway instance can connect to multiple Entrust CA Gateway instances, granting access to one or more Certificate Authorities (CAs). Each enrollment service object in the Domain Controller will connect to a single Certificate Enrollment Gateway instance and request a certificate from a single CA. To request certificates from multiple CAs, multiple enrollment services must be added to the Domain Controller.

Entrust CA Gateway

Entrust CA Gateway enables full certificate lifecycle management and operational management across all your Entrust-supported Certification Authorities (CAs). Each Entrust CA Gateway client can access one or several CAs. Certificate Enrollment Gateway will send certificate requests to Entrust CA Gateway. Entrust CA Gateway will forward the request to the intended Managed CA, and send the generated certificate back to Certificate Enrollment Gateway.

Certificate Authority

A Certificate Authority (CA) issues certificates to enrollment clients. Entrust CA Gateway can support multiple CAs, called Managed CAs. Each enrollment service object in the Domain Controller will connect to a single Certificate Enrollment Gateway instance and request a certificate from a single CA. To request certificates from multiple CAs, multiple enrollment services must be added to the Domain Controller.

Configuring the Windows domain for WSTEP enrollment

This section describes how to configure the Windows domain for WSTEP enrollment with Entrust Certificate Enrollment Gateway.

Certificate Enrollment Gateway supports read-only domain controllers for WSTEP enrollment. A read-only domain controller (RODC) is a server that hosts an Active Directory database's read-only partitions and responds to security authentication requests. Certificate Enrollment Gateway can accept WSTEP enrollment requests and authenticate the request using an RODC.

Any configuration changes to a domain controller that are documented in this guide must be performed on the write-able domain controller.

- [Active Directory schema requirements](#)
- [Active Directory role requirements for running the Entrust-provided PowerShell scripts](#)
- [Creating a service logon account for read-only access to Active Directory](#)

- [Creating a Kerberos Service Account for Kerberos authentication](#)
- [Configuring the Group Policy for cross-forest deployments](#)
- [Adding referrals for cross-forest deployments](#)

Active Directory schema requirements

For WSTEP enrollment with Certificate Enrollment Gateway, your Windows domain must have the 2016 Active Directory schema or later.

To check the current Active Directory schema version

1. Log in to the server hosting Active Directory.
2. Open a PowerShell window. Select **Start > Windows PowerShell > Windows PowerShell**.
3. Enter the following command:

```
Get-ADObject (Get-ADRootDSE).schemaNamingContext -properties objectVersion
```

The version returned by the command must be 87 or greater.

Active Directory role requirements for running the Entrust-provided PowerShell scripts

Entrust provides some Windows PowerShell scripts for configuring WSTEP enrollment with Certificate Enrollment Gateway. You can use these PowerShell scripts to install and configure the Certificate Enrollment Policy Web Service (CEP Service), and an enrollment service for each CA that will issue certificates to the WSTEP endpoints.

To run the PowerShell scripts provided by Entrust, the user account must have the Active Directory roles "Domain Admin" and "Enterprise Admin".

Creating a service logon account for read-only access to Active Directory


For WSTEP enrollment, Certificate Enrollment Gateway requires a domain user account for read-only access to LDAP and the Global Catalog in Active Directory. This domain user account must be a service account without any special permissions.

If you will use Kerberos authentication, it is recommended that you use the Kerberos Service Account for read-only access to Active Directory instead of creating a separate service account for read-only access to Active Directory. For information about creating the Kerberos Service Account, see [Creating a Kerberos Service Account for Kerberos authentication](#).

Creating a Kerberos Service Account for Kerberos authentication

To use Kerberos authentication, you must create a Kerberos Service Account (also called a Kerberos principal) in Active Directory domain controller. You must specify this Kerberos Service Account when generating a Kerberos keytab file.

A Kerberos Service Account is a standard Active Directory user which belongs to the top-level parent domain. No special permissions are required for this user account.

 Certificate Enrollment Gateway supports connections to an entire Domain Forest. Certificate Enrollment Gateway must point to the top-level domain of the forest to work across the entire forest.

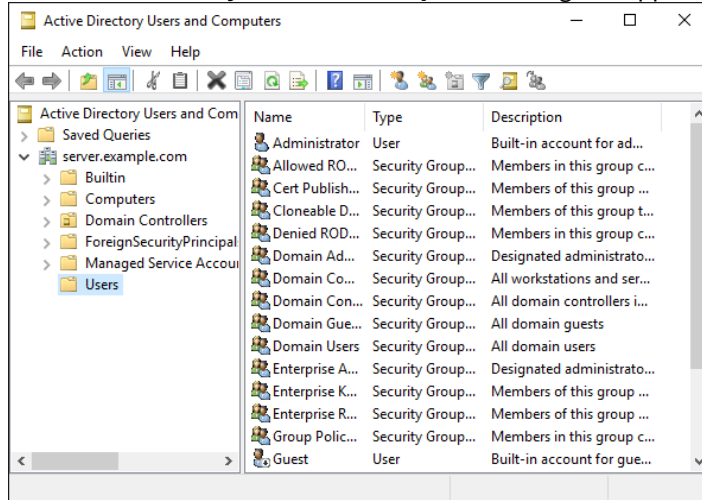
You must create the same Kerberos account for all forests in cross-forest deployments.

To create a Kerberos Service Account

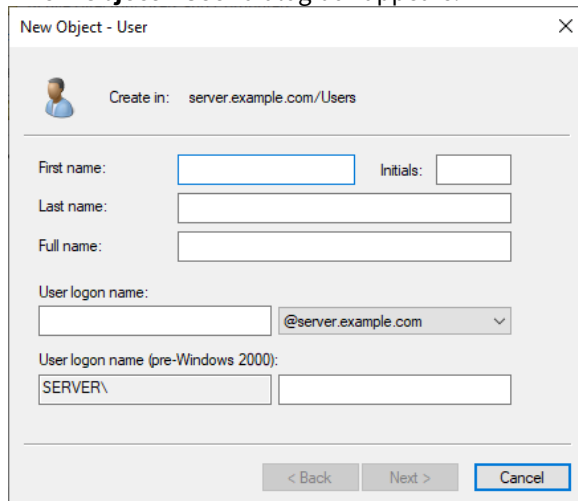
1. Log in to the server hosting the Active Directory domain controller as a domain administrator or a user who is a member of the built-in Account Operators domain group.

- Open the Active Directory Users and Computers administrative tool (select **Start > Windows Administrative Tools > Active Directory Users and Computers**).

The **Active Directory Users and Computers** dialog box appears.



- Right-click the folder where you want to create the new account and select **New > User**. A **New Object – User** dialog box appears.



Create in: server.example.com/Users

First name: Initials:

Last name:

Full name:

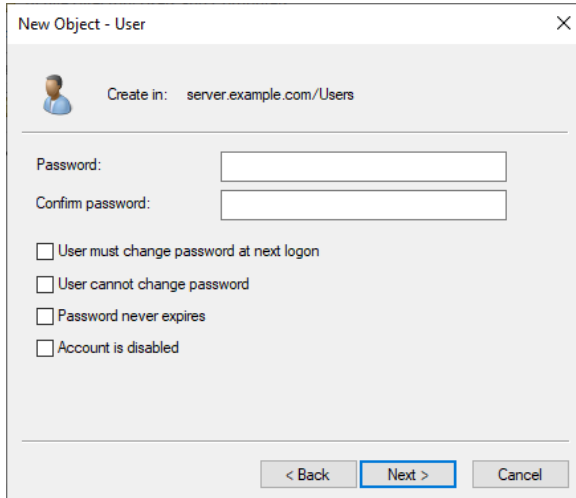
User logon name: @server.example.com

User logon name (pre-Windows 2000): SERVER\

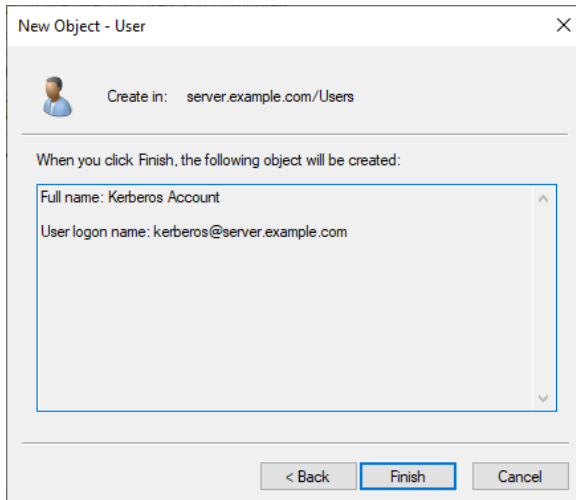
< Back Next > Cancel

- Using the **First name**, **Last name**, and **Full name** fields, enter a name for the new user account. At a minimum, you must enter a value into the **Full name** field. Entering values into the **First name** and **Last name** field will automatically fill the **Full name** field.
- In the **User logon name** field, enter a Windows logon name for the user account.
- (Optional.) In the **User logon name (pre-Windows 2000)** field, enter a logon name for the user account for pre-Windows 2000 computers.

7. Click **Next**.

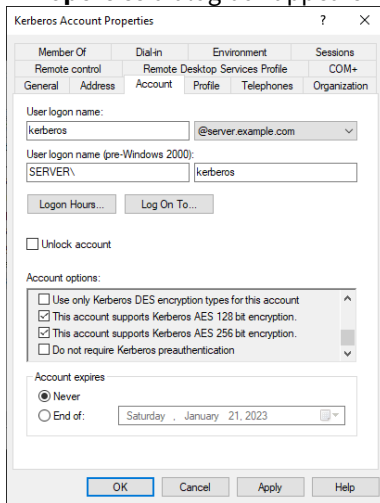


8. In the **Password** field, enter a password for the user account.
9. In the **Confirm password** field, enter the password again to confirm the password.
10. Deselect **User must change password at next logon**.
11. To avoid service interruptions because of an expired password, select **Password never expires**. If the password ever expires, you will need to reset the password, recreate the Kerberos keytab file, and then update the Certificate Enrollment Gateway configuration.
12. Click **Next**.



13. Record the user logon name of the account (such as `kerberos@example.com`). You will use this logon name later to create the Kerberos keytab file.
14. Click **Finish**.

- Double-click the account you just created. A **Properties** dialog box appears for the account.



- Click the **Account** tab.
- Under **Account options**:
 - Select **This account supports Kerberos AES 128 bit encryption**.
 - Select **This account supports Kerberos AES 256 bit encryption**.
- Click **OK**.

Configuring the Group Policy for cross-forest deployments

WSTEP enrollment can use Kerberos authentication to authenticate Windows endpoints. Kerberos authentication uses service principal names to associate a service instance with a service sign-in account. A service principal name is a unique identifier of a service instance. With Kerberos authentication, a service principal name allows a client application to request service authentication for an account, even if the client does not have the account name.

A service principal name (SPN) is a string that consists of either two or three parts, with each part separated by a forward slash. An example of a two-part SPN:

HTTP/server.example.com@EXAMPLE.COM

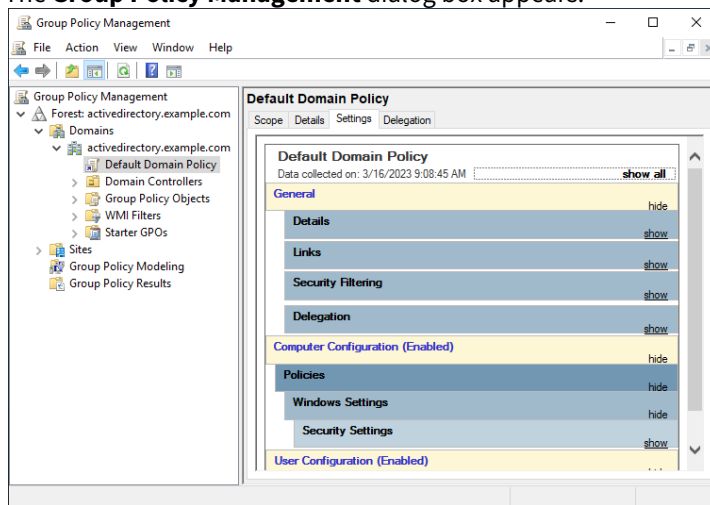
For cross-forest deployments, both the Key Distribution Center (KDC) and the Kerberos client must search a list of trusted forests when attempting to resolve a two-part SPN if the SPN cannot be found in the local forest. The list of trusted forests that the KDC and Kerberos clients can search is controlled by Group Policy settings in the domain controller. Cross-forest WSTEP enrollment can fail if the KDC or Kerberos client cannot resolve the two-part SPN. The list of trusted forests must be the same for both the KDC and Kerberos clients.

i To ensure consistent behavior, the Global Policy settings must be supported and set identically on all domain controllers in the domain.

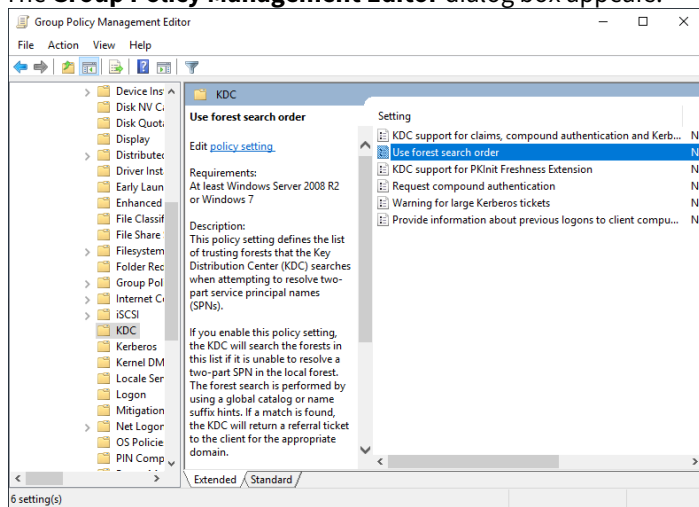
To configure the Group Policy for cross-forest deployments

- Log in to the server hosting Active Directory as a member of the Domain Admins and Enterprise Admins groups.

2. Select **Start > Windows Administrative Tools > Group Policy Management**. The **Group Policy Management** dialog box appears.

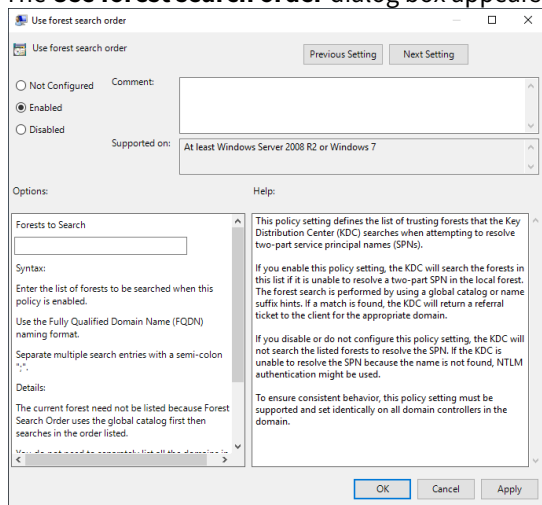


3. In the tree view, select **Group Policy Management > Forest: <forest> > Domains > <domain> > Default Domain Policy**. Where **<forest>** is the FQDN (fully qualified domain name) of the forest, and **<domain>** is the FQDN of the domain.
4. Select **Action > Edit** to edit the default domain policy for the domain. The **Group Policy Management Editor** dialog box appears.



5. Expand **Computer Configuration > Policies > Administrative Templates > System > KDC**.
6. In the **Settings** pane, select **Use Forest Search Order**.

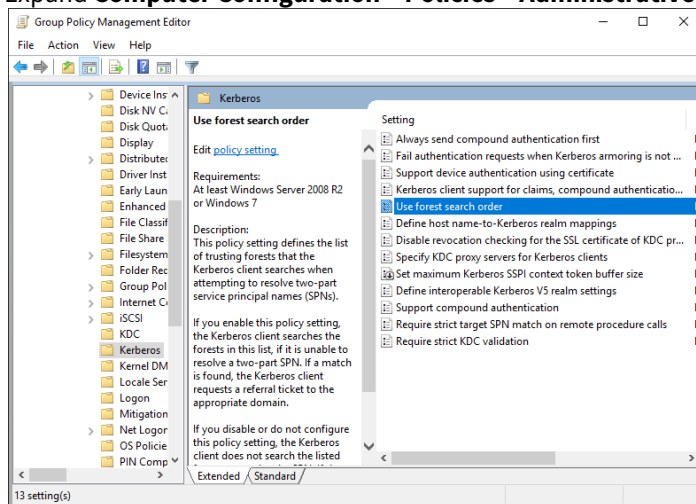
7. Select **Action > Edit** to edit the **Use Forest Search Order** setting. The **Use forest search order** dialog box appears.



8. Select **Enabled**.
9. In the **Options** pane, in the **Forests to Search** field, enter the list of trusted forests that the Key Distribution Center (KDC) will search when attempting to resolve a two-part SPN that does not exist in the local forest. Separate each forest with a semicolon. For example:

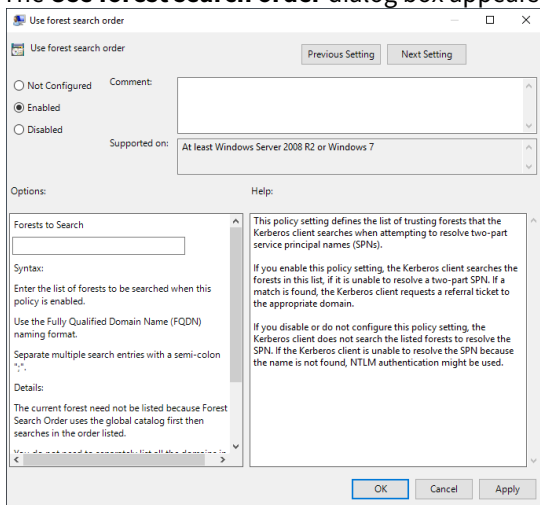
example.com;example.net;example.org

10. Click **OK**.
11. Expand **Computer Configuration > Policies > Administrative Templates > System > Kerberos**.



12. In the **Settings** pane, select **Use Forest Search Order**.

13. Select **Action > Edit** to edit the **Use Forest Search Order** setting. The **Use forest search order** dialog box appears.



14. Select **Enabled**.
15. In the **Options** pane, in the **Forests to Search** field, enter the list of trusted forests that Kerberos clients will search when attempting to resolve a two-part SPN that does not exist in the local forest. Separate each forest with a semicolon. For example:

example.com;example.net;example.org

16. Click **OK**.

Adding referrals for cross-forest deployments

WSTEP enrollment can use Kerberos authentication to authenticate Windows endpoints. For cross-forest deployments, Windows endpoints are located using Kerberos V5 LDAP referrals. The domain controller (Active Directory Domain Services), maintains referral data in its Configuration container, in crossRef objects. For more information about referrals, see the Microsoft documentation.

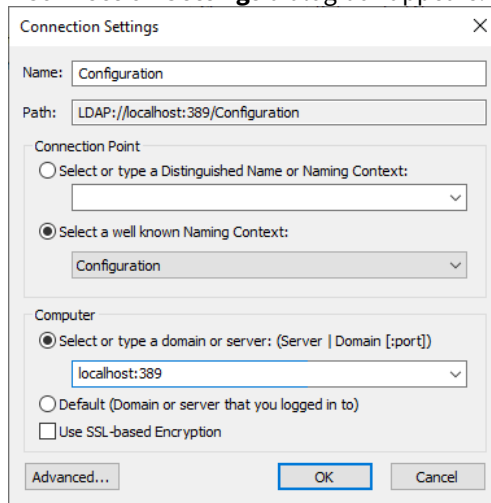
For cross-forest deployments with WSTEP enrollment, you must manually add a crossRef object into the domain controller for each cross-forest domain that you must support.

To add a cross-forest referral in a domain controller for cross-forest deployments

1. Open ADSI Edit. Select **Start > Windows Administrative Tools > ADSI Edit**.
2. Connect to the Configuration context.

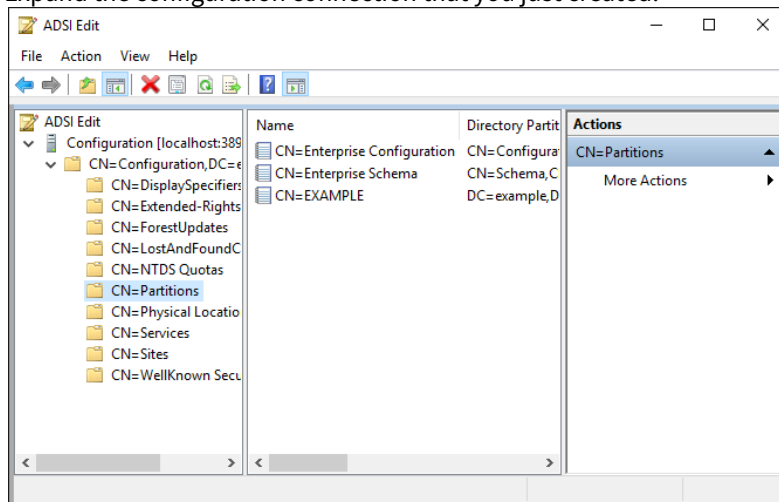
- a. Select **Action > Connect to**.

A **Connection Settings** dialog box appears.



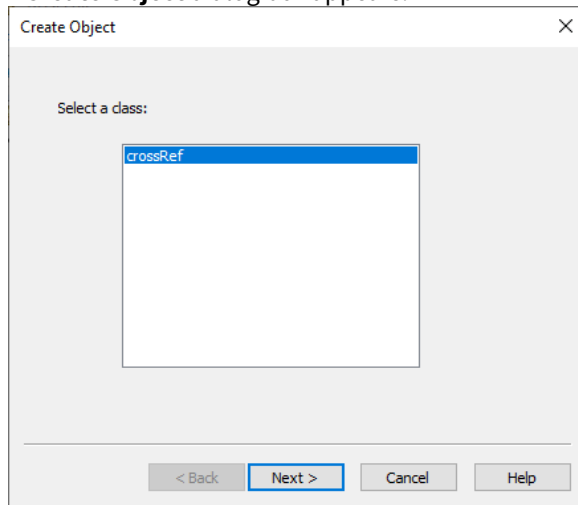
- b. In the **Name** field, enter a unique name for the connection.
- c. Under **Connection Point**, click **Select a well known Naming Context**, and then select **Configuration**.
- d. Under **Computer**, click **Select or type a domain or server**, and then enter the server and port of the domain controller, using the form `<server>:<port>`. If you are on the server hosting the domain controller, you can enter `localhost` for `<server>`.
- e. Click **OK**.

3. Expand the configuration connection that you just created.

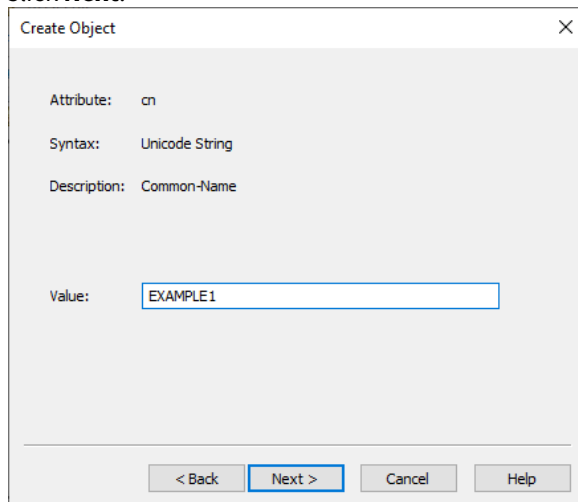


4. Expand **CN=Configuration,<suffix>** > **CN=Partitions**.
Where **<suffix>** is the suffix (distinguished name) of the domain controller.

5. Select **CN=Partitions**, and then select **Action > New > Object**. A **Create Object** dialog box appears.

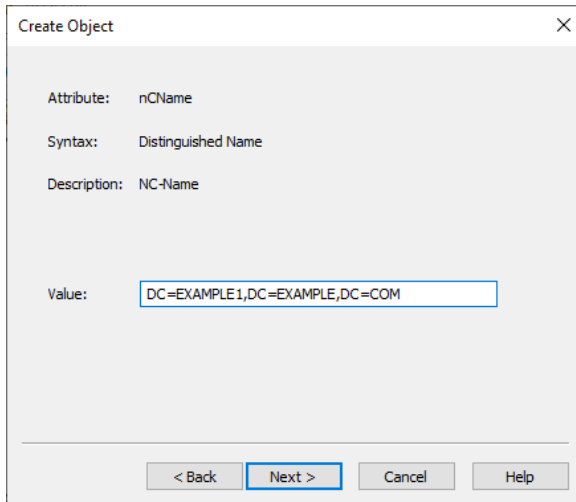


6. Select **crossRef**.
7. Click **Next**.



8. In the **Value** field, enter the NetBIOS name of a cross-forest domain.

9. Click **Next**.



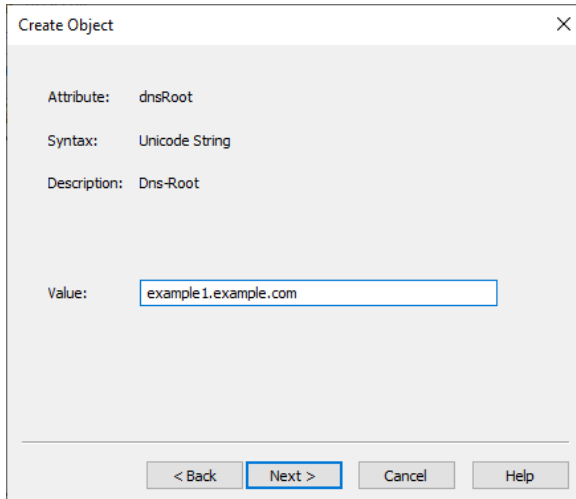
The screenshot shows a 'Create Object' dialog box with the following details:

- Attribute: nCName
- Syntax: Distinguished Name
- Description: NC-Name
- Value: DC=EXAMPLE1,DC=EXAMPLE,DC=COM

At the bottom, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'. The 'Next >' button is highlighted with a blue border.

10. In the **Value** field, enter the distinguished name of the cross-forest domain.

11. Click **Next**.



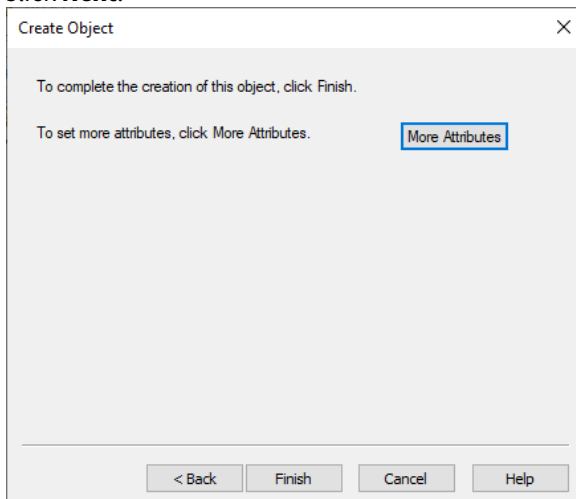
The screenshot shows a 'Create Object' dialog box with the following details:

- Attribute: dnsRoot
- Syntax: Unicode String
- Description: Dns-Root
- Value: example1.example.com

At the bottom, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'. The 'Next >' button is highlighted with a blue border.

12. In the **Value** field, enter the DNS name of the cross-forest domain.

13. Click **Next**.



The screenshot shows the final step of the 'Create Object' dialog box. It contains the following text:

- To complete the creation of this object, click Finish.
- To set more attributes, click More Attributes.

A 'More Attributes' button is highlighted with a blue border. At the bottom, there are four buttons: '< Back', 'Finish', 'Cancel', and 'Help'.

14. Click **Finish**.
A crossRef object is added for the cross-forest domain.
15. Repeat this procedure for each cross-forest domain the domain controller must support for cross-forest referrals.

Creating Kerberos files for Certificate Enrollment Gateway

If you will use Kerberos authentication for WSTEP enrollment, you must create one or more Kerberos files for Certificate Enrollment Gateway. When configuring Certificate Enrollment Gateway, you must provide these files when editing Certificate Enrollment Gateway settings.

- [Creating a Kerberos keytab file for WSTEP enrollment](#)
- [Creating a Kerberos configuration file for cross-forest WSTEP enrollment](#)

Creating a Kerberos keytab file for WSTEP enrollment

To support Kerberos authentication with WSTEP enrollment, Certificate Enrollment Gateway requires a Kerberos keytab file. A keytab file contains pairs of Kerberos principals and encrypted keys derived from the Kerberos password. Certificate Enrollment Gateway will use the keytab file to authenticate to various remote systems using Kerberos without entering a password.

Log in to the Active Directory domain controller as a user with Domain Admin and Enterprise Admin permissions, and run the following command in PowerShell.

```
ktpass /out <keytab_path> /mapuser <user> /princ <principal> /pass <password> /ptype  
KRB5_NT_PRINCIPAL /crypto <algorithm>
```

See below for a description of each parameter.

- <keytab_path>
- <user>
- <principal>
- <password>
- <algorithm>

For example:

```
ktpass /out "C:\Users\cegconfig\Documents\kerberos.keytab" /mapuser  
kerberos@example.com /princ HTTP/cegserver1.example.com@EXAMPLE.COM /pass  
EXAMPLE_password1234 /ptype KRB5_NT_PRINCIPAL /crypto All
```

<keytab_path>

The full path and file name of the keytab file that the command will generate. For example:

```
C:\Users\cegconfig\Documents\kerberos.keytab
```

You will transfer this keytab file to the server hosting Certificate Enrollment Gateway.

<user>

The Kerberos Service Account, using the following format:

```
<logon>@<domain>
```

Where:

- `<logon>` is the Windows logon name of the Kerberos Service Account, in lowercase letters. For example, `kerberos`.
- `<domain>` is the domain name of the Kerberos Service Account, in lowercase characters. For example, `example.com`.

For example:

```
kerberos@example.com
```

`<principal>`

The principal name of the Kerberos Service Account. The principal must be in the form:

```
HTTP/<ceg-fqdn>@<DOMAIN>
```

Where:

- `<ceg-fqdn>` is the fully qualified domain name of the server hosting Certificate Enrollment Gateway. For example, `cegserver1.example.com`.
- `<DOMAIN>` is the domain name of the Kerberos Service Account, in uppercase characters. For example, `EXAMPLE.COM`.

For example:

```
HTTP/cegserver1.example.com@EXAMPLE.COM
```

This parameter is case-sensitive.


`<password>`

The password of the Kerberos Service Account.

`<algorithm>`

The keys that are generated in the keytab file. Permitted values:

- DES-CBC-CRC
- DES-CBC-MD5
- RC4-HMAC-NT
- AES256-SHA1
- AES128-SHA1
- All

 DES and RC4 algorithms are deprecated. You should specify an AES algorithm or All to allow all algorithms.

Creating a Kerberos configuration file for cross-forest WSTEP enrollment

To support Kerberos authentication with WSTEP enrollment in a cross-forest deployment, Certificate Enrollment Gateway requires a Kerberos configuration file. Certificate Enrollment Gateway uses the Kerberos configuration file for authenticating Kerberos V5 LDAP Referrals across forests in Active Directory.

The Kerberos configuration file (typically `krb5.conf`) file must contain the following information:

- A `[libdefaults]` section with a default realm.
If Kerberos authentication uses 3DES or RC4 algorithms, the section must also contain the setting `allow_weak_crypto = true`.
- A `[realms]` section with the top-level domain of each forest defined as a realm.

The following example is a `krb5.conf` file with two cross-forests.

```
[libdefaults]
    default_realm = EXAMPLE.COM
    renew_lifetime = 3600
    ticket_lifetime = 3600
    allow_weak_crypto = true

[realms]
    EXAMPLE.COM = {
        kdc = hostname1.example.com
        kdc = hostname2.example.com
        kdc = example.com
        admin_server = hostname1.example.com
        master_kdc = hostname1.example.com
        default_domain = EXAMPLE.COM
    }
    EXAMPLE.ORG = {
        kdc = hostname1.example.org
        admin_server = hostname1.example.org
        master_kdc = hostname1.example.org
        default_domain = EXAMPLE.ORG
    }
```

For more information about creating a `krb5.conf` file, see https://web.mit.edu/kerberos/krb5-1.12/doc/admin/conf_files/krb5_conf.html.

Adding the Windows Certificate Templates to Active Directory

The Entrust WSTEP Service is the Certificate Enrollment Gateway's implementation of the WSTEP protocol. The Entrust WSTEP Service will use Windows certificate templates when enrolling users, computers, or domain controllers with your Windows-native endpoints.

If you already have a Microsoft CA installed in Active Directory, the Certificate Templates feature is already enabled, and you can skip this section. Otherwise, you must add the Certificate Templates feature using either Windows PowerShell or the Windows graphical interface.

- [Adding the certificate templates feature using PowerShell](#)
- [Adding the certificate templates feature using the Windows graphical interface](#)

Adding the certificate templates feature using PowerShell

To add Certificate Templates using Windows PowerShell, complete the following procedure.

1. Log in to the server hosting Active Directory as a member of the Domain Admins and Enterprise Admins groups.
2. Open an elevated PowerShell window. Select **Start > Windows PowerShell, then right-click Windows PowerShell > Run as administrator.**
3. Run the following command.

```
Add-WindowsFeature RSAT-ADCS-Mgmt
```

4. Launch the Certificate Templates snap-in.

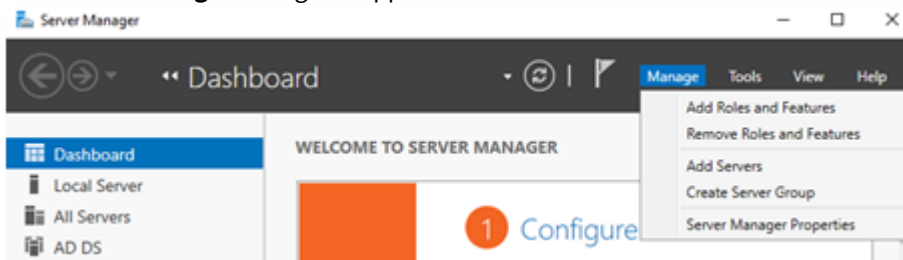
```
C:\Windows\System32\certtmpl.msc
```

5. Answer **Yes** when prompted to install the templates into Active Directory.

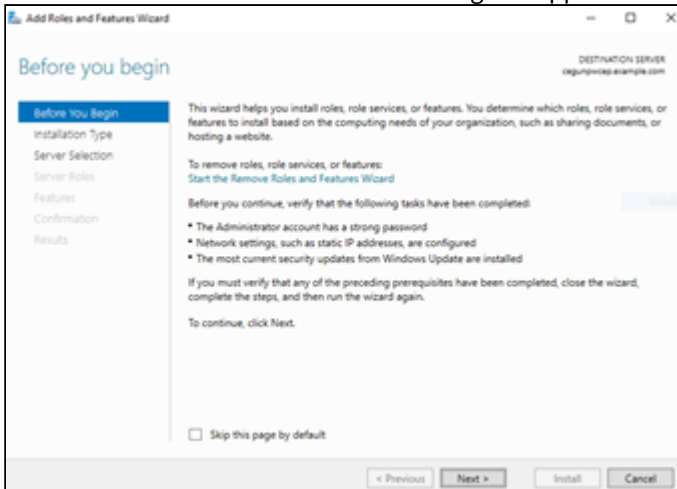
Adding the certificate templates feature using the Windows graphical interface

To add Certificate Templates using the Windows graphical interface, complete the following procedure.

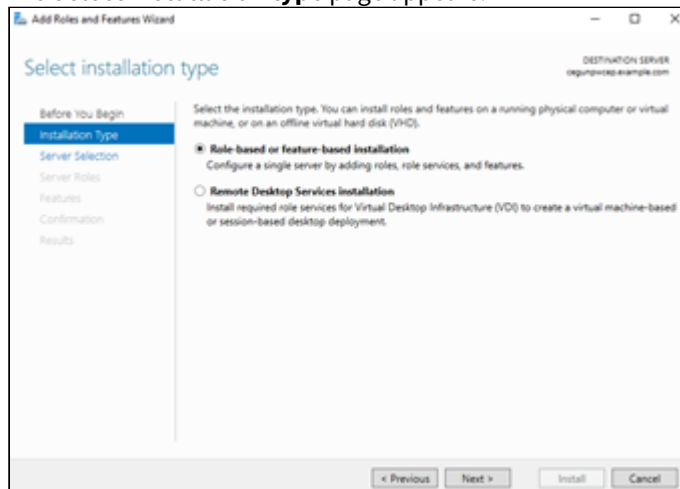
1. Log in to the server hosting Active Directory as a member of the Domain Admins and Enterprise Admins groups.
2. Open Server Manager. Select **Start > Server Manager.**
The **Server Manager** dialog box appears.



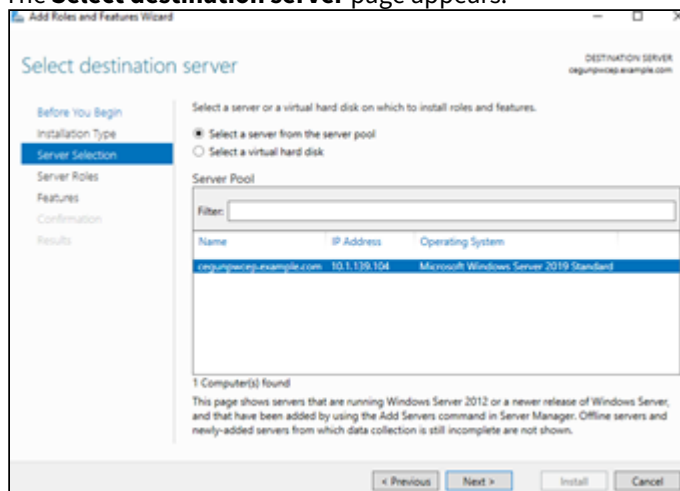
3. Select **Manage > Add Roles and Features.**
The **Add Roles and Features Wizard** dialog box appears.



4. If the **Before you Begin** page appears, click **Next**.
The **Select installation type** page appears.

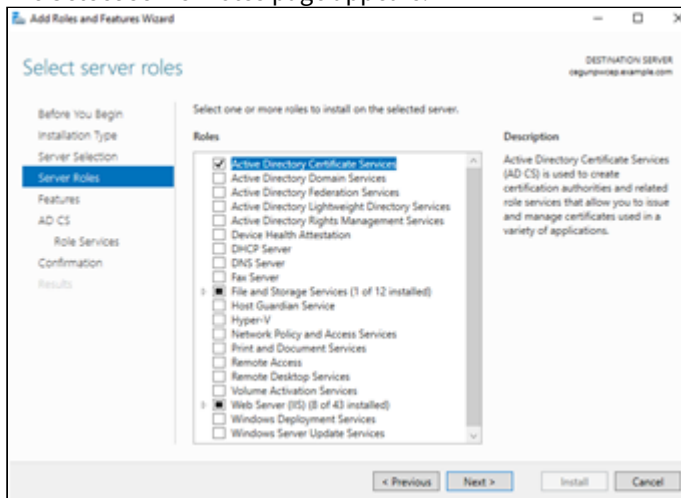


5. Select **Role-based or feature-based installation**.
6. Click **Next**.
The **Select destination server** page appears.

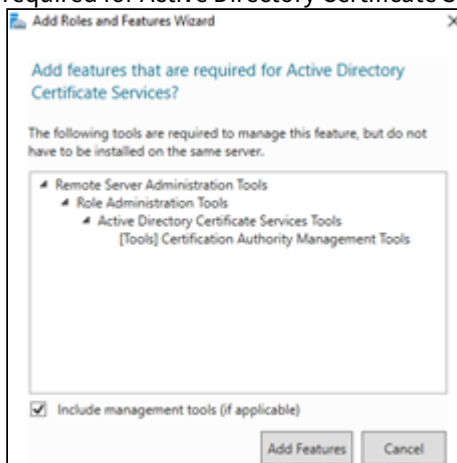


7. Click **Select a server from the pool**.
8. In the **Server Pool** list, select the server.

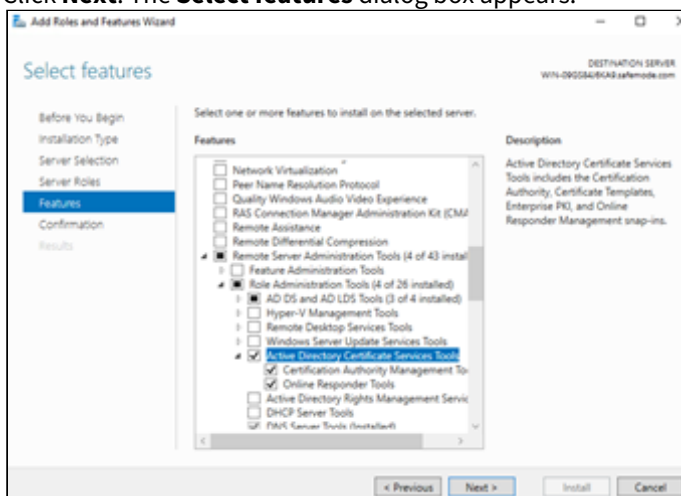
- Click **Next**.
The **Select server roles** page appears.



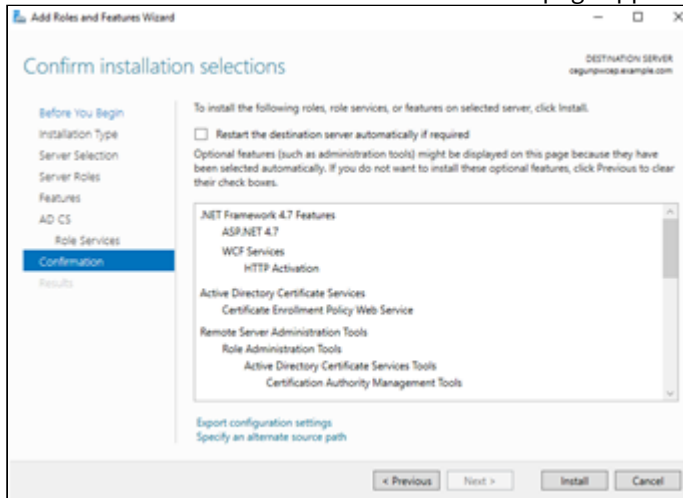
- Select **Active Directory Certificate Services**.
Another **Add Roles and Features Wizard** dialog box may appear, informing you that some features are required for Active Directory Certificate Services.



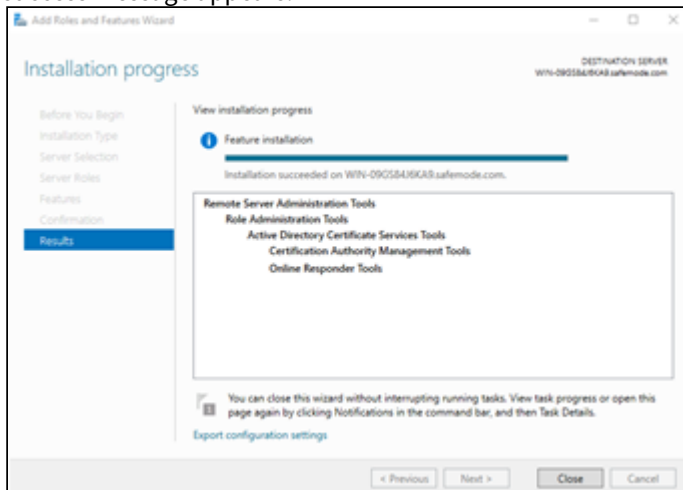
- Click **Add Features** to add these required features and close the dialog box.
- Click **Next**. The **Select features** dialog box appears.



13. Expand **Remote Server Administration Tools > Role Administration Tools**, then select **Active Directory Certificate Services Tools**.
14. Click **Next**. The **Confirm installation selections** page appears.

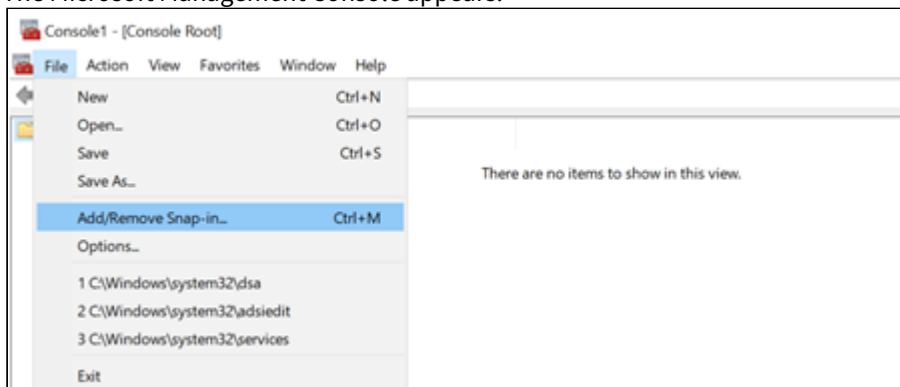


15. Click **Install**. The **Installation Progress** page appears. A progress indicator displays the progress of the installation. After the roles and features are installed, a success message appears.

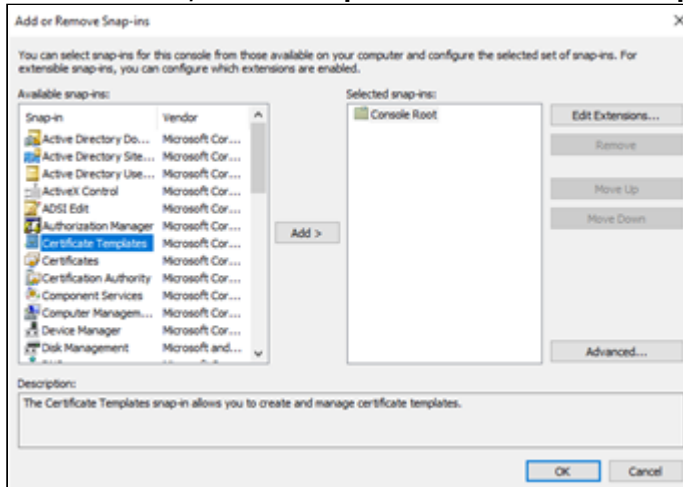


16. Click **Close**.
17. Run `mmc.exe`.

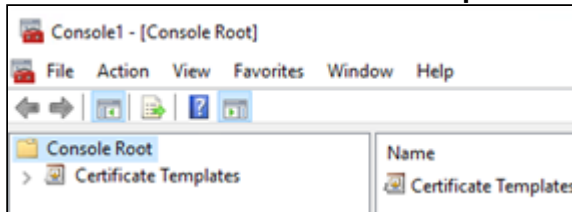
The Microsoft Management Console appears.



18. Select **File > Add/Remove Snap-in**. The **Add or Remove Snap-ins** dialog box appears.



19. In the **Available snap-ins** list, select **Certificate Templates**.
20. Click **Add**.
21. Click **OK** to close the **Add or Remove Snap-ins** dialog and return to the Microsoft Management Console.



22. Select **Certificate Templates**.
A dialog will appear, prompting to install the certificate templates.
23. Click **Yes** to install the templates.

Creating Windows certificate templates for the Entrust WSTEP Service

The Entrust WSTEP Service is a component of Certificate Enrollment Gateway. The Entrust WSTEP Service is Certificate Enrollment Gateway’s implementation of the WSTEP protocol.

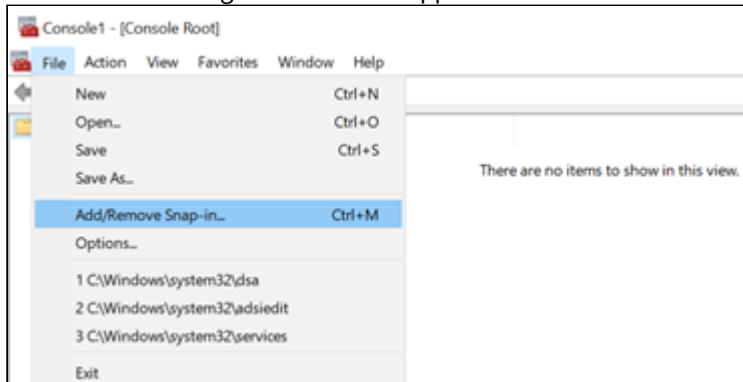
The Entrust WSTEP Service will use Windows certificate templates when enrolling users, computers, or domain controllers with your Windows-native endpoints. Create as many new certificate templates as you require. For example, users may require certificates with two key pairs (such as Encryption and Verification) or one key pair (such as Signature or Encryption).

To create a Windows certificate template for WSTEP

1. Log into Active Directory as a member of the Domain Admins group.

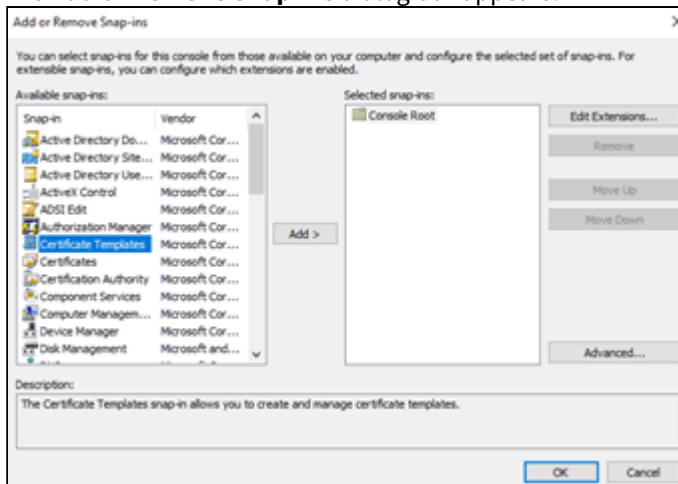
2. Run `mmc.exe` (Select **Start > Windows System > Run**, then enter `mmc.exe`).

The Microsoft Management Console appears.



3. Select **File > Add/Remove Snap-in**.

The **Add or Remove Snap-ins** dialog box appears.



4. In the **Available snap-ins** list, select **Certificate Templates**.
5. Click **Add**.
6. In the tree view, select the **Certificate Templates** snap-in.
7. Select the certificate you want to duplicate for the enrollment service. Supported templates:
 - Computer
 - Domain Controller
 - Kerberos Authentication
 - User
 - User Signature Only

8. Duplicate the template by selecting **Action > Duplicate Template**. A **Properties of New Template** dialog box appears.



9. Under each tab, configure template options as described in the following sections.

✘ Start configuring the template options from the **Compatibility** tab. Otherwise, the **Provider Category** option in the **Cryptography** tab will be locked to **Legacy Cryptographic Service Provider**.

- [Compatibility tab](#)
- [General tab](#)
- [Security tab](#)
- [Request Handling tab](#)
- [Cryptography tab](#)
- [Key Attestation tab](#)
- [Subject Name tab](#)
- [Issuance Requirements tab](#)
- [Extensions tab](#)

Compatibility tab

Under the **Compatibility** tab, select the compatibility settings based on the earliest versions of the operating systems running in your environment.

Option	Minimum version
Certification Authority	Server 2012 R2 <div style="border: 1px solid red; padding: 5px; margin-top: 10px;"> ✘ If you install a Windows Server 2016 CA, read the following Microsoft troubleshooting guide: Cannot select Windows Server 2016 CA-compatible certificate templates from Windows Server 2016 or later-based CAs or CEP servers. </div>
Certificate recipient	Windows 8.1 / Server 2012 R2. Earlier versions of Windows may not recognize template options introduced in later versions of the Windows certificate templates.

General tab

Under the **General** tab, configure the supported options.

Option	Supported	Value
Template display name	Yes	The display name for the new certificate template.
Template name	Yes	The name of the new template. The default value of this field is the value set in the Template display name field but without spaces. We recommend using this value. The name must contain only alphanumeric characters (a-z, A-Z, 0-9).
Validity Period	Yes	Any value allowed by the validity policy of the issuing CA. For information about using short certificate lifetimes with Entrust Authority Security Manager, see the Security Manager documentation.
Renewal Period	Yes	Controlled by the client.
Publish certificate in Active Directory	No	

Security tab

Under the **Security** tab, set the following options.

Option	Value
Groups or usernames	Select the group that will use the new certificate template. For example, select the Domain Users group to use a copy of the User certificate template. If the group is not listed, click Add to add the group to the list.
Permissions for Authenticated Users	Set the following Allow permissions for the selected group: Read, Enroll, Autoenroll.

Request Handling tab

Under the **Request Handling** tab, set the supported options.

Option	Supported	Value
Purpose	Yes	Encryption, Signature, Signature and encryption, and Signature and smartcard logon
Delete revoked or expired certificates (do not archive)	Yes	
Include symmetric algorithms allowed by the subject	Yes	SMIME settings
Archive subject's encryption private key	No	
Authorize additional service accounts to access the private key	No	
Allow private key to be exported	Yes	
Renew with the same key	Yes	
For automatic renewal of smart card certificates, use the existing key if a new key cannot be created	Yes	
Enroll subject without requiring any user input	Yes	
Prompt the user during enrollment	No	

Option	Supported	Value
Prompt the user during enrollment and require user input when the private key is used	No	

Cryptography tab

Under the **Cryptography** tab, set the supported options.

Option	Supported	Value
Provider Category	Yes	
Algorithm name	Yes	
Minimum key size	Yes	2048
Requests can use any provider available on the subject's computer	Yes	
Requests must use one of the following providers	Yes	
Request hash	Yes	
Use alternate signature format	No	

Key Attestation tab

Options under the **Key Attestation** tab are not supported.

Subject Name tab

Under the **Subject Name** tab, set the supported options.

Option	Supported	Value
Supply in request	Yes	The Subject Alternative Name RegisteredID is not supported. The Subject name types Title and Initials are not supported.

Option	Supported	Value
Use subject information from existing certificates for autoenrollment renewal requests	Yes	
Build from Active Directory information	Yes	
Subject name format	Yes	<p>For User certificate templates, the following formats are supported:</p> <ul style="list-style-type: none"> • Common name. • Fully distinguished name. This value is not supported when Certificate Enrollment Gateway has mapped the Windows certificate template to a Profile ID in CA Gateway. <p>For Computer or Domain Controller certificate templates, the following formats are supported:</p> <ul style="list-style-type: none"> • Common name. • DNS. • Fully distinguished name. This value is not supported when Certificate Enrollment Gateway has mapped the Windows certificate template to a Profile ID in CA Gateway.
Include e-mail name in subject name	No	
E-mail name	Yes	
DNS name	Yes	
User principal name (UPN)	Yes	
Service principal name (SPN)	Yes	

Issuance Requirements tab

Options under the **Issuance Requirements** tab are not supported.

Option	Supported
CA certificate manager approval	No. All certificate requests are processed automatically without any approval.
Require the following for reenrollment	No

Extensions tab

All options under the **Extensions** tab are supported.

See the following table for the **Key Usage** combinations supported by each **Purpose** in the **Request Handling** tab.

Purpose	Supported Key Usage Combinations
Encryption	Key Encipherment
Signature	Digital Signature Digital Signature+Non-repudiation
Signature and Encryption	Digital Signature+Key Encipherment
Signature and smartcard logon.	Digital Signature

Configuring Active Directory for secure LDAP (Optional)

For WSTEP enrollment, Certificate Enrollment Gateway supports secure LDAP (LDAPS) connections with Active Directory. LDAPS connections with Active Directory is optional. The following topics describe how to configure Active Directory for secure LDAP.

- [Creating a CSR for an Active Directory server certificate](#)
- [Installing the CA certificate chain for the Active Directory certificate](#)
- [Issuing the Active Directory server certificate with Entrust PKI as a Service](#)
- [Issuing the Active Directory server certificate with an on-premises CA](#)
- [Installing the Active Directory server certificate](#)
- [Verifying LDAPS in Active Directory](#)

Creating a CSR for an Active Directory server certificate

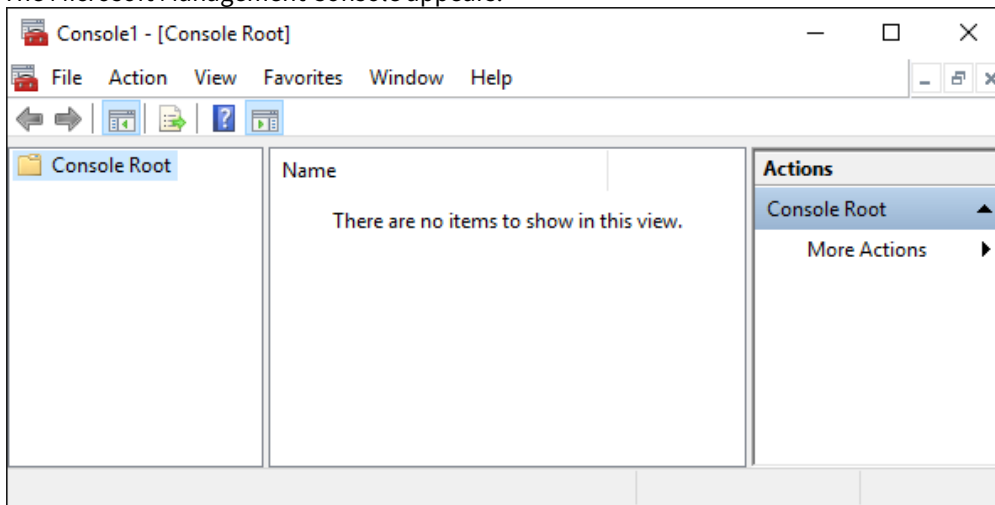
Active Directory requires a server certificate to secure communications to the directory over LDAPS. The following procedure describes how to create a certificate signing request (CSR) an Active Directory server certificate. A CSR contains information that the issuing CA will use to create the certificate. Entrust PKI as a Service or an on-premises CA can process the CSR and issue the certificate.

To create a CSR for an Active Directory server certificate

1. Log into Active Directory as a member of the Domain Admins group.

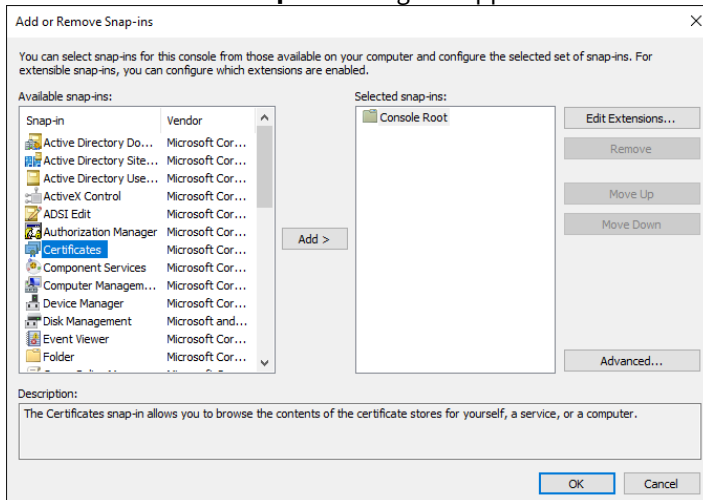
2. Run `mmc.exe` (Select **Start > Windows System > Run**, then enter `mmc.exe`).

The Microsoft Management Console appears.



3. Select **File > Add/Remove Snap-in**.

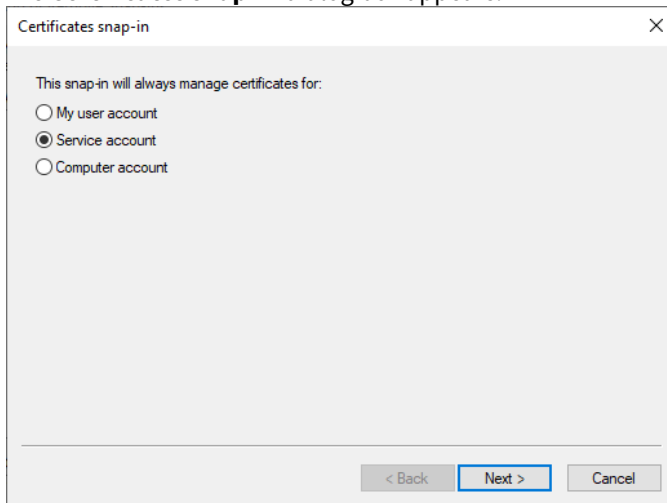
The **Add or Remove Snap-ins** dialog box appears.



4. In the **Available snap-ins** list, select **Certificates**.

5. Click **Add**.

The **Certificates snap-in** dialog box appears.



Certificates snap-in

This snap-in will always manage certificates for:

My user account

Service account

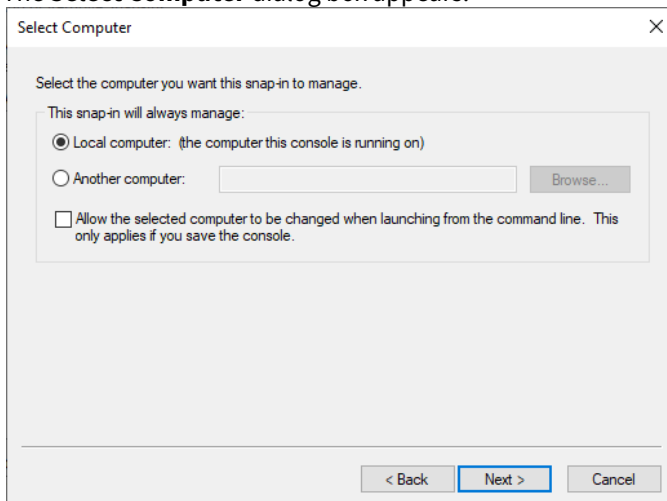
Computer account

< Back Next > Cancel

6. Select **Service account**.

7. Click **Next**.

The **Select Computer** dialog box appears.



Select Computer

Select the computer you want this snap-in to manage.

This snap-in will always manage:

Local computer: (the computer this console is running on)

Another computer: Browse...

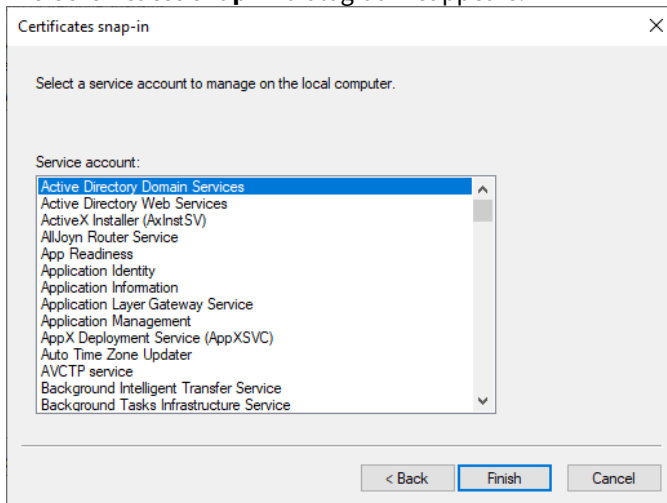
Allow the selected computer to be changed when launching from the command line. This only applies if you save the console.

< Back Next > Cancel

8. Select **Local computer**.

9. Click **Next**.

The **Certificates snap-in** dialog box reappears.



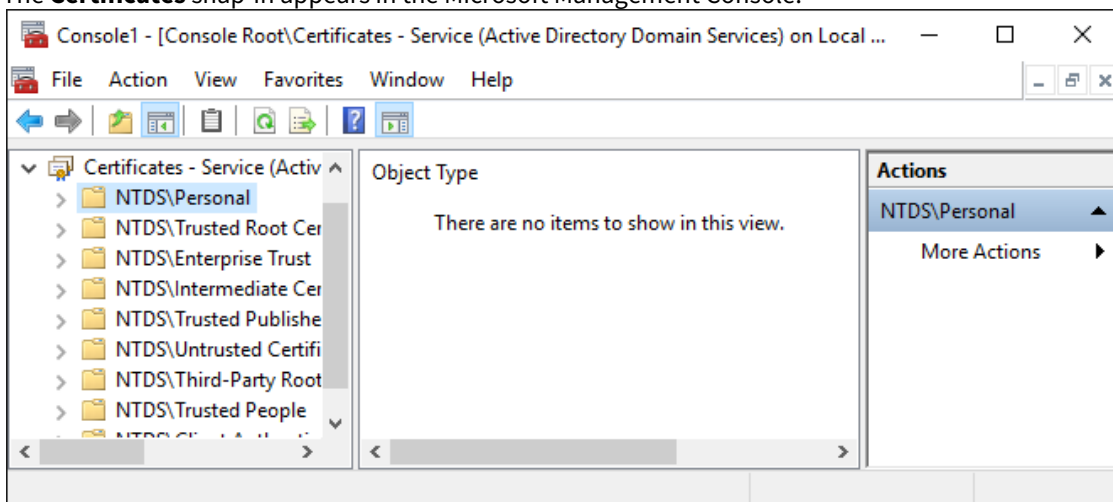
10. Select **Active Directory Domain Services**.

11. Click **Finish**.

The **Certificates** snap-in as added to the list of Selected snap-ins.

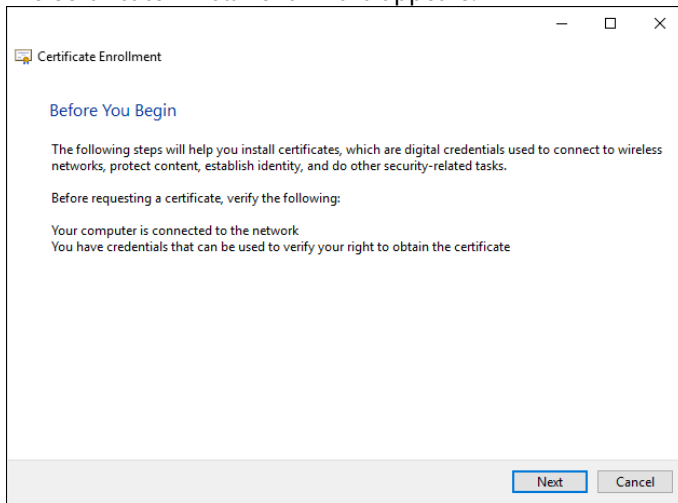
12. Click **OK**.

The **Certificates** snap-in appears in the Microsoft Management Console.

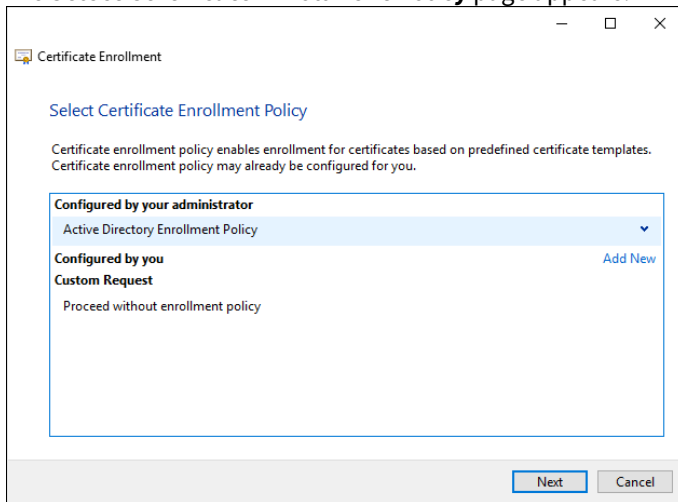


13. In the tree view, select **Certificates > NTDS\Personal**.

14. Select **Action > All Tasks > Advanced Operations > Create Custom Request**.
The Certificate Enrollment wizard appears.



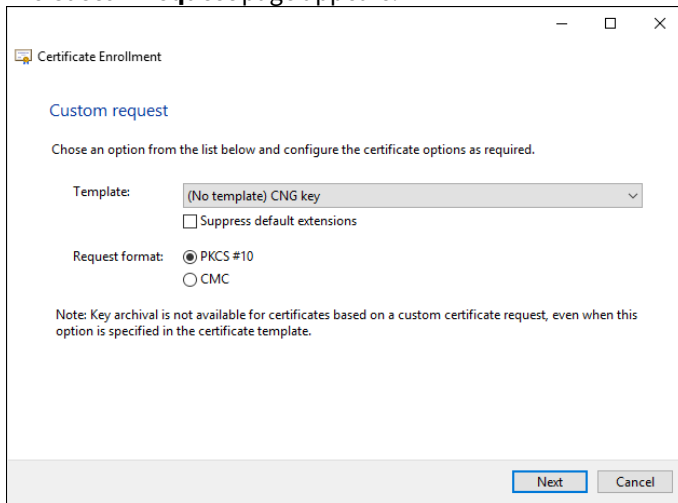
15. Click **Next**.
The **Select Certificate Enrollment Policy** page appears.



16. Under **Configured by your administrator**, select **Active Directory Enrollment Policy**.

17. Click **Next**.

The **Custom request** page appears.



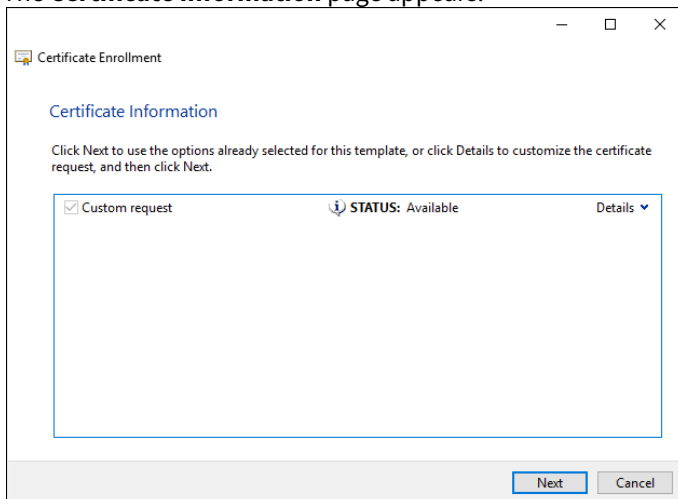
The screenshot shows a window titled "Certificate Enrollment" with a "Custom request" section. It includes a "Template" dropdown menu set to "(No template) CNG key", a "Suppress default extensions" checkbox, and "Request format" radio buttons for "PKCS #10" (selected) and "CMC". A note at the bottom states: "Note: Key archival is not available for certificates based on a custom certificate request, even when this option is specified in the certificate template." "Next" and "Cancel" buttons are at the bottom right.

18. In the **Template** drop-down list, select **(No template) CNG key**.

19. For **Request format**, select **PKCS #10**.

20. Click **Next**.

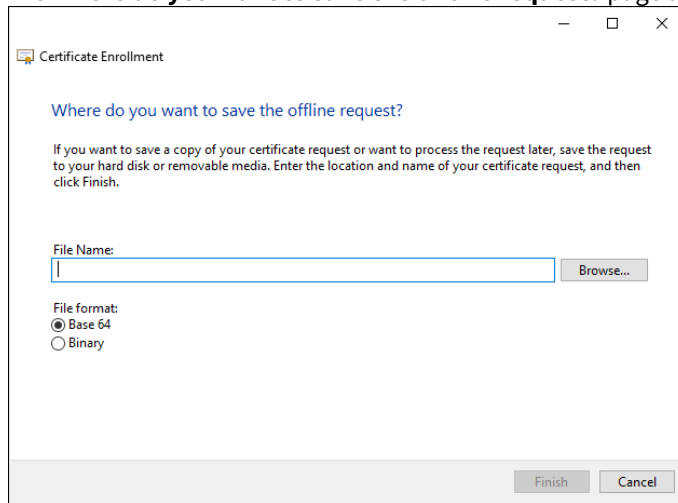
The **Certificate Information** page appears.



The screenshot shows a window titled "Certificate Enrollment" with a "Certificate Information" section. It contains a table with one row: "Custom request" (checked), "STATUS: Available", and "Details" (dropdown). A note above the table says: "Click Next to use the options already selected for this template, or click Details to customize the certificate request, and then click Next." "Next" and "Cancel" buttons are at the bottom right.

21. Click **Next**.

The **Where do you want to save the offline request?** page appears.



22. In the **File Name** field, enter the path and file name for the CSR, or click **Browse** to select a location.

23. For **File format**, select **Base 64**.

24. Click **OK**.

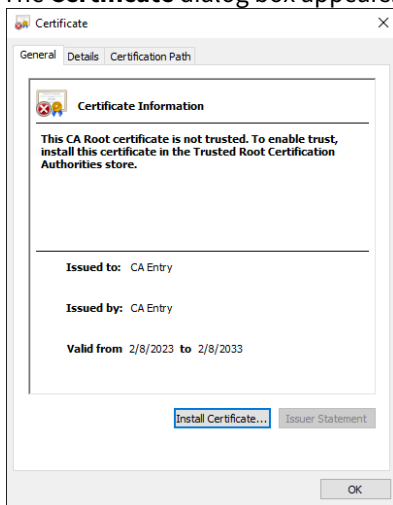
Installing the CA certificate chain for the Active Directory certificate

For Active Directory to trust the server certificate, you must install the CA certificate chain for the certificate into the server hosting Active Directory. You must install the entire CA certificate chain, from the root CA to the issuing CA (the CA that issued the server certificate). For an on-premises CA, the root CA may be the issuing CA. You must install the CA certificate chain into Active Directory before you install the server certificate.

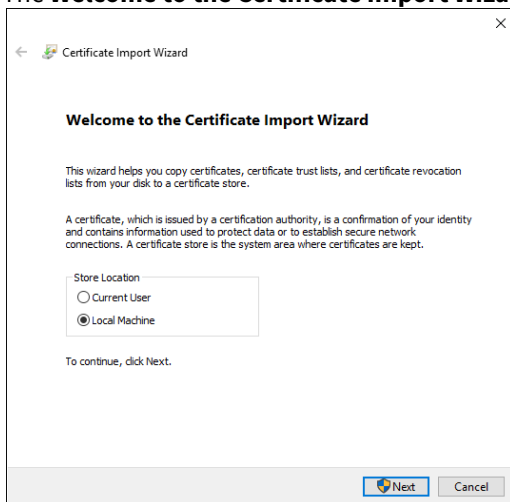
To install a CA certificate into Active Directory

1. For an on-premises CA, obtain all CA certificates in the CA certificate chain using your on-premises CA tools. See the documentation for your on-premises CA for instructions.
2. For Entrust PKI as a Service, download all CA certificates in the certificate chain:
 - a. Log in the Entrust Certificate Services interface.
 - b. Select **Administration > PKIaaS Management**.
A list of private CAs appear.
 - c. For each CA in the TLS certificate chain (from the Issuing CA to the Root CA), select the CA and then click **Download certificate**.

3. Double-click the CA certificate file.
The **Certificate** dialog box appears.

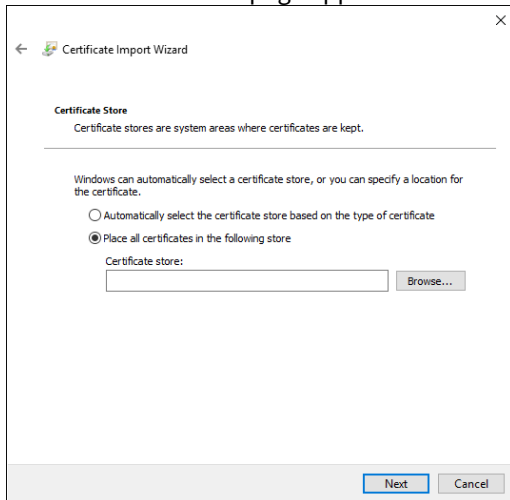


4. Click **Install Certificate**.
The Certificate Import Wizard appears.
5. The **Welcome to the Certificate Import Wizard** page appears.



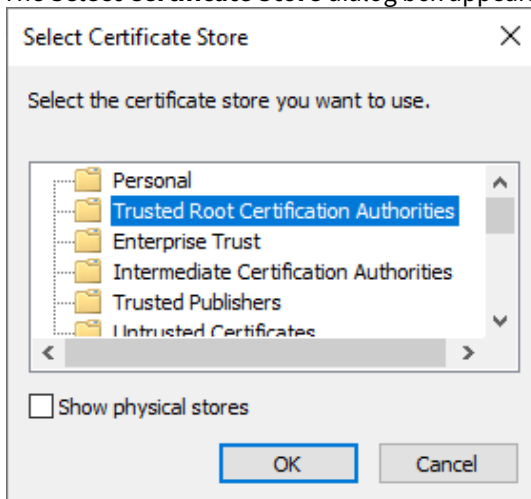
- a. For **Store Location**, select **Local Machine**.
- b. Click **Next**.

6. The **Certificate Store** page appears.



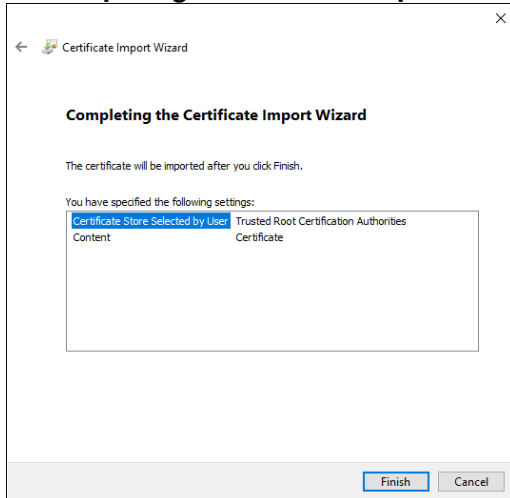
- a. Select **Place all certificates in the following store.**
- b. Click **Browse.**

The **Select Certificate Store** dialog box appears.



- c. If the CA certificate is a root CA certificate, select **Trusted Root Certification Authorities.**
- d. If the CA certificate is a subordinate (intermediate) CA certificate, select **Intermediate Certification Authorities.**
- e. Click **OK.**
- f. Click **Next.**

- The **Completing the Certificate Import Wizard** page appears.



- Click **OK**.

Issuing the Active Directory server certificate with Entrust PKI as a Service

After creating the certificate signing request (CSR) for the Certificate Enrollment Gateway certificate, you can submit the CSR to an Issuing CA in Entrust PKI as a Service. The Issuing CA will process the CSR and generate the certificate.

To submit the CSR to Entrust PKI as a Service and obtain the TLS certificate

- Log in the Entrust Certificate Services interface.
- Select **Create > PKIaaS**.
The Select **Certificate Authority** pane appears.
- From the **Certificate Authority** drop-down list, select the CA you want to issue the TLS certificate.
- From the **Certificate Profile** drop-down list, select the certificate profile you want to use for the TLS certificate. The certificate profile must include Digital Signature for TLS certificates.
- Click **Next**.
The **Certificate Details** pane appears.
- In the **Subject DN** field, enter a value for the certificate's subject DN. The value should be the DNS name of the domain controller. For example, `cn=activedirectory.example.com`.
- For **Certificate Expiry**, provide an expiry date for TLS certificate. It is recommended that the TLS certificate be valid for 1 year or less.
- Under **Subject Alternative Names**, add one or more DNS Name components to the Subject Alternative Name (subjectAltName) extension in the certificate. The subjectAltName extension must have a DNS Name component for each DNS name that may be used by the domain controller.
To add a DNS Name component the Subject Alternative Name extension:
 - For **SAN type**, select **DNS Name**.
 - In the **Value** field, enter a DNS name that may be used by the domain controller.
 - Click **Add** to add the DNS Name component to the Subject Alternative Name extension.
The component is added to the list of components in the Subject Alternative Name extension
 - To remove a component from the Subject Alternative Name extension, click **Remove** next to the extension that you want to remove.
- Copy the contents of the CSR you generated earlier, and paste the contents into the **Certificate Signing Request (CSR)** text box.
- Click **Submit**.
If the certificate is generated successfully, a success message appears.
- Click **Download the newly created certificate** to download the TLS certificate.

After processing the CSR, proceed to [Installing the Active Directory server certificate](#).

Issuing the Active Directory server certificate with an on-premises CA

If you are using Certificate Enrollment Gateway with an on-premises CA, you can use your existing CA tools to process the CSR and create the certificate.

- [Creating or recovering a user account for the Active Directory server certificate](#)
- [Processing the CSR for the Active Directory server certificate](#)

Creating or recovering a user account for the Active Directory server certificate

To issue a certificate for Active Directory, a user account for the certificate must exist in your on-premises CA. You must create a user account to issue the initial Active Directory server certificate. You must recover (reset) the user account to renew the Web server certificate.

To manually create or recover (reset) a user account, you can use an administration application such as Entrust Authority Security Manager Administration or the User Management Service (Entrust Administration Services).

When creating a new user account:

- It is recommended that you configure the user's name (using the directory naming attributes) to be the fully qualified domain name of the domain controller. For example, `activedirectory.example.com`.
- Select a 1-key-pair certificate type with a Dual Usage certificate definition that includes an Extended Key Usage extension with server authentication and client authentication, and a DomainController extension (OID 1.3.6.1.4.1.311.20.2). The certificate definition should also be assigned a certificate definition policy. For example, the Enterprise Domain Controller (ent_ad_dc) certificate type.
- For the Subject Alternative Name (SubjectAltName) extension, add a DNS Name component for each DNS name that may be used by Active Directory.

For information about creating or recovering user accounts, see the documentation for the client application.

Processing the CSR for the Active Directory server certificate

You can process the CSR for the Active Directory server certificate using the Profile Creation Utility. The Profile Creation Utility is a command line utility that can create and manage Entrust profiles for an on-premises Security Manager CA. You can use the Profile Creation Utility to process Certificate Signing Requests (CSRs) and generate certificates. The Profile Creation Utility is available as a separate software download for Entrust CA Gateway.

i When processing a CSR, the Profile Creation Utility will prompt you for the certificate definition required for the certificate. In Security Manager, that certificate definition for the user's certificate type must be assigned a certificate definition policy (user policy). If no certificate definition policy is assigned to the certificate definition you specify, an error will occur and the Profile Creation Utility will fail to process the CSR.


To download and install the Profile Creation Utility

1. Install a Java Development Kit (JDK) and set the `JAVA_HOME` environment variable.
2. Log in to Entrust TrustedCare (<http://trustedcare.entrust.com>).
3. Go to **PKI > Authority > CA Gateway** and click the latest version of the product.
4. Under software downloads, download the Profile Creation Utility for your preferred operating system:
 - `cagw-profilecreationutility-linux64-version.zip` for Linux 64-bit.
 - `cagw-profilecreationutility-win64-version.zip` for Windows 64-bit.
5. Extract the file contents of the ZIP file to a location on the computer.

To process the CSR using the Process Creation Utility

1. Obtain the CSR file along with the reference number and authorization code associated with the Security Manager user account.
When you create a user in Security Manager or set a user for key recovery, Security Manager generates a reference number and authorization code. You need these activation codes to process the CSR.
2. To process the CSR, the Profile Creation Utility requires an Entrust desktop profile (EPF file). the role associated with the profile requires the following permissions:
 - Under the **Certificates** permission category: permissions to administer the certificate category and certificate type of the certificate being issued.
 - Under the **Groups** permission category: **View** and permission to administer the group associated with the Security Manager user being issued the certificate.
 - Under the **Roles** permission category: **View** and permission to administer the role associated with the Security Manager user being issued the certificate.
 - Under the **Searchbase** permission category: **View** and permission to administer the searchbase associated with the Security Manager user being issued the certificate.
 - Under the **Users** permission category: **View** and **Perform PKIX** requests. Obtain the Entrust desktop profile (EPF file) from a Security Manager administrator.
3. Navigate to the directory containing the Profile Creation Utility.
4. Run the following command:
 - On Windows, run `pcu.bat`.
 - On Linux, run `pcu`.
5. The Profile Creation Utility main menu appears:

```
Main Menu
1. Exit
2. Help
3. Create Entrust profile
4. Recover Entrust profile
5. Inspect Entrust profile (read only)
6. Inspect and update Entrust profile (read/write)
7. Create Server Login credentials
8. Create PKCS #12 file (Security Manager)
9. Recover PKCS #12 file (Security Manager)
10. Create PKCS #12 file (3rdParty)
11. Update PKCS #12 file (3rdParty)
12. Process PKCS #10 Certificate Signing Request (CSR)
13. Generate/Process Certificate Signing Request on HSM (3rdParty)
14. Change password
Select an operation [3]:
```

 To return to the main menu at any time, enter a period (.). For help about using the Profile Creating Utility, enter 2 in the main menu.

Enter 12 to process the CSR.

6. The following prompt appears:

```
Take settings from an existing entrust.ini file (y/n) [y]:
```

- To use Certificate Authority (CA) connection settings from an existing `entrust.ini` file, enter `y`.
- To provide CA connection settings manually, enter `n`.

7. If you chose to use an existing `entrust.ini` file, you are prompted to enter the full path to the `entrust.ini` file:

Enter full path to `entrust.ini` file:

Enter the full path and file name of the `entrust.ini` file.

8. If you chose to enter CA connections setting manually, the following prompts appear:
 - a. You are prompted to provide the host name (or IP address) and port of the CA server:

Enter the CA hostname or IP address and port in the form `name:port`:

Enter the host name (or IPv4 address) and CMP port of the server hosting the CA in format of `<hostname>:<port>`. If you omit the port number, it defaults to 829.

- b. You are prompted to provide the host name (or IP address) and port of the directory server:

Enter the directory hostname or IP address and port in the form `name:port`:

Enter the host name (or IPv4 address) and LDAP port of the server hosting the directory in format of `<hostname>:<port>`. The name or address defaults to the same value that you entered for the CA address. If you omit the port number, it defaults to 389.

9. You are prompted for the full path to an administration profile:

Enter full path to administration profile:

Enter the full path and file name of an administration profile.

10. You are prompted to enter the profile password:

Enter profile password:

Enter the profile password.

11. You are asked if the CSR is authenticated:

Is the CSR authenticated? (y/n)? [n]:

Enter `n`. The CSR is not authenticated.

12. You are prompted for the full path to the CSR:

Enter full path to CSR:

Enter the full path and file name of the CSR.

13. You are prompted to enter the reference number for the CSR:

Enter reference number:

Enter the reference number you recorded earlier.

14. You are prompted to enter the authorization code for the CSR:

Enter authorization code:

Enter the authorization code you recorded earlier.

15. You are prompted to enter a file name for the certificate:

Enter certificate file to create:

Enter the full path and file name for the certificate file.

16. You are prompted to enter the certificate definition required for the certificate:

Enter certificate definition required [Verification]:

Enter the certificate definition required for the certificate, such as Verification or Dual Usage.

17. The Profile Creation Utility processes the certificate. If the operation is successful, Security Manager issues a certificate and the Profile Creation Utility writes the certificate to a file.

```
Requesting certificate from Security Manager...
Obtained new certificate with serial number 1340207625 from issuer
o=Example,c=US
Certificate written to c:\new_certificate.cer
```

After processing the CSR and obtaining the certificate, proceed to [Installing the Active Directory server certificate](#).

Installing the Active Directory server certificate

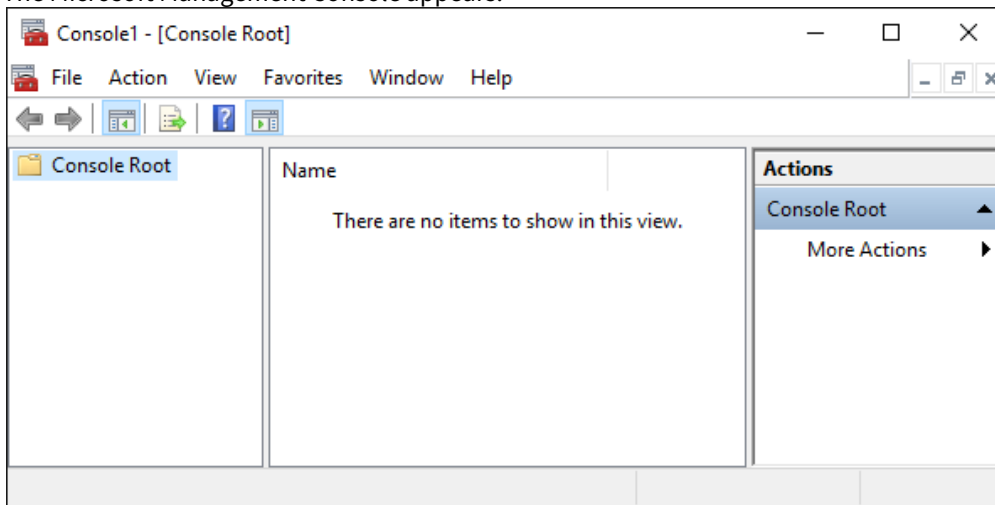
After obtaining the Active Directory server certificate, you must install the certificate into Active Directory. After processing the CSR with Entrust PKI as a Service or an on-premises CA, complete the certificate request to install the certificate into Active Directory. When the certificate is installed, LDAPS is automatically enabled in Active Directory.

To complete a certificate request and install the Active Directory server certificate

1. Log into Active Directory as a member of the Domain Admins group.

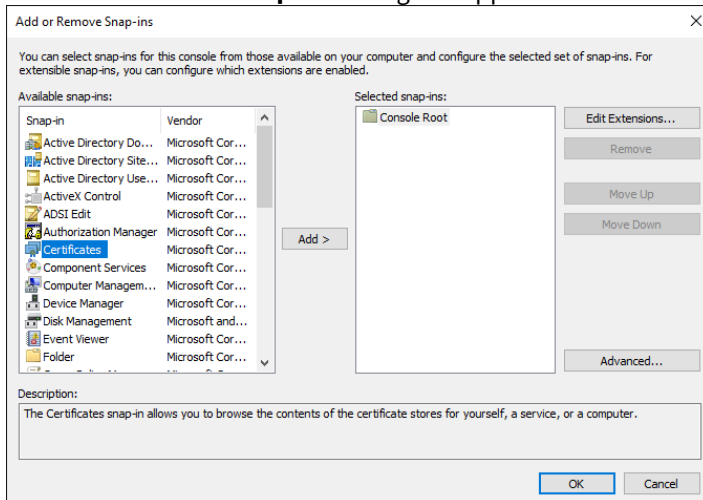
2. Run `mmc.exe` (Select **Start > Windows System > Run**, then enter `mmc.exe`).

The Microsoft Management Console appears.



3. Select **File > Add/Remove Snap-in**.

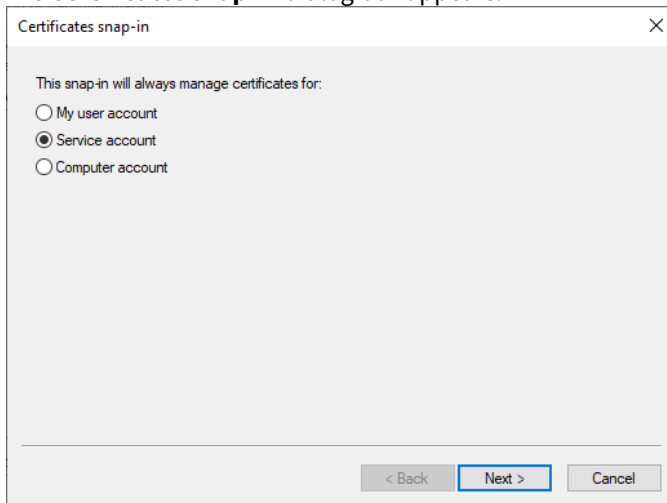
The **Add or Remove Snap-ins** dialog box appears.



4. In the **Available snap-ins** list, select **Certificates**.

5. Click **Add**.

The **Certificates snap-in** dialog box appears.



Certificates snap-in

This snap-in will always manage certificates for:

My user account

Service account

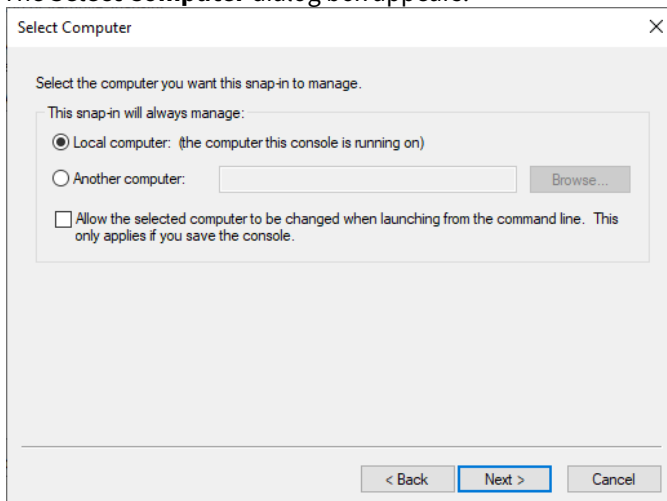
Computer account

< Back Next > Cancel

6. Select **Service account**.

7. Click **Next**.

The **Select Computer** dialog box appears.



Select Computer

Select the computer you want this snap-in to manage.

This snap-in will always manage:

Local computer: (the computer this console is running on)

Another computer: Browse...

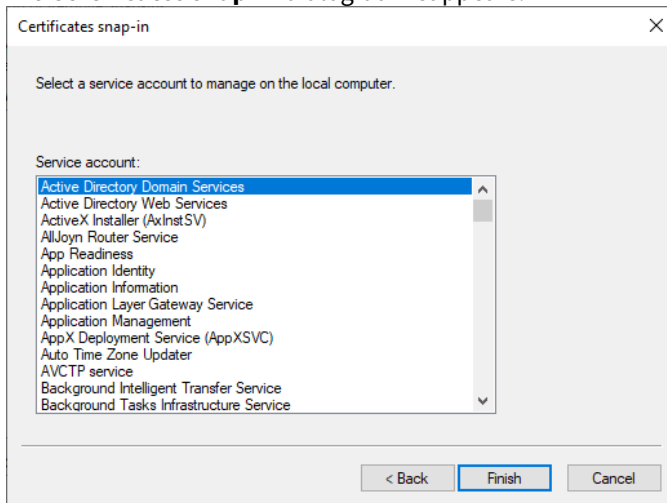
Allow the selected computer to be changed when launching from the command line. This only applies if you save the console.

< Back Next > Cancel

8. Select **Local computer**.

- Click **Next**.

The **Certificates snap-in** dialog box reappears.



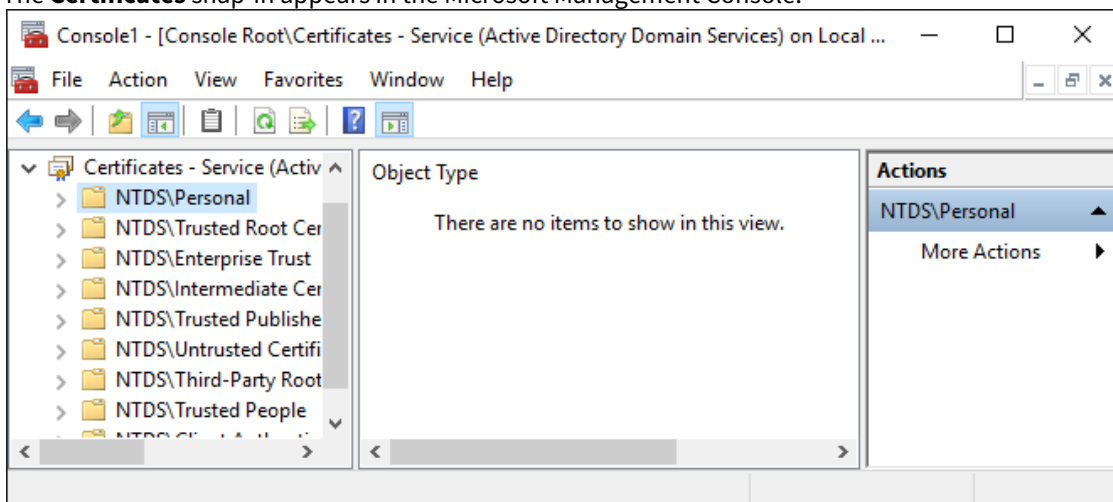
- Select **Active Directory Domain Services**.

- Click **Finish**.

The **Certificates** snap-in is added to the list of Selected snap-ins.

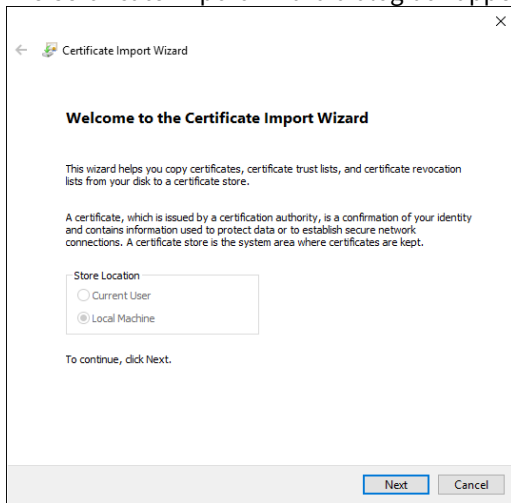
- Click **OK**.

The **Certificates** snap-in appears in the Microsoft Management Console.

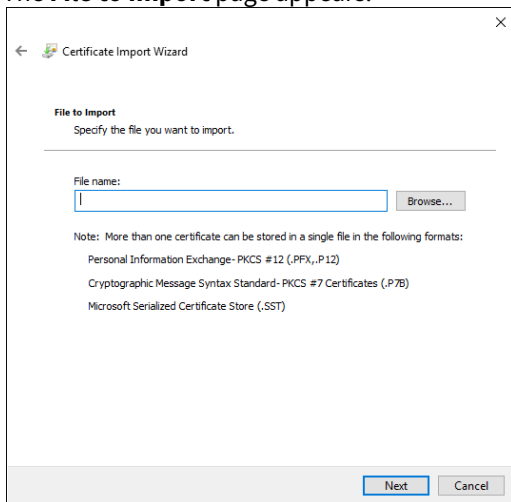


- In the tree view, select **Certificates > NTDS\Personal**.

14. Select **Action > All Tasks > Import**.
The Certificate Import Wizard dialog box appears.



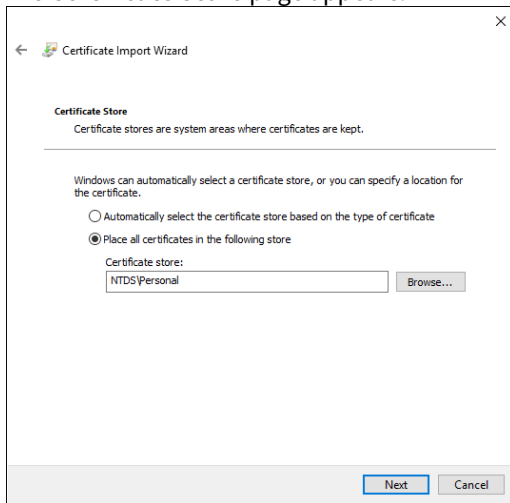
15. Click **Next**.
The **File to Import** page appears.



16. Click **Browse** and then select the Active Directory server certificate.

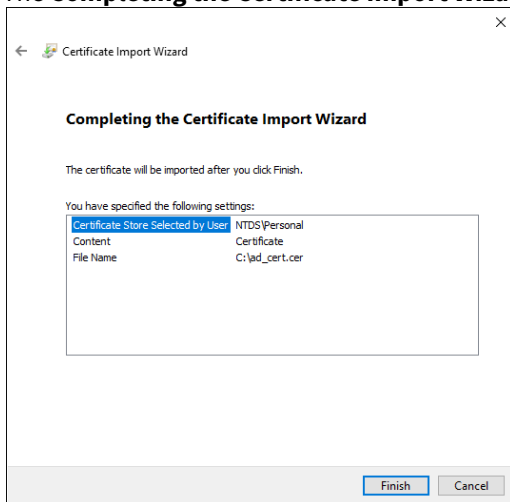
17. Click **Next**.

The **Certificate Store** page appears.



18. The NTDS\Personal certificate store is already selected. Click **Next**.

The **Completing the Certificate Import Wizard** page appears.



19. Click **Finish**.

Verifying LDAPS in Active Directory

After installing the server certificate into Active Directory, test that LDAPS was enabled.

To verify LDAPS in Active Directory

1. Start the Active Directory Administration Tool (`Ldp.exe`):
 - a. Select **Start > Windows System > Run**.
 - b. Enter `ldp.exe`.
2. Select **Connection > Connect**.
The **Connect** dialog box appears.
3. In the **Server** field, enter the domain controller to which you want to connect.
4. In the **Port** field, enter 636.
5. Click **OK**.

Preparing to install the Certificate Enrollment Policy Web Service

The Certificate Enrollment Policy Web Service will authenticate and forward WSTEP enrollment requests to Certificate Enrollment Gateway. The following topics describe how to prepare a server for the Certificate Enrollment Policy Web Service.

- [Installing a server for the Certificate Enrollment Policy Web Service](#)
- [Installing Microsoft Internet Information Services](#)

Installing a server for the Certificate Enrollment Policy Web Service

The Windows server that will host the Certificate Enrollment Policy Web Service can be the domain controller or any other server in the domain. However, it is recommended that you install and configure the Certificate Enrollment Policy Web Service on a different server than the domain controller.

Install a supported version of Microsoft Windows Server (if not already installed), and join it to the Windows domain. Recommended system requirements for the Windows server:

- Windows Server 2016 or later
- Minimum 4 GB of RAM
- Minimum 2 CPU
- Minimum 40 GB free hard disk space

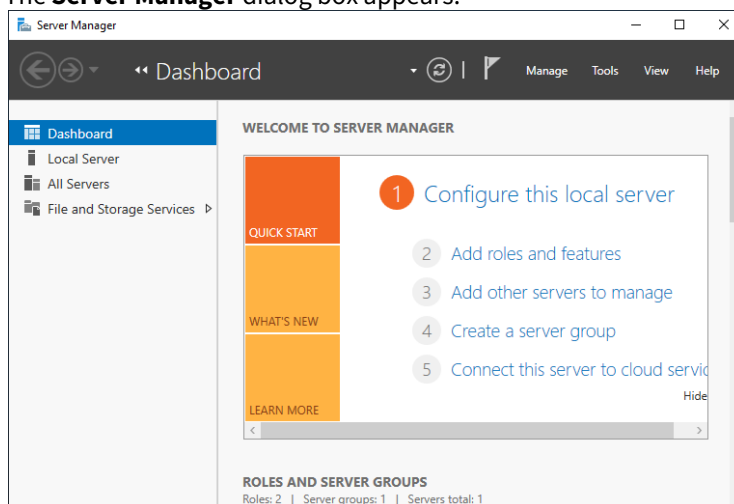
Installing Microsoft Internet Information Services

Microsoft Internet Information Services (IIS) is a Web server. The Certificate Enrollment Policy Web Service will be installed as an application in Microsoft IIS. The following procedure describes how to install Microsoft IIS.

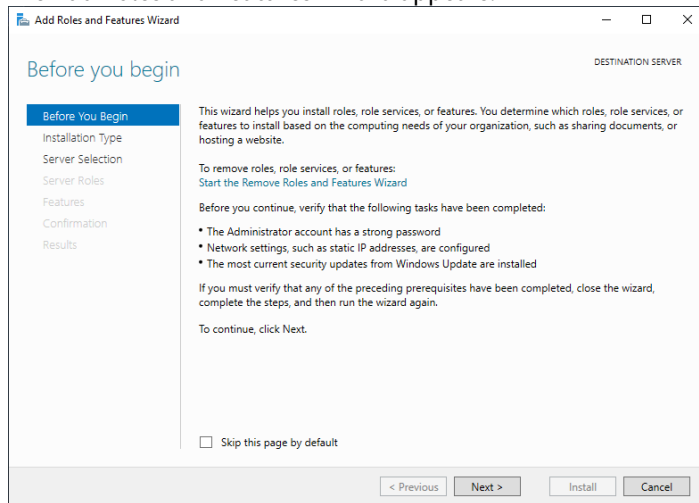
To install Microsoft Internet Information Services

1. Log in to the server that will host the Certificate Enrollment Policy Web Service.
2. Open Server Manager. Select **Start > Server Manager**.

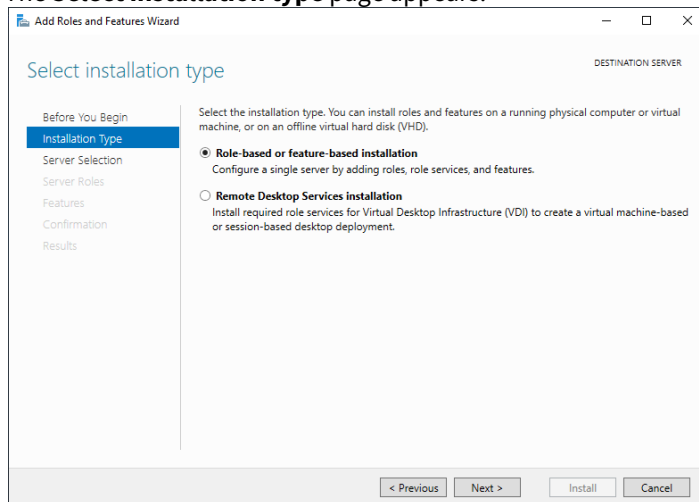
The **Server Manager** dialog box appears.



3. In the top navigation bar, select **Manage > Add Roles and Features**. The Add Roles and Features Wizard appears.



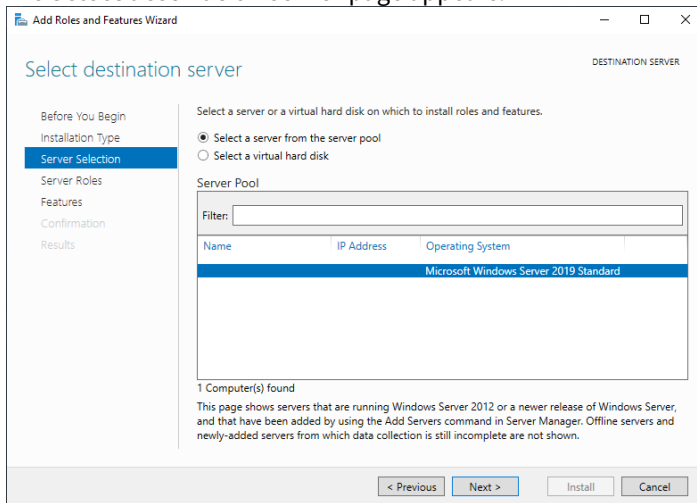
4. Click **Next**. The **Select installation type** page appears.



5. Click **Role-based or feature-based installation**. By default, this option is already selected.

6. Click **Next**.

The **Select destination server** page appears.

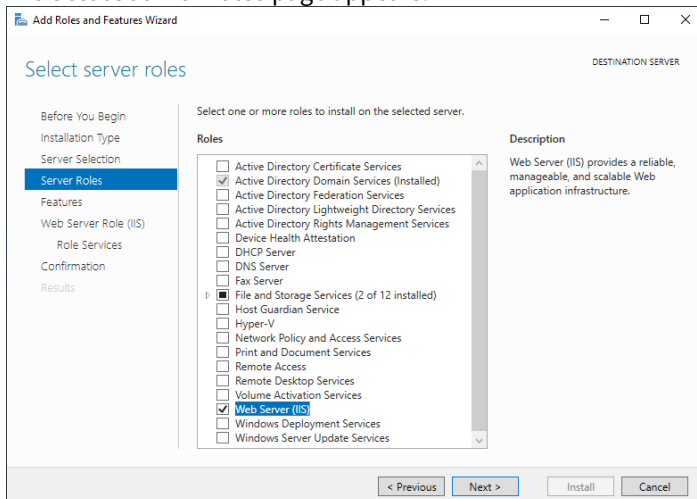


7. Click **Select a server from the server pool**. By default, this option should already be selected.

8. In the **Server Pool** list, select your server. By default, the server should already be selected.

9. Click **Next**.

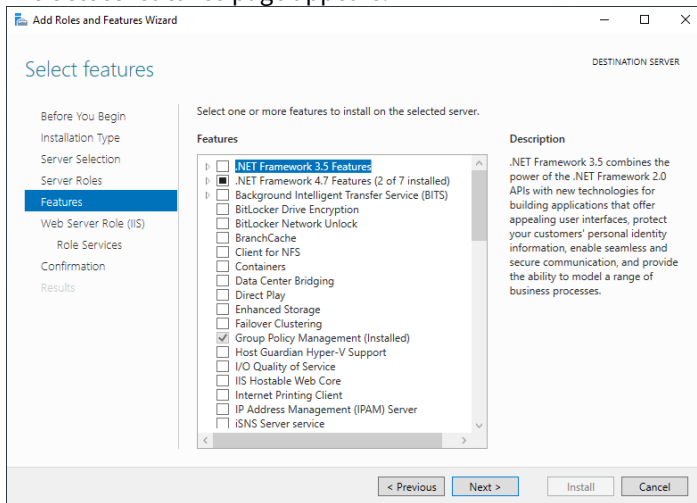
The **Select server roles** page appears.



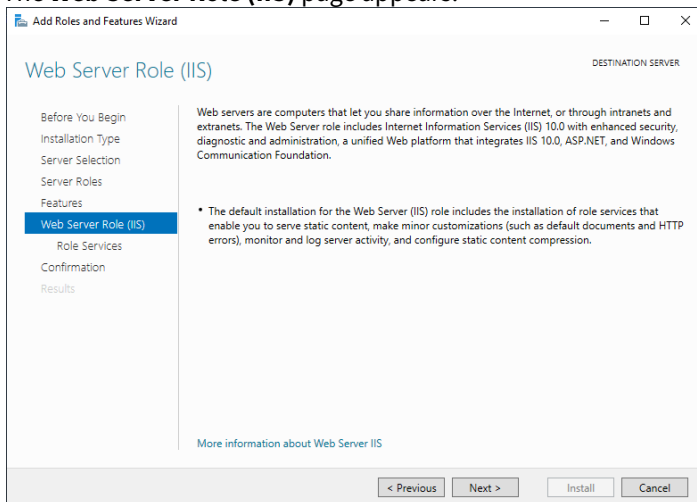
10. Select **Web Server (IIS)**.

11. A dialog box may appear, informing you that some features are required for Microsoft IIS. Click **Add Features** to add these required features and close the dialog box.

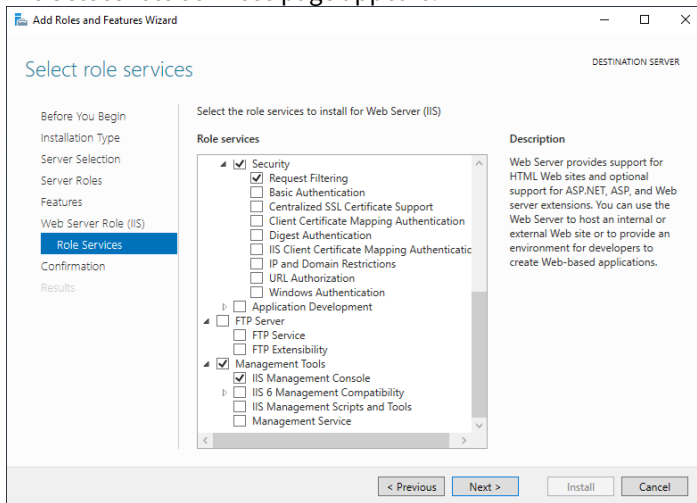
12. Click **Next**.
The **Select features** page appears.



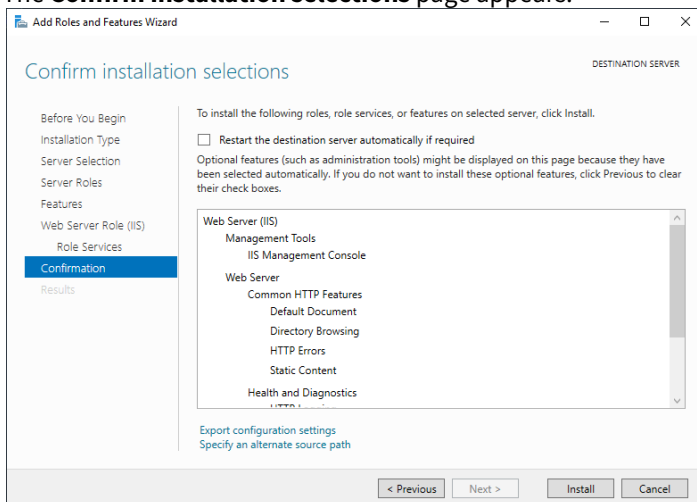
13. You do not need to add any features on this page. Click **Next**.
The **Web Server Role (IIS)** page appears.



14. Click **Next**.
The **Select role services** page appears.

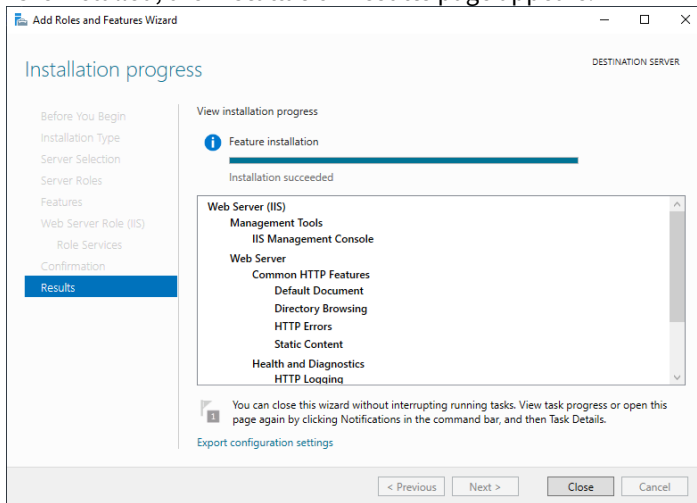


15. Under **Management Tools**, verify that **IIS Management Console** is selected.
16. Click **Next**.
The **Confirm installation selections** page appears.



17. Click **Install**.

18. The **Installation Progress** page appears. A progress indicator displays the progress of the installation. After IIS is installed, the **Installation results** page appears.



19. Click **Close**.

Issuing TLS certificates for the Certificate Enrollment Policy Web Service

The Certificate Enrollment Policy Web Service is installed as an application in Microsoft Internet Information Services (IIS). Microsoft IIS requires a TLS certificate so that the Certificate Enrollment Policy Web Service can accept WSTEP enrollment requests over HTTPS.

The following topics describe how to issue a TLS certificate for Microsoft IIS.

- [Creating a CSR for the Web server certificate](#)
- [Issuing the Web server certificate with an on-premises CA](#)
- [Issuing the Web server certificate with Entrust PKI as a Service](#)
- [Installing the Web server certificate into Microsoft IIS](#)
- [Updating Microsoft IIS to use the Web server certificate](#)
- [Installing the CA certificate chain for the Web server certificate](#)

Creating a CSR for the Web server certificate

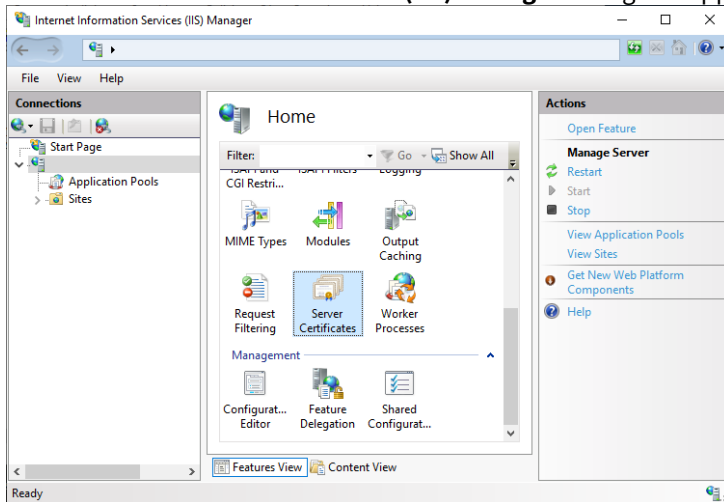
The Certificate Enrollment Policy Web Service is installed as an application in Microsoft Internet Information Services (IIS). Microsoft IIS requires a TLS certificate so that the Certificate Enrollment Policy Web Service can accept WSTEP enrollment requests over HTTPS.

The following procedure describes how to create a certificate signing request (CSR) in Microsoft IIS for a certificate. A CSR contains information that the issuing CA will use to create the certificate. Entrust PKI as a Service or an on-premises CA can process the CSR and issue the certificate.

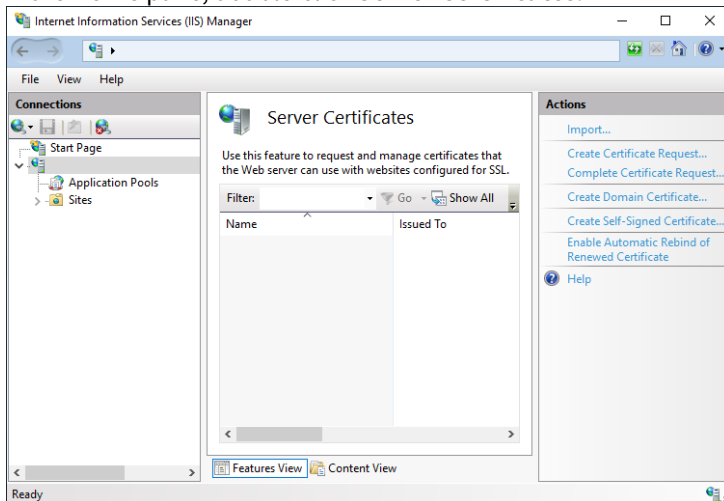
To create a CSR for Microsoft IIS

1. Open the Internet Information Services (IIS) Manager. Select **Start > Windows Administrative Applications > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.

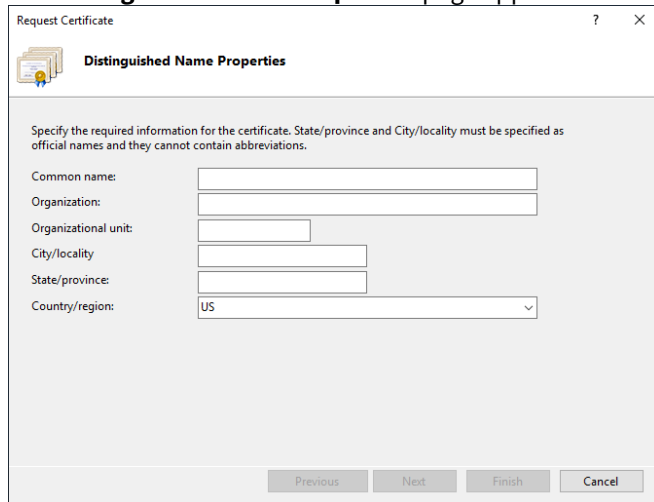


2. Under **Connections**, select the host name of the server.
3. In the **Home** pane, double-click **Server Certificates**.



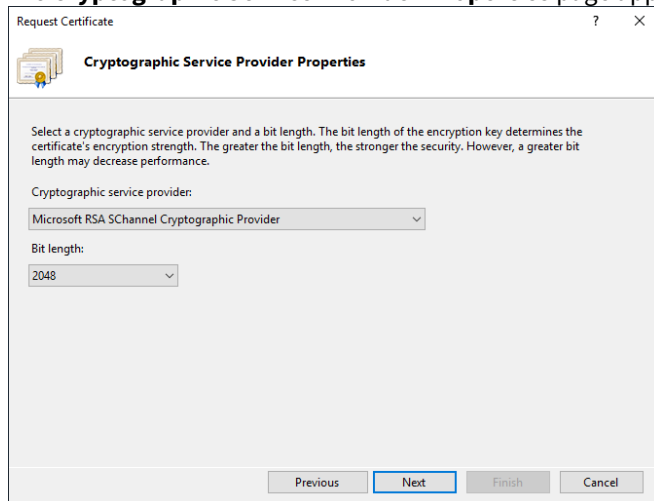
4. In the **Actions** pane, click **Create Certificate Request**.
The **Request Certificate** wizard appears.

5. The **Distinguished Name Properties** page appears.



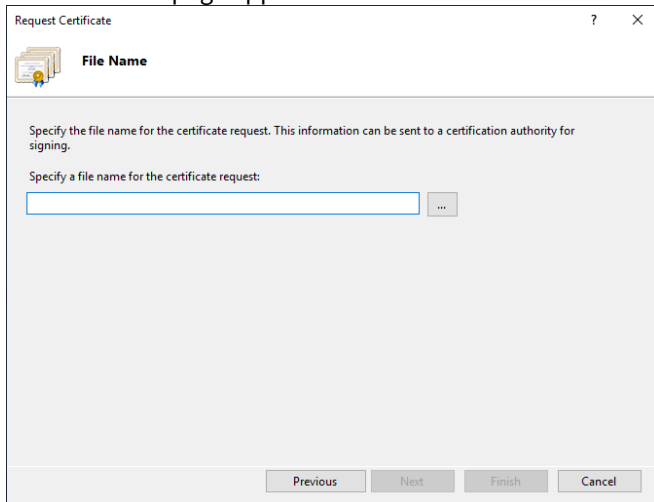
- In the provided fields, enter information that will be included in the CSR. Note that the issuing CA (either a CA in Entrust PKI as a Service, or an on-premises Entrust CA) will ignore this information.
- Click **Next**.

6. The **Cryptographic Service Provider Properties** page appears.



- From the **Cryptographic service provider** drop-down list, select a cryptographic service provider. It is recommended that you select **Microsoft RSA SChannel Cryptographic Provider**.
- In the **Bit length** list, select a bit length. It is recommended that you select 2048 as the bit length.
- Click **Next** to continue.

7. The **File Name** page appears.



- a. In the text field, enter a path and file name for the file that will contain the Web server certificate request.
- b. Click **Finish**.

8. The CSR is saved in the file you specified in the previous step.

Issuing the Web server certificate with an on-premises CA

If you are using Certificate Enrollment Gateway with an on-premises CA, you can use your existing CA tools to process the CSR and create the certificate.

- [Creating or recovering a user account for the Web server certificate](#)
- [Processing the CSR for the Web server certificate](#)

Creating or recovering a user account for the Web server certificate

To issue a certificate for the Web server, a user account for the certificate must exist in your on-premises CA. You must create a user account to issue the initial Web server certificate. You must recover (reset) the user account to renew the Web server certificate.

To manually create or recover (reset) a user account, you can use an administration application such as Entrust Authority Security Manager Administration or the User Management Service (Entrust Administration Services).

When creating a new user account:

- It is recommended that you configure the user's name (using the directory naming attributes) to be the fully qualified domain name of the Web server. For example, `example.com`.
- Select a 1-key-pair certificate type with a Dual Usage certificate definition that includes an Extended Key Usage extension with server authentication and client authentication. The certificate definition should also be assigned a certificate definition policy. For example, the Enterprise Machine (ent_machine) certificate type.
- For the Subject Alternative Name (SubjectAltName) extension, add a DNS Name component for each DNS name that may be used by the Web server.

For information about creating or recovering user accounts, see the documentation for the client application.

Processing the CSR for the Web server certificate

You can process the CSR for the Web server certificate using the Profile Creation Utility. The Profile Creation Utility is a command line utility that can create and manage Entrust profiles for an on-premises Security Manager CA. You

can use the Profile Creation Utility to process Certificate Signing Requests (CSRs) and generate certificates. The Profile Creation Utility is available as a separate software download for Entrust CA Gateway.

i When processing a CSR, the Profile Creation Utility will prompt you for the certificate definition required for the certificate. In Security Manager, that certificate definition for the user's certificate type must be assigned a certificate definition policy (user policy). If no certificate definition policy is assigned to the certificate definition you specify, an error will occur and the Profile Creation Utility will fail to process the CSR.

To download and install the Profile Creation Utility

1. Install a Java Development Kit (JDK) and set the `JAVA_HOME` environment variable.
2. Log in to Entrust TrustedCare (<http://trustedcare.entrust.com>).
3. Go to **PKI > Authority > CA Gateway** and click the latest version of the product.
4. Under software downloads, download the Profile Creation Utility for your preferred operating system:
 - `cagw-profilecreationutility-linux64-version.zip` for Linux 64-bit.
 - `cagw-profilecreationutility-win64-version.zip` for Windows 64-bit.
5. Extract the file contents of the ZIP file to a location on the computer.

To process the CSR using the Process Creation Utility

1. Obtain the CSR file along with the reference number and authorization code associated with the Security Manager user account.
When you create a user in Security Manager or set a user for key recovery, Security Manager generates a reference number and authorization code. You need these activation codes to process the CSR.
2. To process the CSR, the Profile Creation Utility requires an Entrust desktop profile (EPF file). The role associated with the profile requires the following permissions:
 - Under the **Certificates** permission category: permissions to administer the certificate category and certificate type of the certificate being issued.
 - Under the **Groups** permission category: **View** and permission to administer the group associated with the Security Manager user being issued the certificate.
 - Under the **Roles** permission category: **View** and permission to administer the role associated with the Security Manager user being issued the certificate.
 - Under the **Searchbase** permission category: **View** and permission to administer the searchbase associated with the Security Manager user being issued the certificate.
 - Under the **Users** permission category: **View** and **Perform PKIX** requests. Obtain the Entrust desktop profile (EPF file) from a Security Manager administrator.
3. Navigate to the directory containing the Profile Creation Utility.
4. Run the following command:
 - On Windows, run `pcu.bat`.
 - On Linux, run `pcu`.
5. The Profile Creation Utility main menu appears:

```
Main Menu
1. Exit
2. Help
3. Create Entrust profile
4. Recover Entrust profile
5. Inspect Entrust profile (read only)
6. Inspect and update Entrust profile (read/write)
```


7. Create Server Login credentials
 8. Create PKCS #12 file (Security Manager)
 9. Recover PKCS #12 file (Security Manager)
 10. Create PKCS #12 file (3rdParty)
 11. Update PKCS #12 file (3rdParty)
 12. Process PKCS #10 Certificate Signing Request (CSR)
 13. Generate/Process Certificate Signing Request on HSM (3rdParty)
 14. Change password
- Select an operation [3]:

i To return to the main menu at any time, enter a period (.). For help about using the Profile Creating Utility, enter 2 in the main menu.

Enter 12 to process the CSR.

6. The following prompt appears:

Take settings from an existing entrust.ini file (y/n) [y]:

- To use Certificate Authority (CA) connection settings from an existing `entrust.ini` file, enter `y`.
- To provide CA connection settings manually, enter `n`.

7. If you chose to use an existing `entrust.ini` file, you are prompted to enter the full path to the `entrust.ini` file:

Enter full path to entrust.ini file:

Enter the full path and file name of the `entrust.ini` file.

8. If you chose to enter CA connections setting manually, the following prompts appear:
 - a. You are prompted to provide the host name (or IP address) and port of the CA server:

Enter the CA hostname or IP address and port in the form name:port:

Enter the host name (or IPv4 address) and CMP port of the server hosting the CA in format of `<hostname>:<port>`. If you omit the port number, it defaults to 829.

- b. You are prompted to provide the host name (or IP address) and port of the directory server:

Enter the directory hostname or IP address and port in the form name:port:

Enter the host name (or IPv4 address) and LDAP port of the server hosting the directory in format of `<hostname>:<port>`. The name or address defaults to the same value that you entered for the CA address. If you omit the port number, it defaults to 389.

9. You are prompted for the full path to an administration profile:

Enter full path to administration profile:

Enter the full path and file name of an administration profile.

10. You are prompted to enter the profile password:

Enter profile password:

Enter the profile password.

11. You are asked if the CSR is authenticated:

Is the CSR authenticated? (y/n)? [n]:

Enter **n**. The CSR is not authenticated.

12. You are prompted for the full path to the CSR:

Enter full path to CSR:

Enter the full path and file name of the CSR.

13. You are prompted to enter the reference number for the CSR:

Enter reference number:

Enter the reference number you recorded earlier.

14. You are prompted to enter the authorization code for the CSR:

Enter authorization code:

Enter the authorization code you recorded earlier.

15. You are prompted to enter a file name for the certificate:

Enter certificate file to create:

Enter the full path and file name for the certificate file.

16. You are prompted to enter the certificate definition required for the certificate:

Enter certificate definition required [Verification]:

Enter the certificate definition required for the certificate, such as Verification or Dual Usage.

17. The Profile Creation Utility processes the certificate. If the operation is successful, Security Manager issues a certificate and the Profile Creation Utility writes the certificate to a file.

```
Requesting certificate from Security Manager...
Obtained new certificate with serial number 1340207625 from issuer
o=Example,c=US
Certificate written to c:\new_certificate.cer
```

After processing the CSR and obtaining the certificate, proceed to [Installing the Web server certificate into Microsoft IIS](#).

Issuing the Web server certificate with Entrust PKI as a Service

After creating the certificate signing request (CSR) for the Certificate Enrollment Gateway certificate, you can submit the CSR to an Issuing CA in Entrust PKI as a Service. The Issuing CA will process the CSR and generate the certificate.

To submit the CSR to Entrust PKI as a Service and obtain the TLS certificate

1. Log in the Entrust Certificate Services interface.
2. Select **Create > PKIaaS**.
The Select **Certificate Authority** pane appears.
3. From the **Certificate Authority** drop-down list, select the CA you want to issue the TLS certificate.
4. From the **Certificate Profile** drop-down list, select the certificate profile you want to use for the TLS certificate. The certificate profile must include Digital Signature for TLS certificates.
5. Click **Next**.
The **Certificate Details** pane appears.
6. In the **Subject DN** field, enter a value for the certificate's subject DN. The value should be the DNS name of the server hosting Microsoft IIS. For example, `cn=example.com`.
7. For **Certificate Expiry**, provide an expiry date for TLS certificate. It is recommended that the TLS certificate be valid for 1 year or less.
8. Under **Subject Alternative Names**, add one or more DNS Name components to the Subject Alternative Name (subjectAltName) extension in the certificate. The subjectAltName extension must have a DNS Name component for each DNS name that may be used by the server hosting Microsoft IIS.
To add a DNS Name component the Subject Alternative Name extension:
 - a. For **SAN type**, select **DNS Name**.
 - b. In the **Value** field, enter a DNS name that may be used by the server hosting Microsoft IIS.
 - c. Click **Add** to add the DNS Name component to the Subject Alternative Name extension.
The component is added to the list of components in the Subject Alternative Name extension
 - d. To remove a component from the Subject Alternative Name extension, click **Remove** next to the extension that you want to remove.
9. Copy the contents of the CSR you generated earlier, and paste the contents into the **Certificate Signing Request (CSR)** text box.
10. Click **Submit**.
If the certificate is generated successfully, a success message appears.
11. Click **Download the newly created certificate** to download the TLS certificate.

After processing the CSR, proceed to [Installing the Web server certificate into Microsoft IIS](#).

Installing the Web server certificate into Microsoft IIS

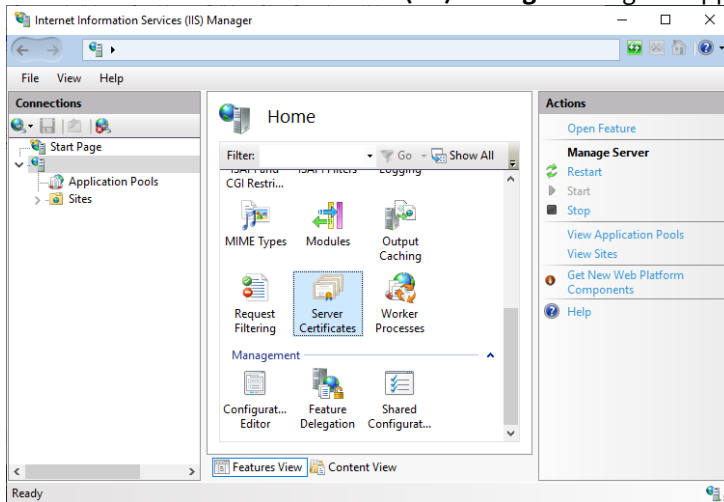
After obtaining the Web server certificate, you must install the certificate into Microsoft IIS. After processing the CSR with Entrust PKI as a Service or an on-premises CA, complete the certificate request in Microsoft IIS to install the certificate.

i If you will use the Entrust-provided PowerShell scripts to install and configure the Certificate Enrollment Policy Web Service, keep the certificate file after installing the certificate into Microsoft IIS. The PowerShell script will prompt you to select an existing certificate or supply a new certificate for the Certificate Enrollment Policy Web Service. If the PowerShell script cannot find the certificate you install into Microsoft IIS, you must supply the certificate.

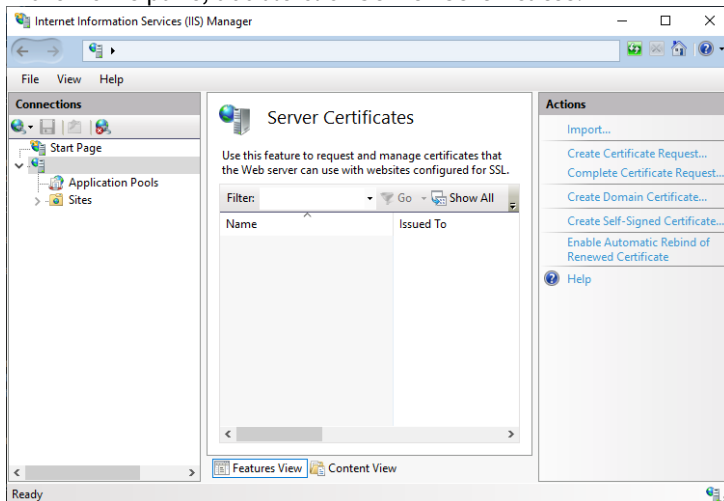
To complete a certificate request in Microsoft IIS

1. Open the Internet Information Services (IIS) Manager. Select **Start > Windows Administrative Applications > Internet Information Services (IIS) Manager**.

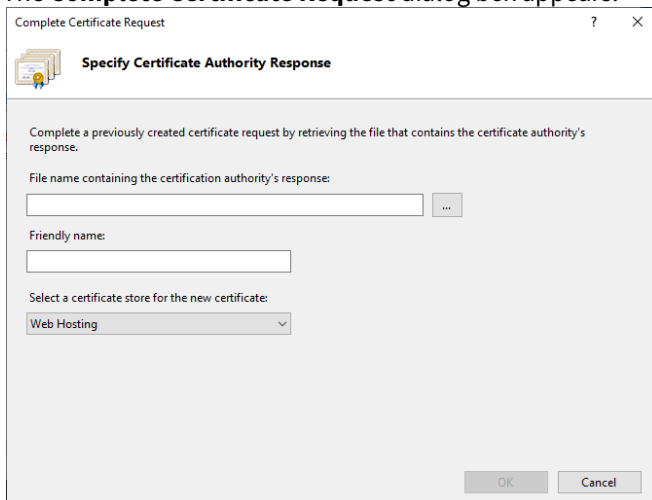
The **Internet Information Services (IIS) Manager** dialog box appears.



2. Under **Connections**, select the host name of the server.
3. In the **Home** pane, double-click **Server Certificates**.



4. In the **Actions** pane, click **Complete Certificate Request**.
The **Complete Certificate Request** dialog box appears.



5. In the **File name containing the certification authority response** field, enter the full path and file name of the Web server certificate, or click the Browse button to select the certificate.
6. In the **Friendly name** field, enter a friendly name for the certificate. This friendly name will be used to identify the certificate in the IIS Manager interface.
7. From the **Select a certificate store for the new certificate** drop-down list, select **Web Hosting**.
8. Click **OK**.
In the **Server Certificates** pane, the certificate appears in the list of server certificates. The certificate is listed by its friendly name.

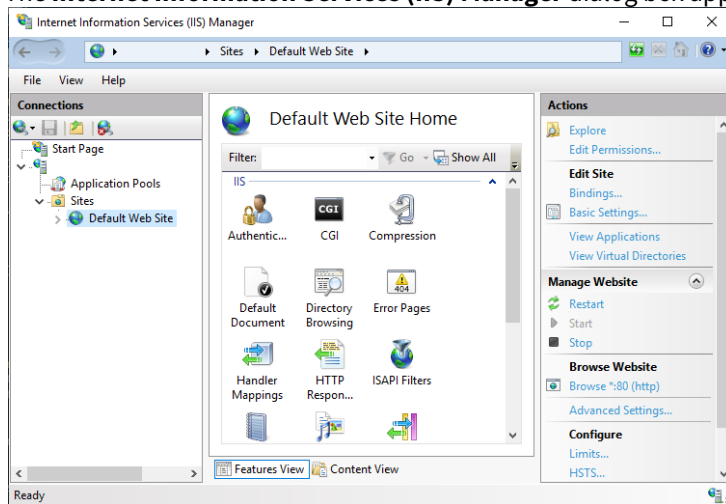
Updating Microsoft IIS to use the Web server certificate

After installing the Web server certificate into Microsoft IIS, you must update the HTTPS site binding for the Web server in Microsoft IIS to use the TLS certificate.

To add an initial HTTPS site binding in Microsoft IIS to use the Web server certificate

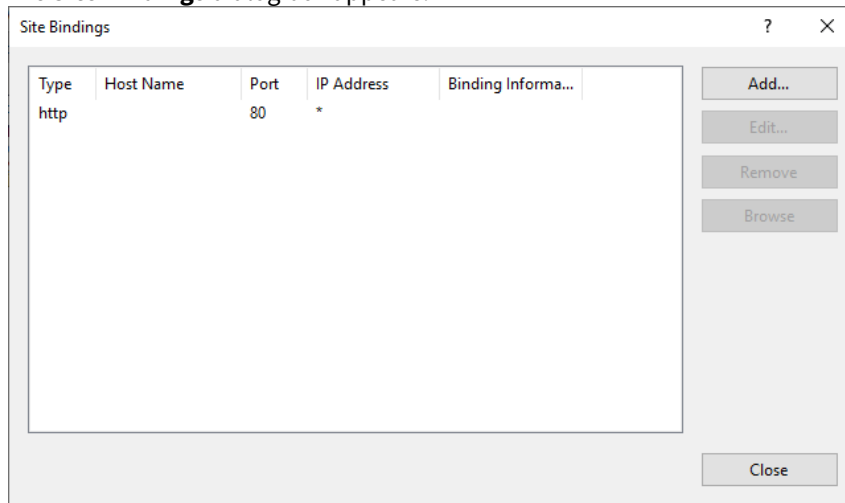
1. Open IIS Manager. Select **Start > Windows Administrative Tools > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.

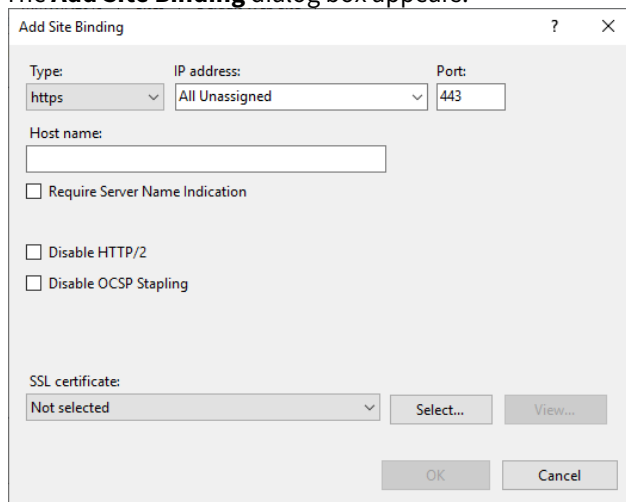


2. In the **Connections** pane, select **Sites > Default Web Site**.
The Certificate Enrollment Policy Web Service will be installed as an application under the default Web site.

- In the **Actions** pane, click **Bindings**.
The **Site Bindings** dialog box appears.



- Click **Add**.
The **Add Site Binding** dialog box appears.

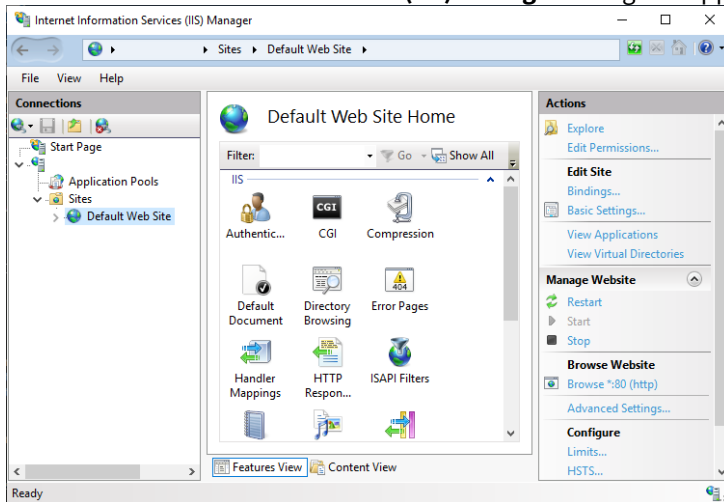


- In the **Type** drop-down list, select **https**.
- Keep **IP address** as **All Unassigned**.
- For **Port**, keep the default 443.
- From the **SSL certificate** drop-down list, select the Web server certificate you installed into Microsoft IIS earlier.
- Click **OK**.
The HTTPS binding is added to the list of site bindings.

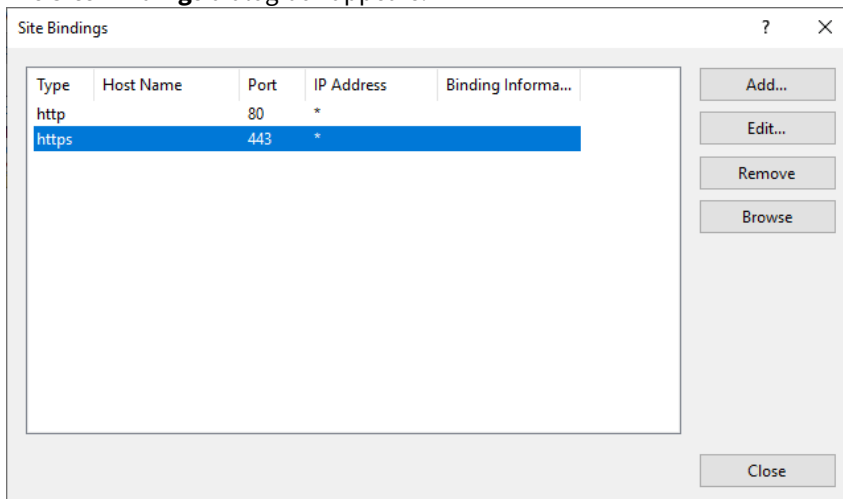
To update an existing HTTPS site binding in Microsoft IIS to use the Web server certificate

- Open IIS Manager. Select **Start > Windows Administrative Tools > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.



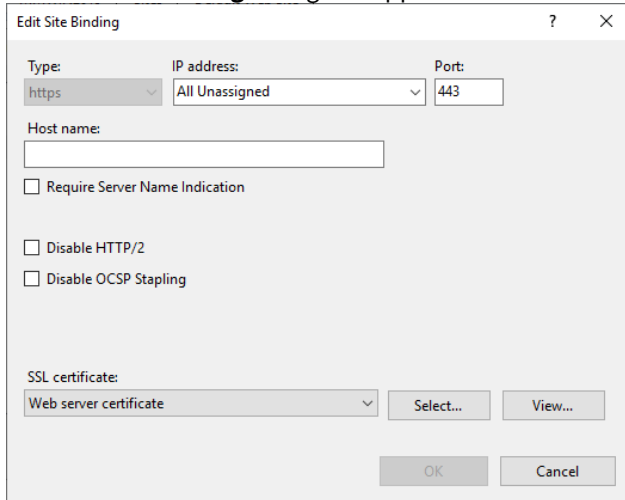
2. In the **Connections** pane, select **Sites > Default Web Site**.
If the Certificate Enrollment Policy Web Service is installed, it appears as an application under the default Web site.
3. In the **Actions** pane, click **Bindings**.
The **Site Bindings** dialog box appears.



4. Select the **https** binding for port **443**.

5. Click **Edit**.

The **Edit Site Binding** dialog box appears.



6. From the **SSL certificate** drop-down list, select the Web server certificate you installed into Microsoft IIS earlier.
7. Click **OK**.

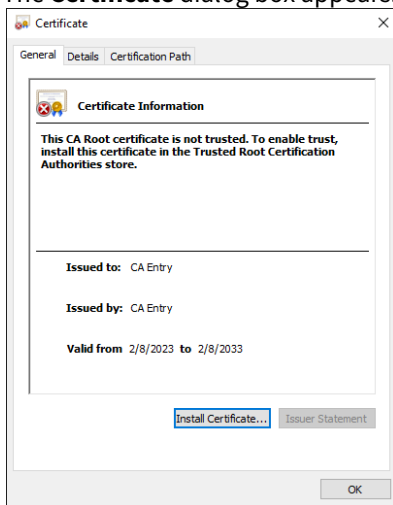
Installing the CA certificate chain for the Web server certificate

For Microsoft IIS to trust the Web server certificate, you must install the CA certificate chain for the Web server certificate into the server hosting Microsoft IIS. You must install the entire CA certificate chain, from the root CA to the issuing CA (the CA that issued the Web server certificate). For an on-premises CA, the root CA may be the issuing CA.

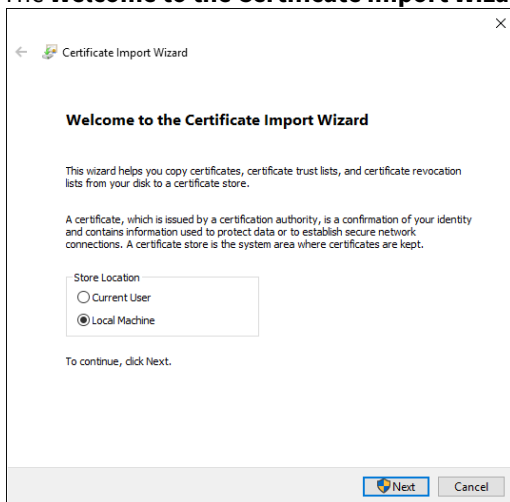
To install a CA certificate into the Web server

1. For an on-premises CA, obtain all CA certificates in the CA certificate chain using your on-premises CA tools. See the documentation for your on-premises CA for instructions.
2. For Entrust PKI as a Service, download all CA certificates in the certificate chain:
 - a. Log in to the Entrust Certificate Services interface.
 - b. Select **Administration > PKIaaS Management**. A list of private CAs appear.
 - c. For each CA in the TLS certificate chain (from the Issuing CA to the Root CA), select the CA and then click **Download certificate**.

3. Double-click the CA certificate file.
The **Certificate** dialog box appears.

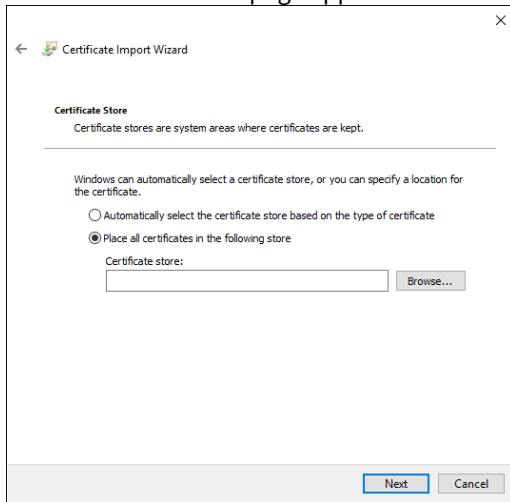


4. Click **Install Certificate**.
The Certificate Import Wizard appears.
5. The **Welcome to the Certificate Import Wizard** page appears.



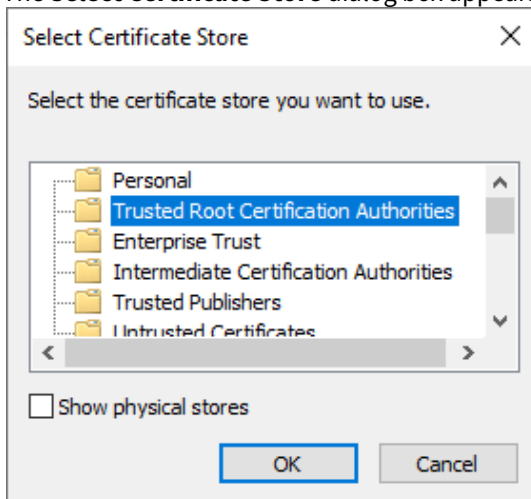
- a. For **Store Location**, select **Local Machine**.
- b. Click **Next**.

6. The **Certificate Store** page appears.



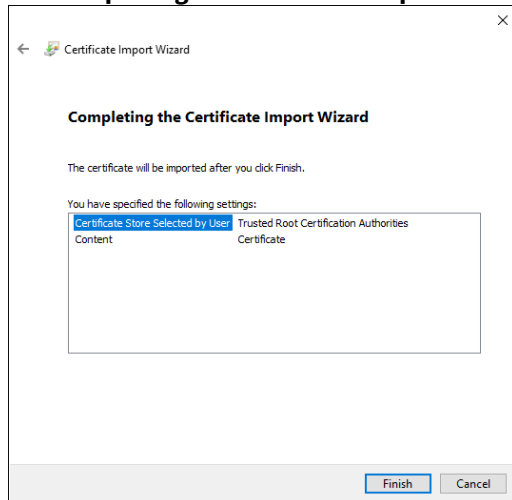
- a. Select **Place all certificates in the following store.**
- b. Click **Browse.**

The **Select Certificate Store** dialog box appears.



- c. If the CA certificate is a root CA certificate, select **Trusted Root Certification Authorities.**
- d. If the CA certificate is a subordinate (intermediate) CA certificate, select **Intermediate Certification Authorities.**
- e. Click **OK.**
- f. Click **Next.**

7. The **Completing the Certificate Import Wizard** page appears.



8. Click **OK**.

Installing and configuring the Certificate Enrollment Policy Web Service

The Certificate Enrollment Policy Web Service allows enrollment clients to retrieve certificate enrollment policies from a Certificate Authority (CA) when the clients are not permitted to access the Domain Controller. After receiving policy information from the Certificate Enrollment Policy Web Service, enrollment clients can then request a certificate from a certificate enrollment service.

In the Windows domain, install and configure the Certificate Enrollment Policy Web Service to forward WSTEP requests to Certificate Enrollment Gateway.

- It is recommended that you programmatically install and configure the service as explained in [Installing and configuring the CEP Web Service using a PowerShell script](#).
- However, you can also perform these operations manually, as explained in [Installing and configuring the CEP Web Service using the Windows graphical interface](#).

Installing and configuring the CEP Web Service using a PowerShell script

It is recommended that you install and configure the Certificate Enrollment Policy Web Service using the `InstallConfigCEP.ps1` PowerShell script provided by Entrust. This script requires the TLS certificate previously obtained in [Issuing TLS certificates for the Certificate Enrollment Policy Web Service](#).

To install and configure the Certificate Enrollment Policy Web Service using a PowerShell script

1. Install a supported version of Microsoft Windows Server (if not already installed), and join it to the Windows domain.
2. From Entrust TrustedCare, download the PowerShell scripts for Certificate Enrollment Gateway.
3. Extract the PowerShell scripts to a directory on the server.
4. PowerShell scripts downloaded from the Internet may be blocked from running. To unblock a PowerShell script:
 - a. Right-click the PowerShell script > **Properties**.
A **Properties** dialog box appears.
 - b. Under the **General** tab, click **Unblock**.
5. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
6. Navigate to the directory where you extracted the PowerShell scripts.

7. Enter the following command to run the

```
InstallConfigCEP.ps1
```

script:

```
.\InstallConfigCEP.ps1
```

The script validates the pre-requisites, and then installs any required Windows packages or features. For example:

i The PowerShell script was tested on specific versions of PowerShell. When validating the prerequisites, the PowerShell version may be listed as Unverified, an "Unverified" version of PowerShell indicates that the script was not tested on that version of PowerShell. You can still use the script on an "Unverified" version of PowerShell.

```
Validating pre-requisites:
Script-Mode: Windows
Script Version: 1.5.1.19
  - Member of Domain:           Verified
  - Domain Admins privileges:   Verified
  - Enterprise Admins privileges: Verified
  - Windows Version:           Verified (Microsoft Windows NT 10.0.17763.0)
  - PowerShell Version:        Verified (5.1.17763.2931)
```

```
-----
Installing ADCS-Enroll-Web-Pol
ADCS-Enroll-Web-Pol installed
```

```
Checking for Web-Mgmt-Console
Installing Web-Mgmt-Console
```

```
Checking for Web-Mgmt-Compat
Installing Web-Mgmt-Compat
```

8. The script prompts you to select the authentication type:

```
CEP Authentication Setting
Choices :

Name           Value
----           -
UserName      4
Kerberos      2

Select Authentication Type [Default: 2]:
```

- To select user name and password authentication, enter 4. User name and password authentication is the only authentication mode supported by non-domain enrollment endpoints.
- To select Kerberos (Windows integrated) authentication, enter 2.

9. The Certificate Enrollment Policy Web Service requires a certificate. The script prompts you to select a certificate option:

```
A CEP webserver certificate has not been selected.

Choose from the following Options:
[S] Select Existing Cert  [N] Supply New Certificate  [C] Continue with
selected Certificate  [E] Exit  [?] Help
(default is "S"):
```

- To select an existing certificate, enter **S** .
 - To supply a new certificate, enter **N** .
10. If you chose to select an existing certificate:
- a. The script will search the server for existing TLS Web certificates.
The script will first search for a certificate assigned to the Default Web Site in Microsoft IIS. If no certificate is assigned to the Default Web Site, the script will then search the certificate store of the local computer for valid (not expired) certificates with the following:
 - a subject name with the fully qualified domain name (FQDN) of the host
 - a private key
 - an extended key usage of Server Authentication
 - b. If the script finds a valid certificate, it asks if you want to use the certificate. For example:

```
Searching for Existing Certificate(s)

A certificate was found.
Subject       : CN=ceusername.example.com
Issuer        : CN=ceusername.example.com
KeyUsage      : DataEncipherment, KeyEncipherment
EKU List      : Server Authentication (1.3.6.1.5.5.7.3.1)
DNS SAN       : ceusername.example.com
Serial Number : 37F2440E97A3AE8046AA54BD7227FAFC
Thumbprint    : 50C09642942060AE1A58C5C3006F2455B57326BC
Not After     : 09/05/2023 20:00:00
PolicyId      :
```

```
Continue with above Certificate? (y/n):
```

- To continue with the certificate found by the script, enter **y** .
 - To go back and provide a different certificate, enter **n** .
- c. If the script finds multiple valid certificates, it will prompt you to select a certificate. For example:

```
Searching for Existing Certificate(s)
More than one certificate with FQDN ceusername.example.com has been
found.
Certificate Index : 1
-----
```

```

Subject       : CN=ceusername.example.com
Issuer       : CN=ceusername.example.com
KeyUsage     : DataEncipherment, KeyEncipherment
EKU List     : Server Authentication (1.3.6.1.5.5.7.3.1)
DNS SAN      : ceusername.example.com
Serial Number : 1862326CAB4507B1411EA7624F6DDDBA
Thumbprint   : EEC5FF53EA64B1B56B8731A7E73C058257A4DC0E
Not After    : 09/06/2023 20:00:00
PolicyId     :
  
```

Certificate Index : 2

```

-----
Subject       : CN=ceusername.example.com
Issuer       : CN=ceusername.example.com
KeyUsage     : DataEncipherment, KeyEncipherment
EKU List     : Server Authentication (1.3.6.1.5.5.7.3.1)
DNS SAN      : ceusername.example.com
Serial Number : 37F2440E97A3AE8046AA54BD7227FAFC
Thumbprint   : 50C09642942060AE1A58C5C3006F2455B57326BC
Not After    : 09/05/2023 20:00:00
PolicyId     :
  
```

Please select the Index to select a Certificate. 0 to **return** to previous menu.:

- To select one of the existing certificates, enter the index number associated with the certificate.
- To go back and provide a different certificate, enter 0.

11. If you chose to supply a certificate:

- a. The script prompts you to provide the certificate:

Please enter full path to certificate file including the filename :

Enter the full path and file name of the certificate, in PFX or P12 format.

- b. When prompted, enter the password of the certificate file.
- c. If the supplied certificate has a subject that does not match the fully qualified domain name (FQDN) of the host, the script displays a warning and asks if you want to continue with the certificate. For example:

The supplied certificate has a subject that does not match the FQDN of **this** host.

Host FQDN : ceusername.example.com

Supplied certificate details:

```

-----
Subject       : CN=CEP Web Service, CN=CA Entry, O=Example, C=US
Issuer       : CN=CA Entry, O=Example, C=US
  
```

```
KeyUsage      : KeyEncipherment, DigitalSignature
EKU List      : Server Authentication (1.3.6.1.5.5.7.3.1) Client
Authentication (1.3.6.1.5.5.7.3.2)
DNS SAN       : cepusername.example.com
Serial Number : 6AFEE3C47A569F95A9C5622D679B42C1
Thumbprint    : 2E6601A98E2ADB4EBE5DF6D8C3A514CD7660BAD
Not After     : 09/06/2025 13:55:27
PolicyId      :
```

Continue with above Certificate? (y/n):

- To continue with the certificate, enter **y**.
- To go back and provide a different certificate, enter **n**.

12. After providing a certificate, the script displays information about the certificate, and prompts you to select a certificate option:

```
Selected Webserver Certificate for CEP:
Subject       : CN=cepusername.example.com
Issuer        : CN=cepusername.example.com
KeyUsage      : DataEncipherment, KeyEncipherment
EKU List      : Server Authentication (1.3.6.1.5.5.7.3.1)
DNS SAN       : cepusername.example.com
Serial Number : 1862326CAB4507B1411EA7624F6DDDBA
Thumbprint    : EEC5FF53EA64B1B56B8731A7E73C058257A4DC0E
Not After     : 09/06/2023 20:00:00
PolicyId      :
```

Choose from the following Options:

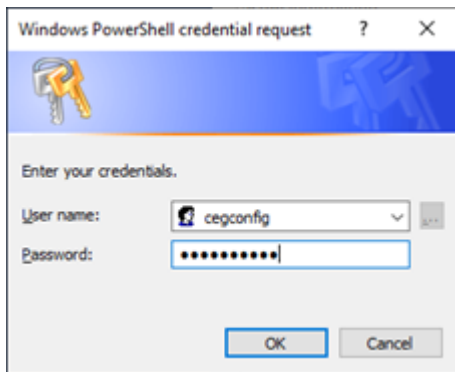
```
[S] Select Existing Cert [N] Supply New Certificate [C] Continue with
selected Certificate [E] Exit [?] Help
(default is "S"):
```

- To go back and select a different existing certificate, enter **S**.
- To go back and supply and different certificate, enter **N**.
- To continue with the selected certificate, enter **C**.

13. After providing a certificate, the script prompts you to provide a Windows user to configure the Certificate Enrollment Policy. The user must have Domain Admin and Enterprise Admin permissions.

```
Configuring CEP Service
Please enter the user information to be able to configure CEP
The user must have Domain Admin and Enterprise Admin rights

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential
```



Enter the user name and password credentials of the Windows user.

14. The script installs the Certificate Enrollment Policy Web Service, then prompts you to provide a friendly name for the service:

```
Installing AdcsEnrollmentPolicyWebService
Authentication : UserName
SSLThumbprint : DE45D685512D0F58A4CA7A60A485D8FD03723276
Install Complete
Please enter the 'Friendly Name' for the CEP Service :
```

Enter a friendly name for the Certificate Enrollment Policy Web Service. The friendly name must be unique for the domain. The friendly name will appear in some interfaces. For example: `WSTEP UserName CEP`.

15. To properly function with an existing Microsoft CA, the Certificate Enrollment Policy Web Service requires a UUID (Universally Unique Identifier). The script asks whether you want to generate a UUID or supply an external UUID.

```
All instances of the CEP Service must use the same UUID.
You must generate the UUID for the first instance, then supply the UUID for all
subsequent instances.
Generate or supply a UUID for the CEP Service?
Generate a UUID                1
Supply a UUID                  2

Enter Selection ( 1 | 2 ) :
```

- If you are installing the initial instance of the Certificate Enrollment Policy Web Service and you want the script to generate the UUID, enter 1. For example:

```
Generating Unique UUID
UUID 6e42b254-0302-4428-9bc5-c34d11c3b4b6
WARNING: Use the same UUID on all the CEP instances for Entrust WSTEP.
Selected UUID : 6e42b254-0302-4428-9bc5-c34d11c3b4b6
```

Record the generated UUID. All instances of the Certificate Enrollment Policy Web Service must use the same UUID for Entrust WSTEP enrollment.

- If you are installing subsequent instances of the Certificate Enrollment Policy Web Service and you want to supply the UUID generated in the first instance, enter 2. For example:


```
Enter a unique valid UUID : 1435d47b-a043-4b39-9420-0ff067344e4e
Selected UUID : 1435d47b-a043-4b39-9420-0ff067344e4e
```

When using Kerberos authentication, you can obtain the UUID of the installed Certificate Enrollment Policy Web Service by entering the following PowerShell command:

```
(Get-WebConfigurationProperty -pspath "MACHINE/WEBROOT/APPHOST/Default Web
Site/ADPolicyProvider_CEP_Kerberos" -filter "appSettings/add[@key='ID']"
-name "value").value
```

When using user name and password authentication, you can obtain the UUID of the installed Certificate Enrollment Policy Web Service by entering the following PowerShell command.

```
(Get-WebConfigurationProperty -pspath "MACHINE/WEBROOT/APPHOST/Default Web
Site/ADPolicyProvider_CEP_UsernamePassword" -filter "appSettings/
add[@key='ID']" -name "value").value
```

16. The script asks if you want to continue with the provided UUID:

```
Continue with above UUID? (y/n):
```

- To continue with the provided UUID, enter `y`.
- To go back and change the UUID, enter `n`.

17. The script asks if you want to restart Microsoft IIS:

```
Recommended: Restart IIS
Restart IIS now ? (y/n):
```

It is recommend that you restart Microsoft IIS to ensure the changes are applied.

- To have the script restart Microsoft IIS, enter `y`.
- To not restart Microsoft IIS, enter `n`.

It is recommended that you manually restart Microsoft IIS to ensure that the changes are applied.

Installing and configuring the CEP Web Service using the Windows graphical interface

To install and configure the Certificate Enrollment Policy Web Service, it is recommended that you run a PowerShell script as explained in [Installing and configuring the CEP Web Service using a PowerShell script](#). However, you can also use the Windows graphical interface, as explained in the following sections.

- [Installing the CEP Web Service using the Windows graphical interface](#)
- [Selecting the authentication mode of the CEP Web Service using the Windows graphical interface](#)
- [Assigning a friendly name to the CEP Web Service using the Windows graphical interface](#)
- [Assigning a unique Enrollment Policy Identifier](#)

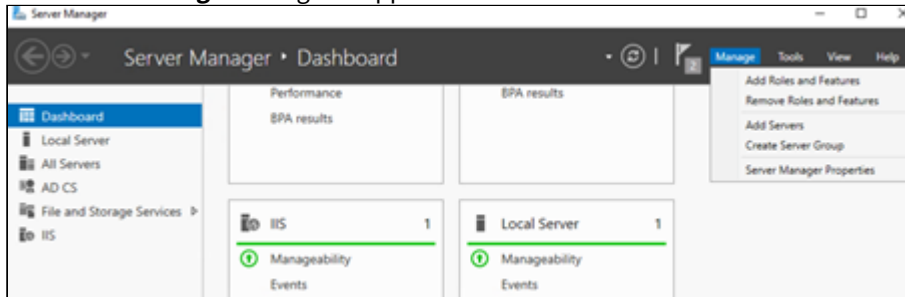
Installing the CEP Web Service using the Windows graphical interface

On a Windows server, install the Certificate Enrollment Policy Web Service.

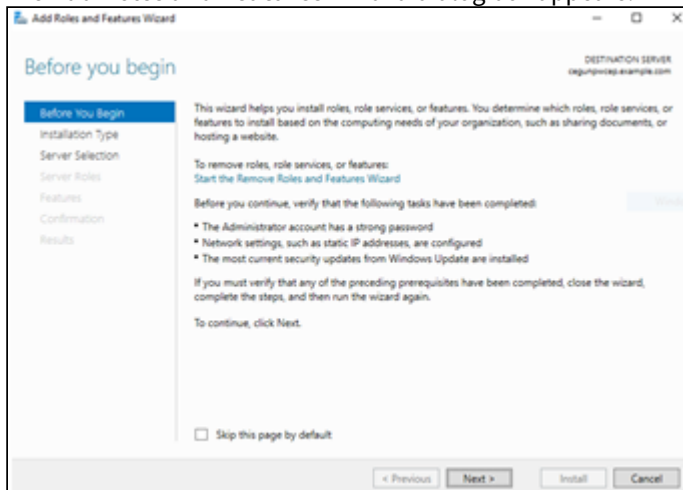
- i** The Windows server hosting the Certificate Enrollment Policy Web Service can be the Active Directory server or any other server in the domain. However, it is recommended that you install and configure the Certificate Enrollment Policy Web Service on a different server than Active Directory.

To install the CEP Web Service using the Windows graphical interface

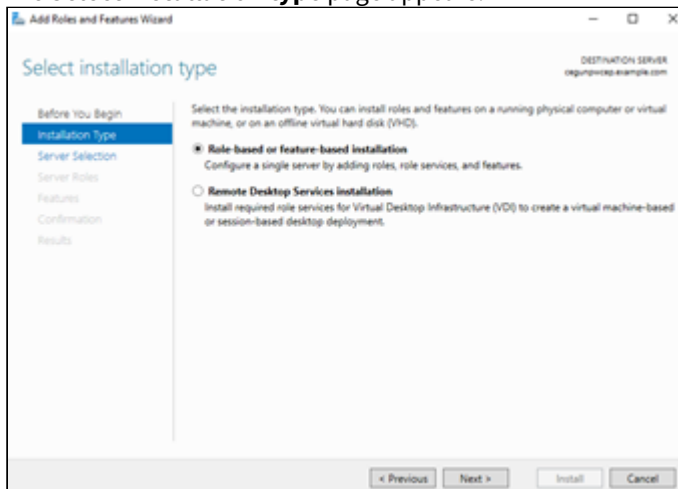
1. Log in to a Windows server hosting Microsoft IIS as a user with Domain Administrator and Enterprise Administrator permissions.
2. Open Server Manager. Select **Start > Server Manager**. The **Server Manager** dialog box appears.



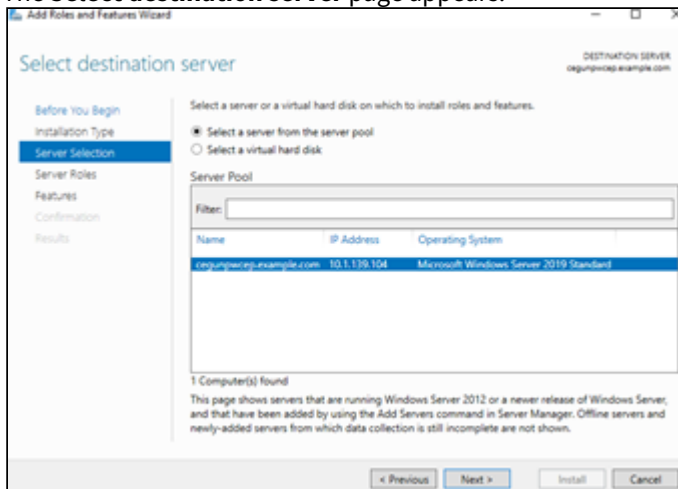
3. Select **Manage > Add Roles and Features**. The **Add Roles and Features Wizard** dialog box appears.



4. If the **Before you Begin** page appears, click **Next**.
The **Select installation type** page appears.

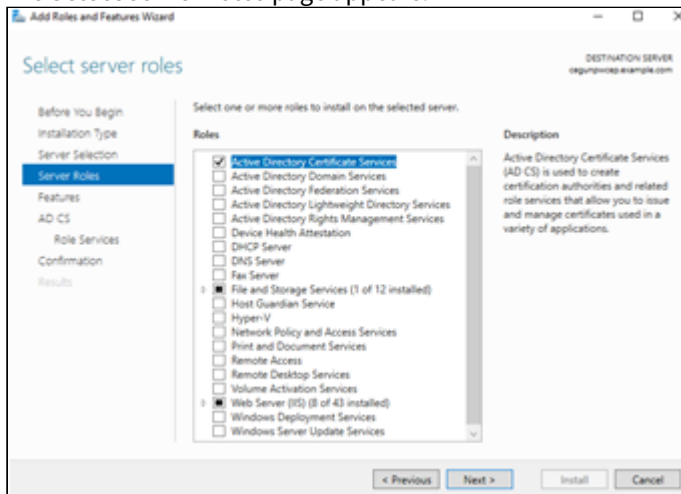


5. Select **Role-based or feature-based installation**.
6. Click **Next**.
The **Select destination server** page appears.

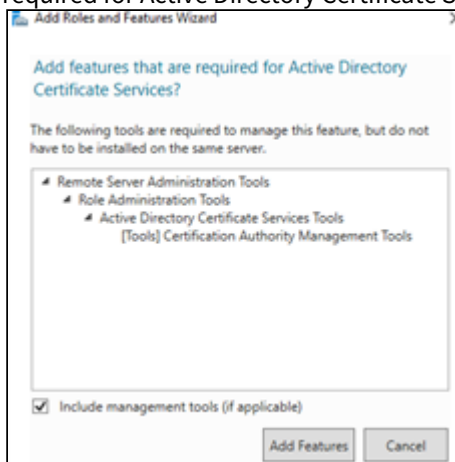


7. Click **Select a server from the pool**.
8. In the **Server Pool** list, select the server.

9. Click **Next**.
The **Select server roles** page appears.

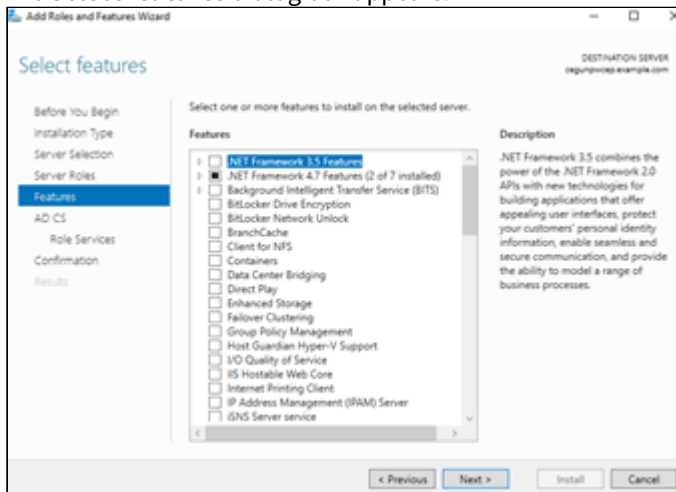


10. Select **Active Directory Certificate Services**.
Another **Add Roles and Features Wizard** dialog box may appear, informing you that some features are required for Active Directory Certificate Services.

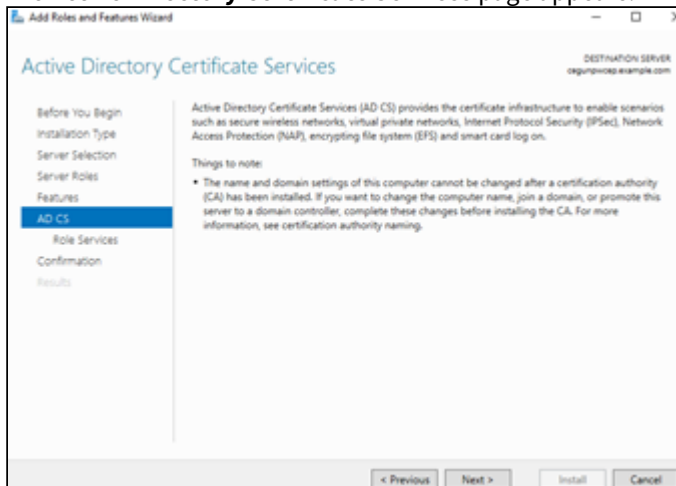


11. Click **Add Features** to add these required features and close the dialog box.

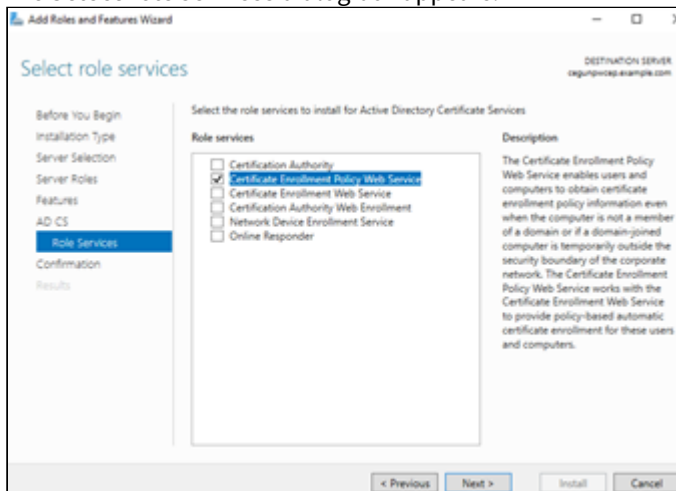
12. Click **Next**.
The **Select features** dialog box appears.



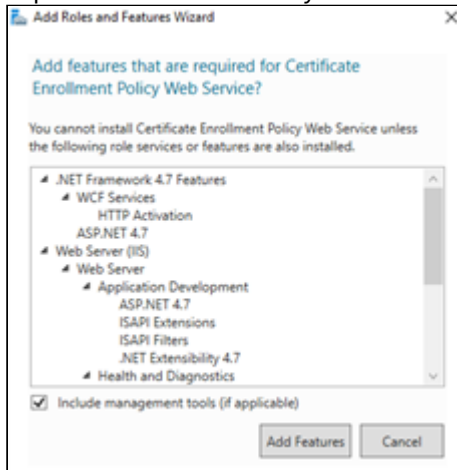
13. Click **Next**.
The **Active Directory Certificate Services** page appears.



14. Click **Next**.
The **Select role services** dialog box appears.

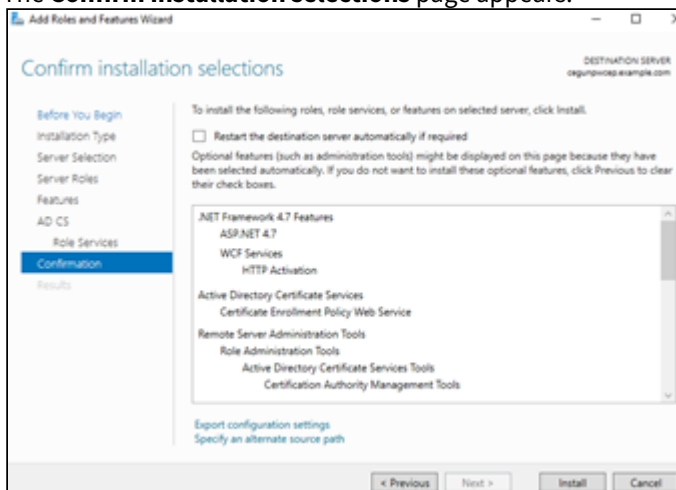


15. If you are on a different server than Active Directory, deselect **Certificate Authority**.
16. Select Certificate Enrollment Policy Web Service.
Another **Add Roles and Features Wizard** dialog box may appear, informing you that some features are required for Active Directory Certificate Services.



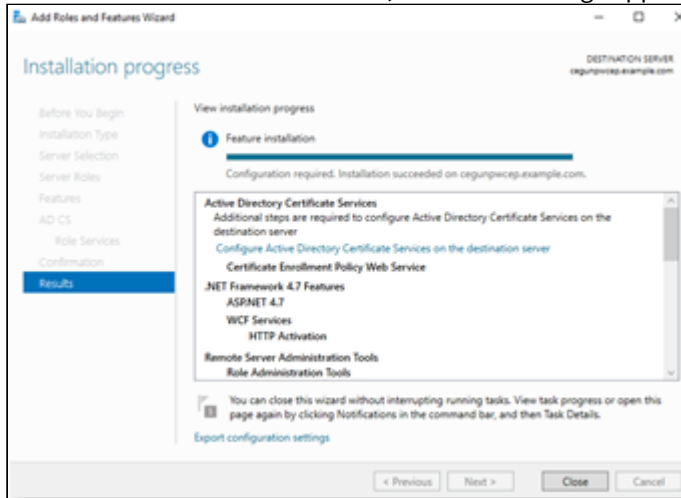
17. Click **Add Features** to add these required features and close the dialog box.
18. Click **Next**.

The **Confirm installation selections** page appears.




19. Click **Install**.
The **Installation Progress** page appears. A progress indicator displays the progress of the installation. After

the roles and features are installed, a success message appears.



20. Click **Close**.

Selecting the authentication mode of the CEP Web Service using the Windows graphical interface

 The Certificate Enrollment Policy Web Service requires a TLS certificate in Microsoft IIS. You will set the TLS certificate later, as explained in [Configuring the TLS certificate of the Windows endpoints](#).

After installing the Windows Certificate Enrollment Policy Web Service as documented in [Installing the CEP Web Service using the Windows graphical interface](#), you must select the authentication mode supported for the Certificate Enrollment Policy Web Service endpoints.

The Certificate Enrollment Gateway supports both Windows integrated authentication (Kerberos) and password authentication. Client certificate authentication is currently not supported.

If you support non-domain enrollment endpoints, you must configure the Certificate Enrollment Policy Web Service for username and password authentication.

When authenticating with username and password, the CEP Service supports the following username formats.

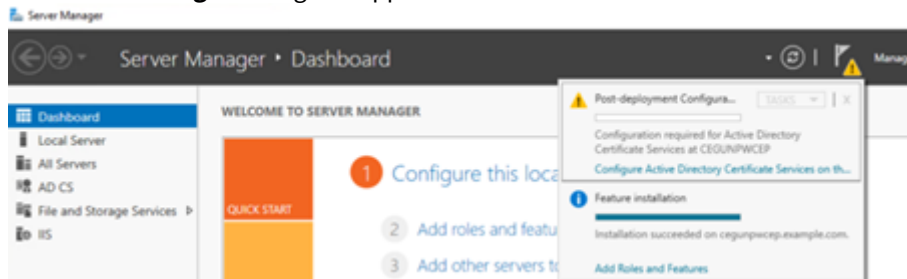
- username
- domain\username
- username@domain
- domainfqdn\username
- username@domainfqdn

WSTEP will use the supplied domain information to validate the user. If the domain is not supplied, WSTEP will attempt to use the domain information in the SOAP request. If the SOAP request domain information does not exist, WSTEP will use the configured domain from the Certificate Enrollment Gateway configuring.

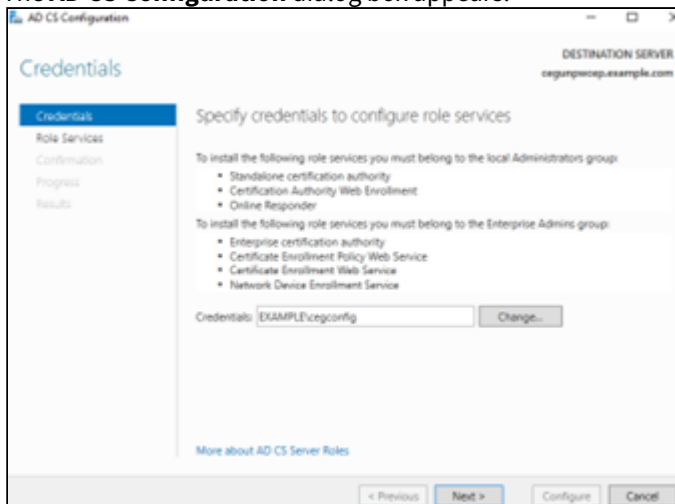
To select the authentication mode for the CEP Web Service using the Windows graphical interface

1. Log in to the server where you installed the Certificate Enrollment Policy Web Service.

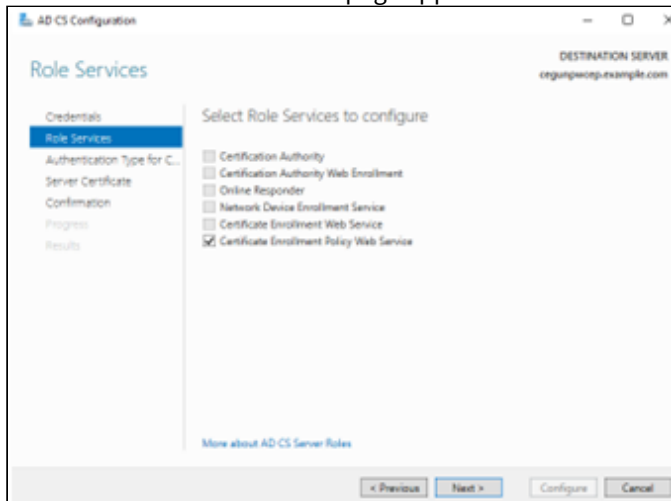
- Open Server Manager. Select **Start > Server Manager**. The **Server Manager** dialog box appears.



- Select **Notifications > Configure Active Directory Certificate Services** on the destination server. The **AD CS Configuration** dialog box appears.



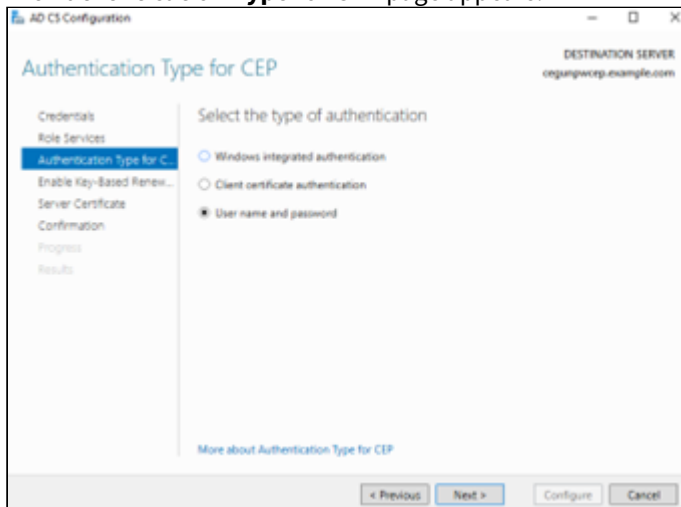
- Enter or select the administrator credentials you will use to configure role services.
- Click **Next**. The **Role Services** page appears.




- Select **Certificate Enrollment Policy Web Service**.

7. Click **Next**.

The **Authentication Type for CEP** page appears.



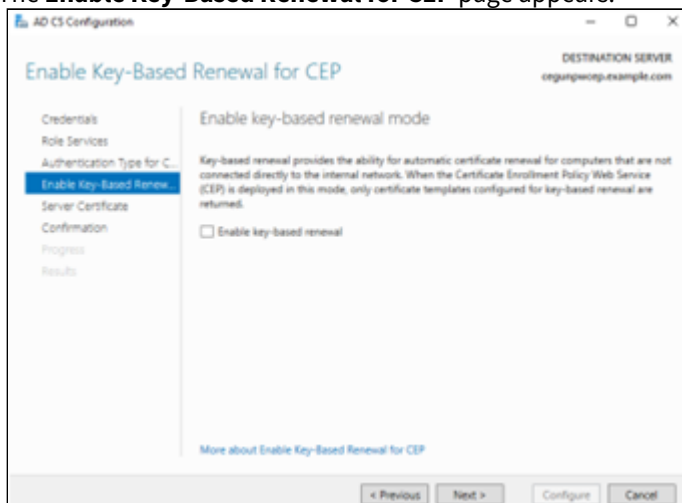
8. Select an authentication method supported by Certificate Enrollment Gateway.

 Client certificate authentication is currently not supported by Certificate Enrollment Gateway.

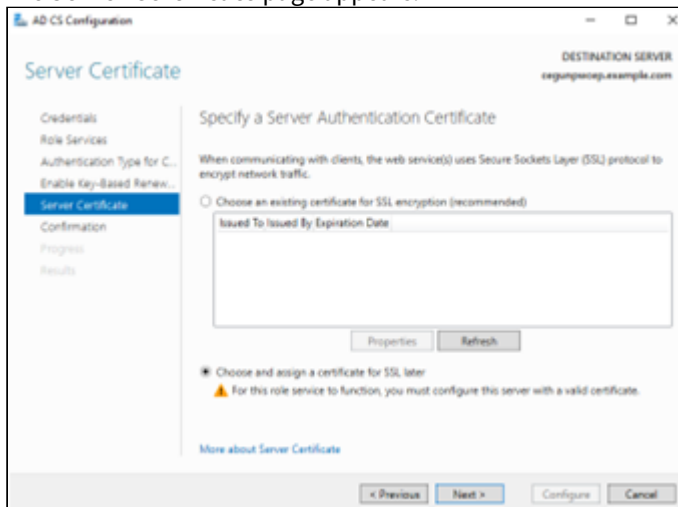
If you support non-domain enrollment endpoints, you must select **User name and password** as the authentication method.

9. Click **Next**.

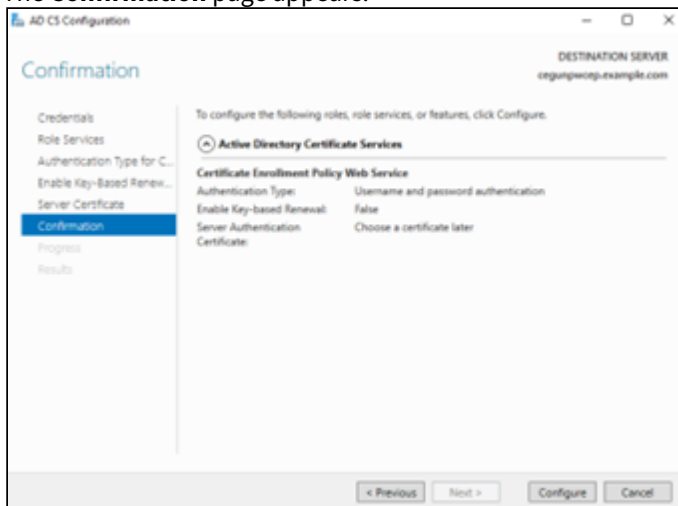
The **Enable Key-Based Renewal for CEP** page appears.



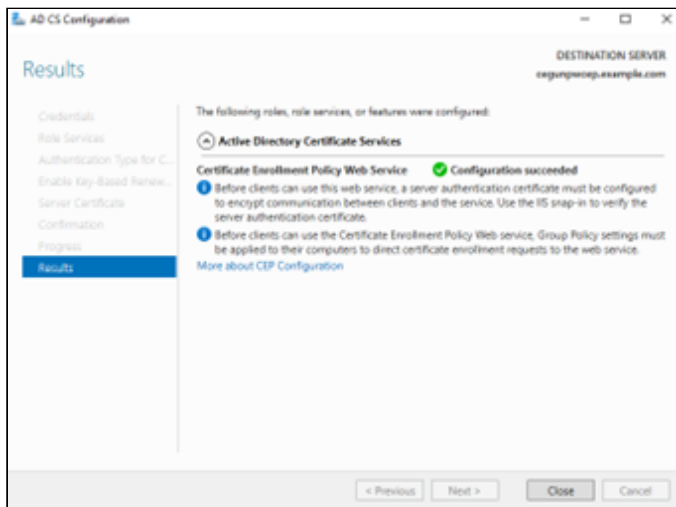
- Do not select any options. Click **Next**.
The **Server Certificate** page appears.



- Select **Choose and assign a certificate for SSL later**.
- Click **Next**.
The **Confirmation** page appears.



- Click **Configure**.
After the authentication mode is configured, the **Results** page appears.



14. Click **Close**.

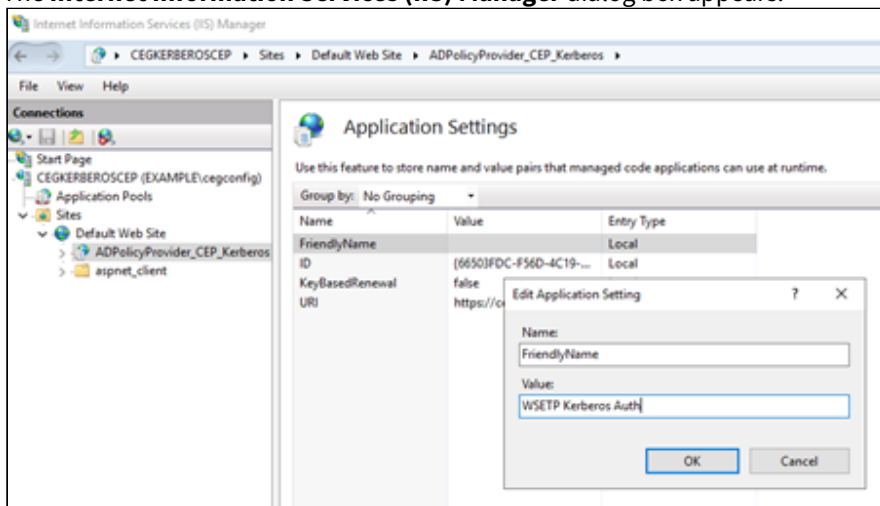
Assigning a friendly name to the CEP Web Service using the Windows graphical interface

Set a friendly name for each one of the configured Certificate Enrollment Policy Web Service instances. This friendly name will appear in some interfaces.

To set a friendly name for a CEP Web Service using the Windows graphical interface

1. Log in to the server where you installed the Certificate Enrollment Policy Web Service.
2. Open IIS Manager. Select **Start > Windows Administrative Tools > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.



3. In the **Connections** pane, expand **Sites > Default Web Site**.
4. Select the name of the Certificate Enrollment Policy Web Service application.
 - If you configured user name and password authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_UsernamePassword**.
 - If you configured Kerberos (Windows integrated) authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_Kerberos**.
5. In the content pane, double-click **Application Settings**.
An **Application Settings** pane appears.

6. In the **Application Settings** pane, double-click **FriendlyName**. An **Edit Application Setting** dialog box appears.
7. In the **Value** field, enter a unique and friendly name for the service.
8. Click **OK**.

Assigning a unique Enrollment Policy Identifier

For Entrust WSTEP to work alongside an existing Microsoft CA, you must change the Enrollment Policy ID to something unique. You can perform this operation using either PowerShell or the Windows graphical interface.

To assign a unique Enrollment Policy Identifier using PowerShell

1. Log in to the server hosting the Certificate Enrollment Policy Web Service.
2. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
3. Generate a unique identifier with the following command.

```
[guid]::NewGuid()
```

For example:

```
PS C:\> [guid]::NewGuid()
Guid
----
1c84d0f5-0eb4-4189-9e8d-a02b5d4079bd
```

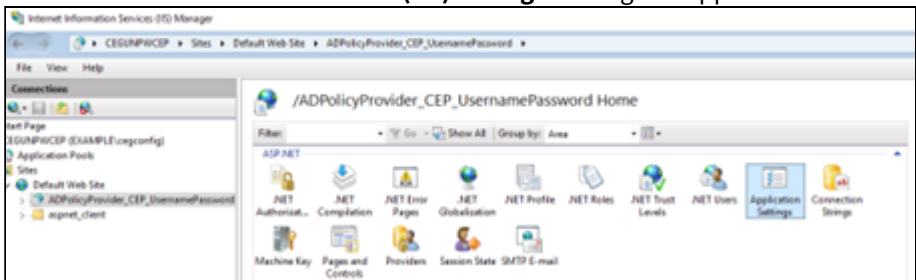
4. Set the new identifier. For example:

```
Set-WebConfigurationProperty -pspath "MACHINE/WEBROOT/APPHOST/Default Web Site/ADPolicyProvider_CEP_UsernamePassword" -filter "appSettings/add[@key='ID']" -name "value" -value "1c84d0f5-0eb4-4189-9e8d-a02b5d4079bd"
```

To assign a unique Enrollment Policy Identifier using the Windows graphical interface

1. Log in to the server hosting the Certificate Enrollment Policy Web Service.
2. Open IIS Manager. Select **Start > Windows Administrative Tools > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.

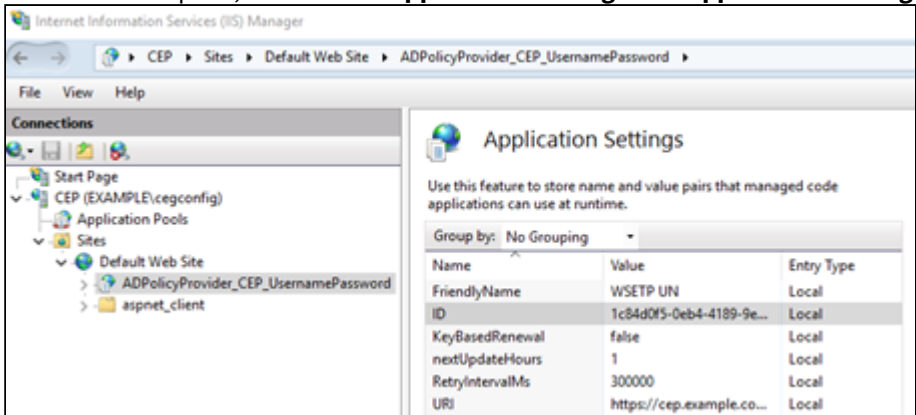


3. In the **Connections** pane, expand **Sites > Default Web Site**.
4. Select the name of the Certificate Enrollment Policy Web Service application.
 - If you configured user name and password authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_UsernamePassword**.

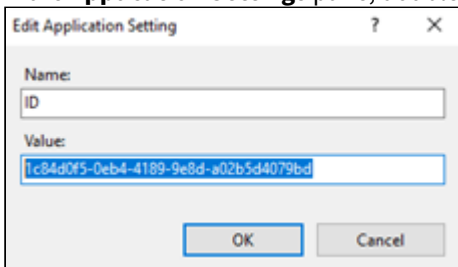
- If you configured Kerberos (Windows integrated) authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_Kerberos**.
5. If **Application Settings** is not available in the **Features** pane, run the following PowerShell command to install IIS Compatibility:

```
PS C:\>Add-WindowsFeature -Name Web-Mgmt-Compat
```

6. In the **Features** pane, double-click **Application Settings**. The **Application Settings** pane appears.



7. In the **Application Settings** pane, double-click on ID. The **Edit Application Setting** dialog box appears.



8. In the **Value** field of the **Edit Application Setting** dialog box, enter a unique identifier.
9. Click **OK**.
10. Restart IIS.
11. If you already added the Certificate Enrollment Policy to the group policy, remove the group policy's service and add it again to use the new identifier.

Adjusting the polling interval of the Certificate Enrollment Policy Web Service (Optional)

By default, enrollment endpoints will poll the server hosting the Certificate Enrollment Policy Web Service every 8 hours. If you want to increase or decrease how often enrollment endpoints poll the Certificate Enrollment Policy Web Service, complete the following procedure.

To adjust the polling interval of the Certificate Enrollment Policy Web Service using PowerShell

1. Log in to the server hosting the Certificate Enrollment Policy Web Service.
2. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
3. Set the **\$PSPATH** variable for the authentication type:
 - For user name and password authentication, enter the following command:

```
$PSPath = "MACHINE/WEBROOT/APPHOST/Default Web Site/  
ADPolicyProvider_CEP_UsernamePassword"
```

- For Kerberos authentication, enter the following command:

```
$PSPath = "MACHINE/WEBROOT/APPHOST/Default Web Site/  
ADPolicyProvider_CEP_Kerberos"
```

4. Enter the following command to add the configuration option **nextUpdateHours**:

```
Add-WebConfigurationProperty -pspath "$PSPath" -filter "appSettings" -name "."  
-value @{key="nextUpdateHours"}
```

The configuration option **nextUpdateHours** controls how often, in hours, enrollment endpoints will poll the Certificate Enrollment Policy Web Service.

5. Enter the following command to set the value of **nextUpdateHours** to 1 hour (the minimum interval supported).

```
Set-WebConfigurationProperty -pspath "$PSPath" -filter "appSettings/  
add[@key='nextUpdateHours']" -name "value" -value "1"
```

6. Enter the following command to add the configuration option **RetryIntervalMs**:

```
Add-WebConfigurationProperty -pspath "$PSPath" -filter "appSettings" -name "."  
-value @{key="RetryIntervalMs"}
```

The configuration option **RetryIntervalMs** controls how frequently, in milliseconds, Certificate Enrollment Policy Web Service refreshes templates and Certificate Authority (CA) information.

7. Enter the following command to set the value of **RetryIntervalMs** to 300,000 milliseconds (5 minutes):

```
Set-WebConfigurationProperty -pspath "$PSPath" -filter "appSettings/  
add[@key='RetryIntervalMs']" -name "value" -value "300000"
```

Avoid setting the value too small (values less than 1000) to avoid overhead on Microsoft IIS and Active Directory servers.

8. Enter the following command to restart Microsoft IIS and apply the changes:

```
iisreset
```

Creating an enrollment service in Active Directory using a PowerShell script

You must create an enrollment service for each CA that will issue certificates to the WSTEP endpoints. Entrust provides an `InstallEnrollmentService.ps1` PowerShell script that allows you to create, edit, and remove enrollment services in Active Directory.

To run the script, you must use a Windows user account with Domain Admin and Enterprise Admin permissions.


An enrollment service requires a DER-encoded CA certificate from the issuing CA. The script will prompt you to provide the CA certificate when creating an enrollment service.

To create an enrollment service with the `InstallEnrollmentService.ps1` script

1. Log in to a Windows server that is joined to the Active Directory domain. It is recommended that you run the PowerShell script on a different server than the domain controller.
2. From Entrust TrustedCare, download the PowerShell scripts for Certificate Enrollment Gateway.
3. Extract the PowerShell scripts to a directory on the server.
4. PowerShell scripts downloaded from the Internet may be blocked from running. To unblock a PowerShell script:
 - a. Right-click the PowerShell script > **Properties**.
A **Properties** dialog box appears.
 - b. Under the **General** tab, click **Unblock**.
5. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
6. Navigate to the directory where you extracted the PowerShell scripts.
7. Enter the following command to run the `InstallEnrollmentService.ps1` script:

```
.\InstallEnrollmentService.ps1
```

The script validates the pre-requisites and installs any missing Windows packages or features. For example:

 The PowerShell script was tested on specific versions of PowerShell. When validating the prerequisites, the PowerShell version may be listed as Unverified, an "Unverified" version of PowerShell indicates that the script was not tested on that version of PowerShell. You can still use the script on an "Unverified" version of PowerShell.

```
Validating pre-requisites:
Script-Mode: Windows
Script Version: 1.5.1.19
  - Member of Domain:           Verified
  - Domain Admins privileges:   Verified
  - Enterprise Admins privileges: Verified
  - Windows Version:           Verified (Microsoft Windows NT 10.0.17763.0)
  - PowerShell Version:        Verified (5.1.17763.2931)

-----
Validating ldifde is installed.

ldifde.exe is installed.

Validating Windows Feature RSAT-ADCS-Mgmt is installed
Installing RSAT-ADCS-Mgmt
```

8. The script prompts you to select a management option:

```
Entrust Enrollment Service PowerShell
```

```
Using this PowerShell script, Enrollments servers can be created, removed
and Edited.
```

```
Please select from the following options to continue :
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

Enter **N** to create a new enrollment service.

9. The script prompts you to provide the distinguished name (DN) of the configuration context.

```
Configuration Context DN
Format : DC=Example,DC=com
Configuration Context For Enrollment Service [Default: DC=example,DC=com]:
```

Enter the DN of the configuration context for Active Directory. The default value is the configuration context of the Active Directory forest.

10. The script prompts you to provide the host name for the enrollment service:

```
Hostname for Enrollment Service
Enrollment Service Hostname [Default: mmwin2019-2.example-ad.local]:
```

Enter the fully qualified domain name (FQDN) for the enrollment service. The default host name is the FQDN of the local server.

11. The script prompts you to provide a name for the enrollment service.

```
Enrollment Service Name is required to continue configuration
Enrollment Service Name [Default: Entrust WSTEP]:
```

Enter a name for the enrollment service. When entering a name:

- The name must be unique in the Active Directory forest.
- The name must start with an alphanumeric character.
- The name must contain only alphanumeric characters, spaces, hyphens, and underscores.

12. The script prompts you to provide the CA certificate from the issuing Certificate Authority (CA).

```
A der formatted certificate is required from the issuing CA.
Please use the full pathname and filename.
Example : C:\Users\admin\Downloads\cacert.der
Provide the full path and filename for the issuing CA certificate to proceed:
```

Enter the full path and file name of the certificate file. The CA certificate must be DER-encoded.

13. The script parses the file contents, displays the certificate settings, then asks if you want to use the certificate. For example:

```
Parsing Issuing CA cert for Subject DN.
Issuing CA Subject DN           : CN=Subordinate, OU=pki, O=Entrust
Issuing CA Certificate Effective Date : 5/25/2021 2:52:36 PM
Issuing CA Certificate Expiry Date   : 5/23/2031 2:52:36 PM
Use this CA Certificate? (y/n): y
```

- To use the selected CA certificate and continue, enter **y**.

- To go back and provide a different CA certificate, enter `n`.

14. The script prompts you to select the initial Certificate Template to be associated with the enrollment service:

```
Please select the initial Certificate Template to be associated
with the Enrollment Service.
The 'Template Name' cannot contain any spaces.
Certificate Template [Default: User]:
```

Enter the name of an existing Certificate Template to use as the initial Certificate Template for the enrollment service. The name cannot contain spaces.

15. The script asks if you want to continue with the selected Certificate Template.

```
Continue with User ? (y/n):
```

- To continue with the selected initial Certificate Template for the enrollment service, enter `y`.
- To go back and change the initial Certificate Template for the enrollment service, enter `n`.

16. The script displays the information you provided for the enrollment service and asks if you want to continue. For example:

```
Configuration Context DN      : DC=example,DC=com
Forest                       : example.com
Local hostname as DNS Hostname : cegaddc.example.com
Enrollment Service Name     : Entrust WSTEP
Issuing CA Der formatted Certificate : C:\EntrustPSScripts\ca.cer
Issuing CA                   : CN=Subordinate, OU=pki, O=Entrust
Certificate Template         : User
```

```
Continue with the above settings? (y/n):
```

- To continue with the settings and add the enrollment service to Active Directory, enter `y`.
- To go back and change all the settings for the enrollment service, enter `n`.

17. The script prompts you to provide the name of a new access group:

```
An Access group will be created for the Enrollment Service
```

```
By default, Active Directory provides the following domain groups for users,
computers, and domain controllers: Domain Users, Domain Computers, and Domain
Controllers. Creating a custom domain group for your Entrust WSTEP clients
ensures that only members of the custom domain group (your Windows-native
clients) can request certificates.
```

```
Configuring Access Group for example.com
```

```
Access Group Name [Default: Entrust WSTEP Access]:
```

By default, Active Directory provides the following domain groups for users, computers, and domain controllers: Domain Users, Domain Computers, and Domain Controllers. Creating a custom access group for

your Windows-native clients ensures that only members of the custom access group (your Windows-native clients) can request certificates through the enrollment service.

Enter a name for the new access group (by default, Entrust WSTEP Access).

18. The script asks if you want to continue adding the new access group:

```
Continue adding Access Group : Entrust WSTEP Access ? (y/n):
```

- To add the access group to the forest and continue, enter `y`.

The script waits 20 seconds to allow the group to propagate in Active Directory.

```
Pausing for 20 seconds to allow for the group to propagate
19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Resuming

Adding the Access group : Entrust WSTEP Access to the Enrollment Service :
Entrust WSTEP
Access Group example.com\Entrust WSTEP Access added to Enrollment
Service : Entrust WSTEP
```

- To go back and enter and enter a new access group name, enter `n`.
19. If the current domain is a top-level domain with subdomains, the script will ask if you want to set up an access group for enabling the enrollment service in one of the subdomains.
- To add the access group to the subdomain, enter `y`.
 - To go back and enter and enter a new access group name, enter `n`.
20. The script asks if you want to configure enrollment server URLs using the script:

```
Continue script to configure Enrollment Server URL(s) (y/n):
```

- To continue and configure enrollment server URLs using the script, enter `y`.
- The script displays a list of enrollment servers for the configured enrollment service. By default, the list should be NULL (no enrollment servers for the enrollment service). For example:

```
Enrollment Service Name : Entrust WSTEP

NULL set of Enrollment servers.
```

- To exit the script and configure the enrollment server URLs using the `certutil` utility later, enter `n`.
21. If you chose to configure enrollment server URLs using the script, the script asks if you want to configure an enrollment URL for user name and password authentication:

```
Configure UserName Enrollment URL ? (y/n):
```

- To configure an enrollment URL for user name and password authentication, enter `y`.
 - To skip configuring an enrollment URL for user name and password authentication, enter `n`.
22. If you chose to configure an enrollment URL for user name and password authentication:
- a. The script prompts you to enter an enrollment URL:

Please enter the Enrollment Server URL :

Enter the enrollment URL using the following format:

```
https://<CEG-server>:443/wstep/usertoken/services/<tenant-ID>/<CA-ID>
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example:

```
https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/example-ca1
```

- b. The script prompts you to specify the priority of the enrollment server:

The URI for the enrollment server which has the lowest priority number as defined in the enrollment policy. If two enrollment servers have the same priority then

a. The URI with the following authentication type is preferred in order:

Kerberos, Anonymous, Username/Password cached in the vault or Client Auth Certificate cached in the vault, Username/Password or Client Auth Certificate.

b. If all properties are equal then a URI is randomly selected.

Please enter the Priority of this Enrollment URL [Default : 1]:

If multiple enrollment servers are defined, then the priority determines which enrollment server is preferred. Enter the priority for the enrollment server.

- c. The script asks if the URL will be used for certificate renewal only:

Will this URL be used for Renewal ONLY ? (y/n):

- If the enrollment URL is for certificate renewal only, enter `y`.
- If the enrollment URL is for certificate enrollment and renewal, enter `n`.

23. If you chose to configure enrollment server URLs using the script, the script asks if you want to configure an enrollment URL for Kerberos (Windows integrated) authentication:

Configure Kerberos Enrollment URL ? (y/n):

- To configure an enrollment URL for Kerberos authentication, enter `y`.
- To skip configuring an enrollment URL for Kerberos authentication, enter `n`.

24. If you chose to configure an enrollment URL for Kerberos authentication:
- The script prompts you to enter an enrollment URL:

```
Please enter the Enrollment Server URL :
```

Enter the enrollment URL using the following format:

```
https://<CEG-server>:443/wstep/kerberos/services/<tenant-ID>/<CA-ID>
```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example:

```
https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/  
example-ca1
```

- The script prompts you to specify the priority of the enrollment server:

```
The URI for the enrollment server which has the lowest priority number as  
defined in the enrollment policy. If two enrollment servers have the same  
priority then
```

- The URI with the following authentication type is preferred in order:

```
Kerberos, Anonymous, Username/Password cached in the vault or  
Client Auth Certificate cached in the vault, Username/Password or  
Client Auth Certificate.
```

- If all properties are equal then a URI is randomly selected.

```
Please enter the Priority of this Enrollment URL [Default : 1]:
```

If multiple enrollment servers are defined, then the priority determines which enrollment server is preferred. Enter the priority for the enrollment server.

- The script asks if the URL will be used for certificate renewal only:

```
Will this URL be used for Renewal ONLY ? (y/n):
```

- If the enrollment URL is for certificate renewal only, enter `y`.
- If the enrollment URL is for certificate enrollment and renewal, enter `n`.

25. The main menu reappears:

```
Entrust Enrollment Service PowerShell
```

Using **this** PowerShell script, Enrollments servers can be created, removed and Edited.

Please select from the following options to **continue** :
[N] New Service [E] Edit Service [Q] Quit [?] Help (**default** is "N"):

To exit the script, enter **Q** .

Editing an enrollment service in Active Directory using a PowerShell script

Entrust provides an `InstallEnrollmentService.ps1` PowerShell script that allows you to create, edit, and remove enrollment services in Active Directory. When editing an enrollment service in Active Directory, you can:

- Update the enrollment URLs assigned to the enrollment service.
- Update security groups (access groups) assigned to the enrollment service.

To run the script, you must use a Windows user account with Domain Admin and Enterprise Admin permissions.

- [Updating the enrollment URLs for an enrollment service using a PowerShell script](#)
- [Updating the security groups for an enrollment service using a PowerShell script](#)

Updating the enrollment URLs for an enrollment service using a PowerShell script

Entrust provides an `InstallEnrollmentService.ps1` PowerShell script that allows you to create, edit, and remove enrollment services in Active Directory. When editing an enrollment service in Active Directory, you can update the enrollment URLs assigned to the enrollment service. When updating the enrollment URLs assigned to an enrollment service, you can:

- List all the enrollment URLs that are assigned to the enrollment service.
- Add an enrollment URL to the enrollment service.
- Remove one or all enrollment URLs from the enrollment service.


To run the script, you must use a Windows user account with Domain Admin and Enterprise Admin permissions.

To update the enrollment URLs for an enrollment service using the `InstallEnrollmentService.ps1` script

1. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
2. Navigate to the directory where you extracted the PowerShell scripts.
3. Enter the following command to run the `InstallEnrollmentService.ps1` script:

```
.\InstallEnrollmentService.ps1
```

The script validates the prerequisites and installs any missing Windows packages or features. For example:

-  The PowerShell script was tested on specific versions of PowerShell. When validating the prerequisites, the PowerShell version may be listed as Unverified, an "Unverified" version of PowerShell indicates that the script was not tested on that version of PowerShell. You can still use the script on an "Unverified" version of PowerShell.

```
Validating pre-requisites:  
Script-Mode: Windows
```

```
Script Version: 1.5.1.19
- Member of Domain:          Verified
- Domain Admins privileges:  Verified
- Enterprise Admins privileges: Verified
- Windows Version:          Verified (Microsoft Windows NT 10.0.17763.0)
- PowerShell Version:       Verified (5.1.17763.2931)
```

```
-----
Validating ldifde is installed.
```

```
ldifde.exe is installed.
```

```
Validating Windows Feature RSAT-ADCS-Mgmt is installed
Installing RSAT-ADCS-Mgmt
```

4. The script prompts you to select a management option:

```
Entrust Enrollment Service PowerShell
```

```
Using this PowerShell script, Enrollments servers can be created, removed
and Edited.
```

```
Please select from the following options to continue :
```

```
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

Enter **E** to edit an existing enrollment service.

5. If more than one enrollment service is defined in Active Directory, the script displays the list of enrollment services and asks you to select one of the enrollment services:

```
Select from the following List of defined Enrollment Services :
```

```
Index Enrollment Service Name
```

```
-----
1      CEG WSTEP
2      Entrust WSTEP
```

```
Please select the Index to select an Enrollment Service. 0 to quit.:
```

Enter the number associated with the enrollment service you want to edit. If only one enrollment service exists, that service is automatically selected by the script.

6. The script displays the currently-selected enrollment service, and prompts you to choose from a list of options:

```
Currently Selected Enrollment Service : Entrust WSTEP
```

```
Choose from the following Options:
```

```
[E] Edit [R] Remove [P] Previous [?] Help (default is "E"):
```

Enter **E** to edit the selected service.

7. The script prompts you to select an edit option:

```
Updating Enrollment Service : Entrust WSTEP

Menu to select between:
Updating the Enrollment Service URL(s)
Updating the Security Groups for the Enrollment Service.

Choose from the following Options:
[U] Update URL(s) [S] Update Security Group(s) [P] Previous [?] Help
(default is "U"):
```

Enter **U** to update the enrollment URLs.

8. The script asks you to select an update option:

```
Editing the URL(s) for Enrollment Service : Entrust WSTEP

Choose from the following Options:
[A] Add URL [D] Delete URL [L] List URL [P] Previous [?] Help (default is
"L"):
```

- To list all enrollment URLs for the enrollment service, enter **L**.
- To delete an enrollment URL from the enrollment service, enter **D**.
- To add an enrollment URL to the enrollment service, enter **A**.

9. If you chose to list the enrollment URLs for the enrollment service, the script displays information about each enrollment URL for the enrollment service. For example:

```
Enrollment Service Name : Entrust WSTEP

Priority      : 1
Auth Type    : UserName
Renewal Only : 0
URL          : https://cegserver1.example.com:443/wstep/usertoken/services/
tenant1/example-ca1
```

For each URL, the script displays the following information:

- **Priority** displays the priority of the enrollment server. If multiple enrollment servers are defined, then the priority determines which enrollment server is preferred.
- **Auth Type** displays the authentication type, either **UserName** for user name and password authentication, or **Kerberos** for Kerberos authentication (integrated Windows authentication).
- **Renewal Only** indicates if the enrollment URL is for certificate renewal only. 0 indicates that the enrollment URL is for both certificate enrollment and renewal. 1 indicates that the enrollment URL is for certificate renewal only.
- **URL** displays the enrollment URL.

10. If you chose to delete an enrollment URL from the enrollment service:

- a. The script displays a list of enrollment URLs and asks you to select which URL to remove:

```
Selected Enrollment Service Name : Entrust WSTEP
```

Retrieving URL(s) from AD.

```

Index   URL
-----
1       https://cegserver1.example.com:443/wstep/usertoken/services/
tenant1/example-ca1
Select the URL to remove. -1 to remove all, 0 to quit.:

```

- To remove a specific enrollment URL, enter the number associated with the URL in the list.
- To remove all enrollment URLs, enter -1.
- To go back without removing any enrollment URLs, enter 0.

b. The script asks you to confirm the removal of the URL or URLs. For example:

```

https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/
example-ca1 is slated to be removed
Continue with removal of URL (y/n)?:

```

- To confirm that you want remove the URL, enter `y`.
- To cancel the removal, enter `n`.

11. If you chose to add an enrollment URL to the enrollment service, the script asks if you want to configure an enrollment URL for user name and password authentication:

```

Configure UserName Enrollment URL ? (y/n):

```

- To configure an enrollment URL for user name and password authentication, enter `y`.
- To skip configuring an enrollment URL for user name and password authentication, enter `n`.

12. If you chose to add an enrollment URL for user name and password authentication:

a. The script prompts you to enter an enrollment URL:

```

Please enter the Enrollment Server URL :

```

Enter the enrollment URL using the following format:

```

https://<CEG-server>:443/wstep/usertoken/services/<tenant-ID>/<CA-ID>

```

Where:

- `<CEG-server>` is the hostname or IP address of the Certificate Enrollment Gateway server.
- `<tenant-ID>` is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- `<CA-ID>` is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example:

```

https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/
example-ca1

```


- b. The script prompts you to specify the priority of the enrollment server:

```
The URI for the enrollment server which has the lowest priority number as
defined in the enrollment policy. If two enrollment servers have the same
priority then
  a. The URI with the following authentication type is preferred in
  order:
      Kerberos, Anonymous, Username/Password cached in the vault or
      Client Auth Certificate cached in the vault, Username/Password or
      Client Auth Certificate.
  b. If all properties are equal then a URI is randomly selected.

Please enter the Priority of this Enrollment URL [Default : 1]:
```

If multiple enrollment servers are defined, then the priority determines which enrollment server is preferred. Enter the priority for the enrollment server.

- c. The script asks if the URL will be used for certificate renewal only:

```
Will this URL be used for Renewal ONLY ? (y/n):
```

- If the enrollment URL is for certificate renewal only, enter **y**.
- If the enrollment URL is for certificate enrollment and renewal, enter **n**.

- d. The script displays information about the enrollment URL and asks if you want to continue:

```
Enrollment Service : Entrust WSTEP
Authentication Type : UserName
Enrollment URL      : https://cegsrver1.example.com:443/wstep/usertoken/
services/tenant1/example-ca1
Priority              : 1
Modifiers            :
Continue with above settings? (y/n):
```

- To continue and add the enrollment URL, enter **y**.
- To go back and re-enter information about the enrollment URL, enter **n**.

13. If you chose to add an enrollment URL to the enrollment service, the script asks if you want to configure an enrollment URL for Kerberos (Windows integrated) authentication:

```
Configure Kerberos Enrollment URL ? (y/n):
```

- To configure an enrollment URL for Kerberos authentication, enter **y**.
- To skip configuring an enrollment URL for Kerberos authentication, enter **n**.

14. If you chose to add an enrollment URL for Kerberos authentication:

- a. The script prompts you to enter an enrollment URL:

```
Please enter the Enrollment Server URL :
```

Enter the enrollment URL using the following format:

```
https://<CEG-server>:443/wstep/kerberos/services/<tenant-ID>/<CA-ID>
```

Where:

- <CEG-server> is the hostname or IP address of the Certificate Enrollment Gateway server.
- <tenant-ID> is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- <CA-ID> is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example:

```
https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/  
example-ca1
```

- b. The script prompts you to specify the priority of the enrollment server:

The URI **for** the enrollment server which has the lowest priority number as defined in the enrollment policy. If two enrollment servers have the same priority then

- a. The URI with the following authentication type is preferred in order:

Kerberos, Anonymous, Username/Password cached in the vault or Client Auth Certificate cached in the vault, Username/Password or Client Auth Certificate.

- b. If all properties are equal then a URI is randomly selected.

Please enter the Priority of **this** Enrollment URL [Default : 1]:

If multiple enrollment servers are defined, then the priority determines which enrollment server is preferred. Enter the priority for the enrollment server.

- c. The script asks if the URL will be used for certificate renewal only:

```
Will this URL be used for Renewal ONLY ? (y/n):
```

- If the enrollment URL is for certificate renewal only, enter **y**.
- If the enrollment URL is for certificate enrollment and renewal, enter **n**.

- d. The script displays information about the enrollment URL and asks if you want to continue:

```
Enrollment Service : Entrust WSTEP  
Authentication Type : Kerberos  
Enrollment URL    : https://cegserver1.example.com:443/wstep/kerberos/  
services/tenant1/example-ca1  
Priority           : 1  
Modifiers         :  
Continue with above settings? (y/n):
```

- To continue and add the enrollment URL, enter **y**.
- To go back and re-enter information about the enrollment URL, enter **n**.

15. To exit the script after updating the enrollment URLs:
 - a. Keep entering **P** to return to a previous menu until you return to the main menu:

```
Entrust Enrollment Service PowerShell

Using this PowerShell script, Enrollments servers can be created, removed
and Edited.

Please select from the following options to continue :
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

- b. Enter **Q** to exit the script.

Updating the security groups for an enrollment service using a PowerShell script

Entrust provides an `InstallEnrollmentService.ps1` PowerShell script that allows you to create, edit, and remove enrollment services in Active Directory. When editing an enrollment service in Active Directory, you can update the security groups (access groups) assigned to the enrollment service. When updating the security groups assigned to an enrollment service, you can:

- List all the security groups that are assigned to the enrollment service.
- Add a security group to the enrollment service.
- Remove a security group from to the enrollment service.


To run the script, you must use a Windows user account with Domain Admin and Enterprise Admin permissions.

To update the security groups for an enrollment service using the `InstallEnrollmentService.ps1` script

1. Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
2. Navigate to the directory where you extracted the PowerShell scripts.
3. Enter the following command to run the `InstallEnrollmentService.ps1` script:

```
.\InstallEnrollmentService.ps1
```

The script validates the pre-requisites and installs any missing Windows packages or features. For example:

-  The PowerShell script was tested on specific versions of PowerShell. When validating the prerequisites, the PowerShell version may be listed as Unverified, an "Unverified" version of PowerShell indicates that the script was not tested on that version of PowerShell. You can still use the script on an "Unverified" version of PowerShell.

```
Validating pre-requisites:
Script-Mode: Windows
Script Version: 1.5.1.19
- Member of Domain:           Verified
- Domain Admins privileges:   Verified
- Enterprise Admins privileges: Verified
- Windows Version:           Verified (Microsoft Windows NT 10.0.17763.0)
- PowerShell Version:         Verified (5.1.17763.2931)
```

```
-----  
Validating ldifde is installed.  
  
ldifde.exe is installed.  
  
Validating Windows Feature RSAT-ADCS-Mgmt is installed  
Installing RSAT-ADCS-Mgmt
```

4. The script prompts you to select a management option:

```
Entrust Enrollment Service PowerShell  
  
Using this PowerShell script, Enrollments servers can be created, removed  
and Edited.  
  
Please select from the following options to continue :  
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

Enter **E** to edit an existing enrollment service.

5. If more than one enrollment service is defined in Active Directory, the script displays the list of enrollment services and asks you to select one of the enrollment services:

```
Select from the following List of defined Enrollment Services :  
  
Index Enrollment Service Name  
-----  
1      CEG WSTEP  
2      Entrust WSTEP  
  
Please select the Index to select an Enrollment Service. 0 to quit.:
```

Enter the number associated with the enrollment service you want to edit. If only one enrollment service exists, that service is automatically selected by the script.

6. The script displays the currently-selected enrollment service, and prompts you to choose from a list of options:

```
Currently Selected Enrollment Service : Entrust WSTEP  
  
Choose from the following Options:  
[E] Edit [R] Remove [P] Previous [?] Help (default is "E"):
```

Enter **E** to edit the selected service.

7. The script prompts you to select an edit option:

```
Updating Enrollment Service : Entrust WSTEP
```

```

Menu to select between:
Updating the Enrollment Service URL(s)
Updating the Security Groups for the Enrollment Service.

Choose from the following Options:
[U] Update URL(s) [S] Update Security Group(s) [P] Previous [?] Help
(default is "U"):
  
```

Enter **S** to update the security groups.

8. The script asks you to select an update option:

```

Updating Security Groups for Enrollment Service: Entrust WSTEP

Choose from the following Options:
[A] Add Security Group(s) [R] Remove Security Group(s) [L] List Security
Group(s) [P] Previous [?] Help
(default is "L"):
  
```

- To list all security groups for the enrollment service, enter **L**.
- To remove a security group from the enrollment service, enter **D**.
- To add a security group to the enrollment service, enter **A**.

9. If you chose to list the security groups for the enrollment service, the script displays a list of security groups assigned to the enrollment service. For example:

```

Index SecurityGroup                Permission
-----
 1 NT AUTHORITY\Authenticated Users    Allow
 2 NT AUTHORITY\SYSTEM                 Allow
 3 EXAMPLE-AD\Domain Admins           Allow
 4 EXAMPLE-AD\Entrust WSTEP Access     Allow
 5 EXAMPLE-AD\Enterprise Admins       Allow
 6 EXAMPLE-AD\Domain Admins           Allow
  
```

10. If you chose to remove a security group from the enrollment service:
 - a. The script displays a list of security groups assigned to the enrollment service and asks you to select which security group to remove:

```

Index SecurityGroup                Permission
-----
 1 NT AUTHORITY\Authenticated Users    Allow
 2 NT AUTHORITY\SYSTEM                 Allow
 3 EXAMPLE-AD\Domain Admins           Allow
 4 EXAMPLE-AD\Entrust WSTEP Access     Allow
 5 EXAMPLE-AD\Enterprise Admins       Allow
 6 EXAMPLE-AD\Domain Admins           Allow

Select the index of the Security Group to be removed. 0 to quit. :
  
```

Enter the index number associated with the security group you want to remove from the enrollment service.

- b. The script asks you to confirm the removal of security group. For example:

```
Removing Security Group : EXAMPLE-AD\Domain Admins
Are you sure you want to remove the Security Group (y/n)?:
```

- To confirm that you want remove the security group from the enrollment service, enter `y`.
- To cancel the removal, enter `n`.

11. If you chose to add a security group to the enrollment service:

- a. The script displays information about how to enter the name of the security group, then asks you to enter the security group you want to add to the enrollment service:

```
The Security group must exist before this script can add to the Enrollment
Service.
```

```
Enrollment service to be modified : Entrust WSTEP
```

```
For a Security Group, the following options are supported :
```

- 1) domain\group
- 2) domain.com\group
- 3) group@domain
- 4) group@domain.com

```
Security Groups must be:
```

- 1) GroupCategory must be Security
- 2) objectClass must be type group

```
Enter security Group. 0 to quit.:
```

Enter the name of the security group you want to add to the enrollment service, or enter 0 to go back without adding a security group.

The security group must already exist in Active Directory. In Active Directory, the GroupCategory of the group must be **Security**, and the objectClass of the group must be **group**.

You must enter the name of the group using one of the following formats:

- <domain>\<group>
- <fqdn>\<group>
- <group>@<domain>
- <group>@<fqdn>

Where:

- `<domain>` is the Active Directory domain where the security group is located.
- `<fqdn>` is the fully-qualified domain name of the Active Directory forest where the security group is located.
- `<group>` is the name of the security group.

Examples:

```
EXAMPLE\Example Group
EXAMPLE.COM\Example Group
```

```
Example Group@EXAMPLE
Example Group@EXAMPLE.COM
```

- b. The script adds the security group to the enrollment service. For example:

```
Adding the Access group : EXAMPLE-AD\Domain Users to the Enrollment
Service : Entrust WSTEP
Access Group EXAMPLE-AD\Domain Users added to Enrollment Service : Entrust
WSTEP
```

12. To exit the script after updating the security groups assigned to the enrollment service:

- a. Keep entering **P** to return to a previous menu until you return to the main menu:

```
Entrust Enrollment Service PowerShell

Using this PowerShell script, Enrollments servers can be created, removed
and Edited.

Please select from the following options to continue :
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

- b. Enter **Q** to exit the script.

Editing an enrollment service in Active Directory using Windows tools

This section describes how to edit an enrollment service in Active Directory using native Windows tools.

- [Building the Enrollment URL](#)
- [Adding the enrollment URL to the enrollment service using the certutil utility](#)
- [Changing the enrollment URL of the enrollment service using ADSI Edit](#)

Building the Enrollment URL

To build the Enrollment Service URL, use the following syntax:

```
https://<CEG-server>:443/wstep/<auth>/services/<tenant-ID>/<CA-ID>
```

Where:

- **<CEG-server>** is the hostname or IP address of the Certificate Enrollment Gateway server.
- **<auth>** is the authentication method, either **usertoken** for user name and password authentication or **kerberos** for Kerberos (Windows integrated) authentication.
- **<tenant-ID>** is the unique identifier of a tenant defined in Certificate Enrollment Gateway. The value is case-sensitive.
- **<CA-ID>** is the CA ID of the Certificate Authority (CA) defined in CA Gateway that will issue certificates to the Windows endpoint.

For example, when authenticating with a user name and password:

```
https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/example-ca1
```

For example, when authenticating with Kerberos:

```
https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/example-ca1
```

Adding the enrollment URL to the enrollment service using the `certutil` utility

On the Active Directory server, open a Command Prompt window and run the following command to add the Enrollment Service URL with the `certutil` utility.

```
certutil -config "<name>" -enrollmentserverURL <url> <auth> [<priority>]
```

Where:

- `<name>` is the name of the enrollment service.
- `<url>` is the URL described in [Building the Enrollment URL](#).
- `<auth>` is the identifier of the authentication mode: `kerberos` for Kerberos authentication, or `usertoken` for user name and password authentication.
- `<priority>` is the server priority. If you omit this parameter, the value defaults to 1.

For example, when authenticating with a user name and password:

```
certutil -config "CEGMSCA" -enrollmentserverURL https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/example-ca1 username
```

For example, when authenticating with Kerberos:

```
certutil -config "CEGMSCA" -enrollmentserverURL https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/example-ca1 kerberos
```

To check the added URL, run `certutil` without arguments. For example:

```
PS C:\Windows\system32> certutil
Entry 0:
Name: "CEGMSCA"
Organizational Unit: ""
Organization: ""
Locality: ""
State: ""
Country/region: ""
Config: "cegmsca.example.com\CEGMSCA"
Exchange Certificate: ""
Signature Certificate: ""
Description: ""
Server: "cegmsca.example.com"
Authority: "CEGMSCA"
```



```

Sanitized Name:          "CEGMSCA"
Short Name:              "CEGMSCA"
Sanitized Short Name:    "CEGMSCA"
Flags:                   "1"
Web Enrollment Servers:
1
4
0
https://cegserver1.example.com:443/wstep/usertoken/services/tenant1/example-ca1
0

1
2
0
https://cegserver1.example.com:443/wstep/kerberos/services/tenant1/example-ca1
0
CertUtil: -dump command completed successfully.

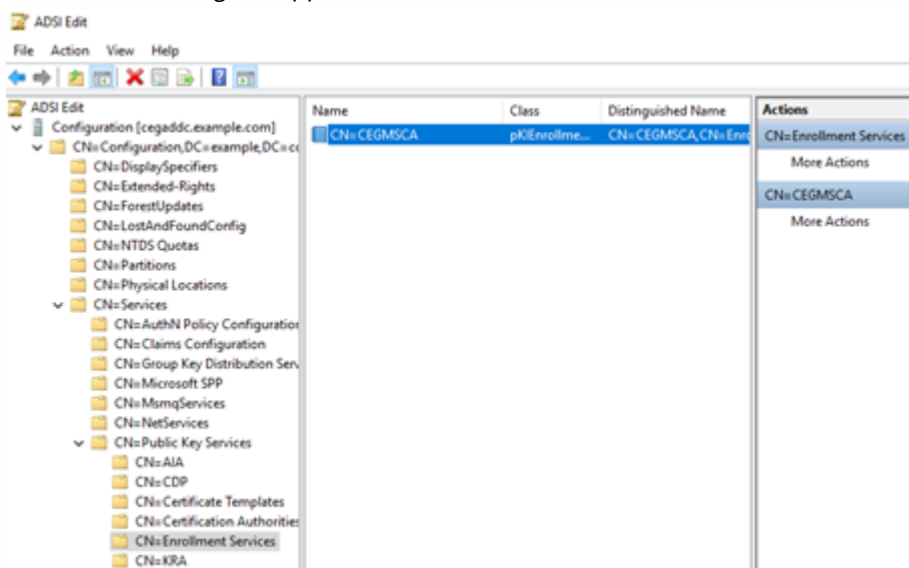
```

Changing the enrollment URL of the enrollment service using ADSI Edit

To add the enrollment service URL to Active Directory using ADSI Edit, complete the following procedure.

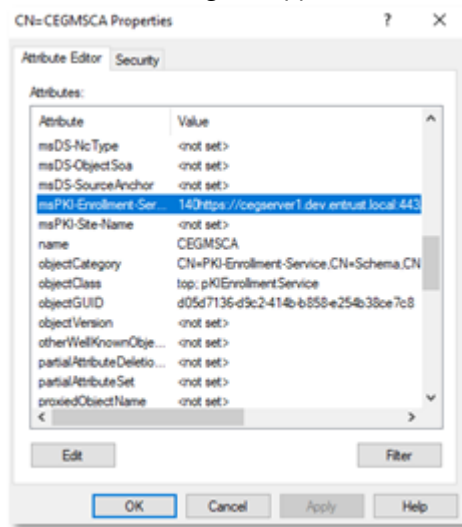
To add the enrollment service URL to Active Directory using ADSI Edit

1. Log in to the server hosting Active Directory.
2. Open ADSI Edit. Select **Start > Windows Administrative Tools > ADSI Edit**. The **ADSI Edit** dialog box appears.




3. In the tree view, expand **ADSI Edit > Configuration > CN=Configuration,<suffix> > CN=Services > CN=Public Key Services > CN=Enrollment Services**.

- Double-click the Active Directory CA enrollment service. A **Properties** dialog box appears.



- Click the **Attribute Editor** tab.
- Under **Attributes**, select **msPKI-Enrollment-Servers**. The URL in this field is preceded by three integers, where:
 - The first integer is the Priority of the service URL.
 - The second integer is the authentication method: 2 for Kerberos authentication, 4 for username and password authentication.
 - The third integer is 0 for certificate enrollment and renewal or 1 for certificate renewal only.
- Click **Edit**.
- Replace the current URL with the URL you built earlier in [Building the Enrollment URL](#).

 Do not overwrite the preceding integers when replacing the URL. The integers are required.

- Click **OK**.

Removing an enrollment service from Active Directory using a PowerShell script

Entrust provides an `InstallEnrollmentService.ps1` PowerShell script that allows you to create, edit, and remove enrollment services in Active Directory. The script also allows you to remove enrollment services from Active Directory.

You should remove an enrollment service only when you will no longer use that enrollment service with Certificate Enrollment Gateway.

To run the script, you must use a Windows user account with Domain Admin and Enterprise Admin permissions.

To update the enrollment URLs for an enrollment service using the `InstallEnrollmentService.ps1` script

- Open an elevated PowerShell window. Select **Start > Windows PowerShell**, then right-click **Windows PowerShell > Run as administrator**.
- Navigate to the directory where you extracted the PowerShell scripts.
- Enter the following command to run the `InstallEnrollmentService.ps1` script:

```
.\InstallEnrollmentService.ps1
```

The script validates the pre-requisites and installs any missing Windows packages or features. For example:

i The PowerShell script was tested on specific versions of PowerShell. When validating the prerequisites, the PowerShell version may be listed as Unverified, an "Unverified" version of PowerShell indicates that the script was not tested on that version of PowerShell. You can still use the script on an "Unverified" version of PowerShell.

```
Validating pre-requisites:
Script-Mode: Windows
Script Version: 1.5.1.19
  - Member of Domain:          Verified
  - Domain Admins privileges:  Verified
  - Enterprise Admins privileges: Verified
  - Windows Version:          Verified (Microsoft Windows NT 10.0.17763.0)
  - PowerShell Version:       Verified (5.1.17763.2931)
```

Validating ldifde is installed.

ldifde.exe is installed.

Validating Windows Feature RSAT-ADCS-Mgmt is installed
Installing RSAT-ADCS-Mgmt

4. The script prompts you to select a management option:

```
Entrust Enrollment Service PowerShell

Using this PowerShell script, Enrollments servers can be created, removed
and Edited.

Please select from the following options to continue :
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

Enter **E** to edit an existing enrollment service.

5. If more than one enrollment service is defined in Active Directory, the script displays the list of enrollment services and asks you to select one of the enrollment services:

```
Select from the following List of defined Enrollment Services :

Index Enrollment Service Name
-----
1      CEG WSTEP
2      Entrust WSTEP

Please select the Index to select an Enrollment Service. 0 to quit.:
```

Enter the number associated with the enrollment service you want to edit. If only one enrollment service exists, that service is automatically selected by the script.

- The script displays the currently-selected enrollment service, and prompts you to choose from a list of options:

```
Currently Selected Enrollment Service : Entrust WSTEP

Choose from the following Options:
[E] Edit [R] Remove [P] Previous [?] Help (default is "E"):
```

Enter **R** to remove the selected enrollment service.

- The script asks you to confirm the operation:

```
Removing Enrollment Service : CEG WSTEP

Are you sure you want to delete this Enrollment Service(y/n)?:
```

- To confirm that you want to delete the enrollment service, enter **y**. The script deletes the enrollment service from Active Directory.
- To cancel the operation and return to the previous menu, enter **n**.

- To exit the script after removing the enrollment service:

- Keep entering **P** to return to a previous menu until you return to the main menu:

```
Entrust Enrollment Service PowerShell

Using this PowerShell script, Enrollments servers can be created, removed
and Edited.

Please select from the following options to continue :
[N] New Service [E] Edit Service [Q] Quit [?] Help (default is "N"):
```

- Enter **Q** to exit the script.

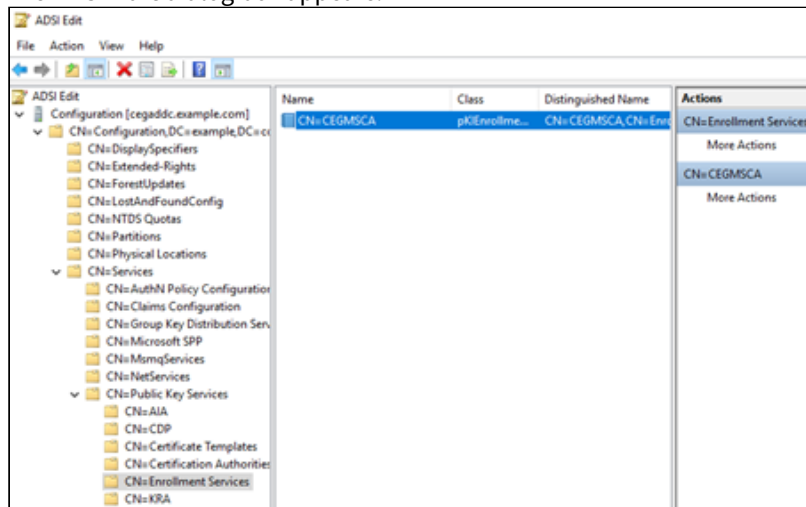
Adding certificate templates to the enrollment service

Previously, you created certificate templates for the Entrust WSTEP Service (see [Creating Windows certificate templates for the Entrust WSTEP Service](#)). You must add all the certificate templates you created to the enrollment service.

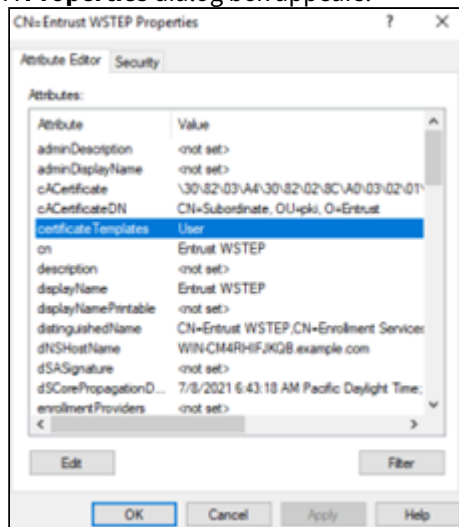
To add certificate templates to the enrollment service

- Log in to the server hosting Active Directory.

- Open ADSI Edit. Select **Start > Windows Administrative Tools > ADSI Edit**. The **ADSI Edit** dialog box appears.



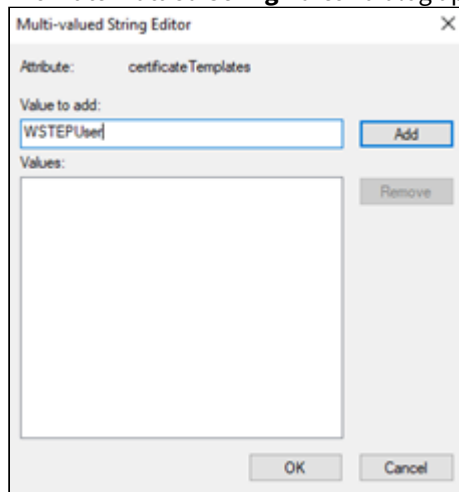
- In the tree view, expand **ADSI Edit > Configuration > CN=Configuration,<suffix> > CN=Services > CN=Public Key Services > CN=Enrollment Services**.
- Double-click the Active Directory CA enrollment service. A **Properties** dialog box appears.



- Click the **Attribute Editor** tab.
- Under **Attributes**, select **certificateTemplates**.

7. Click **Edit**.

The **Multi-valued String Editor** dialog appears.



8. In the **Value to add** field, paste the name of the certificate template. You must add at least the template with the WSTEPUser name.

i The template name is the value of the template's common name (CN) value. You can get all certificate template names at **ADSI Edit > Configuration > CN=Configuration,<suffix> > CN=Services > CN=Public Key Services > CN=Certificate Templates**.

9. Click **Add**.

10. If required, add additional templates to the list.

11. Click **OK**.

Configuring enrollment endpoints

You must configure all Windows domain and non-domain endpoints for which the Certificate Enrollment Gateway will issue certificates.

- [Configuring Windows Domain Endpoints](#)
- [Configuring non-domain endpoints](#)

Configuring Windows Domain Endpoints

You must configure all Windows domain endpoints—domain controllers and computers in a Windows domain—that will be issued certificates by Certificate Enrollment Gateway.

i For WSTEP enrollment, some machines also need TLS certificates. You will be configuring the TLS certificates later, starting in [Configuring the TLS certificate of the Windows endpoints](#).

- [Obtaining the URL of the Certificate Enrollment Policy Web Service](#)
- [Importing the CA certificate into Windows domain endpoints](#)
- [Configuring the Certificate Enrollment Policy Web Service for Windows domain endpoints](#)
- [Configuring the Certificate Enrollment Policy Web Service for Windows users](#)
- [Enabling certificate auto-enrollment for computers and domain controllers](#)
- [Enabling certificate auto-enrollment for users](#)

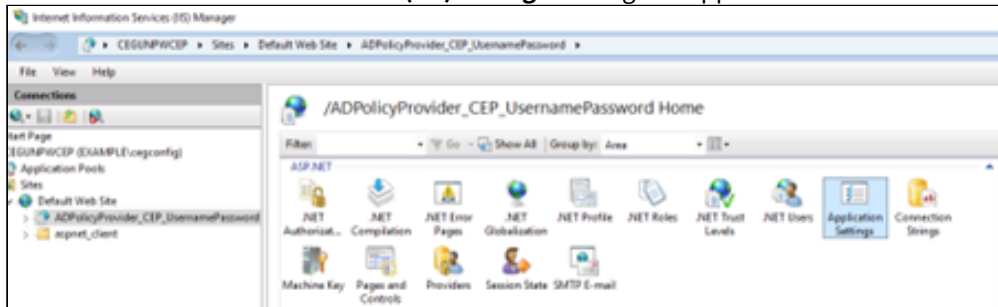
Obtaining the URL of the Certificate Enrollment Policy Web Service

To work with Certificate Enrollment Gateway, Windows domain endpoints need the Certificate Enrollment Policy Web Service URL. To obtain the URL, complete the following procedure.

To obtain the URL of the Certificate Enrollment Policy Web Service

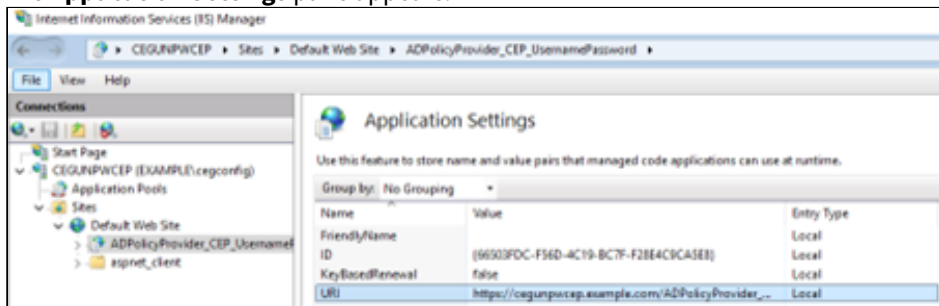
1. Log in to the server hosting the Certificate Enrollment Policy Web Service.
2. Open IIS Manager. Select **Start > Windows Administrative Tools > Internet Information Services (IIS) Manager**.

The **Internet Information Services (IIS) Manager** dialog box appears.



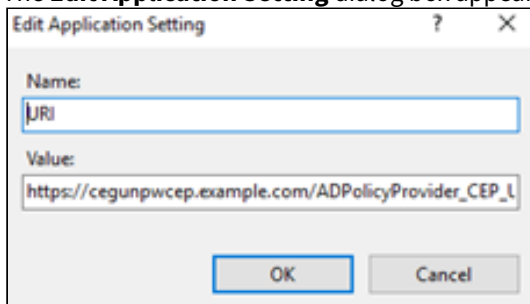
3. In the **Connections** pane, expand **Sites > Default Web site**.
4. Select the name of the Certificate Enrollment Policy Web Service application.
 - If you configured user name and password authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_UsernamePassword**.
 - If you configured Kerberos (Windows integrated) authentication for the Certificate Enrollment Policy Web Service, the identifier is **ADPolicyProvider_CEP_Kerberos**.
5. In the Features pane, double-click **Application Settings**.

The **Application Settings** pane appears.



6. In the **Application Settings** pane, right-click **URI > Edit**.

The **Edit Application Setting** dialog box appears.



7. Copy the URL from the **Value** field.

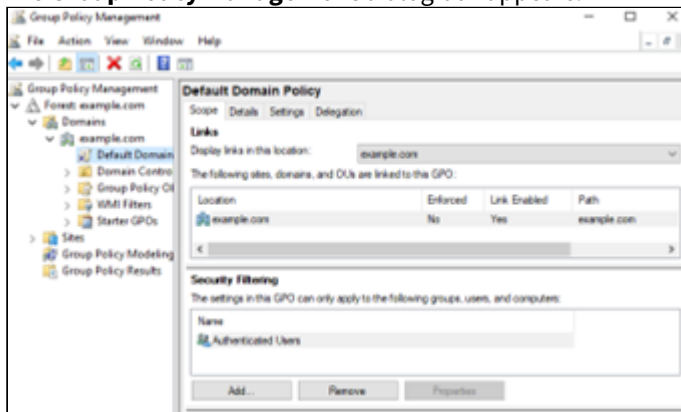
Importing the CA certificate into Windows domain endpoints

In each Windows domain enrollment endpoint, import the root certificate of the CA that will issue certificates for the enrollment service.

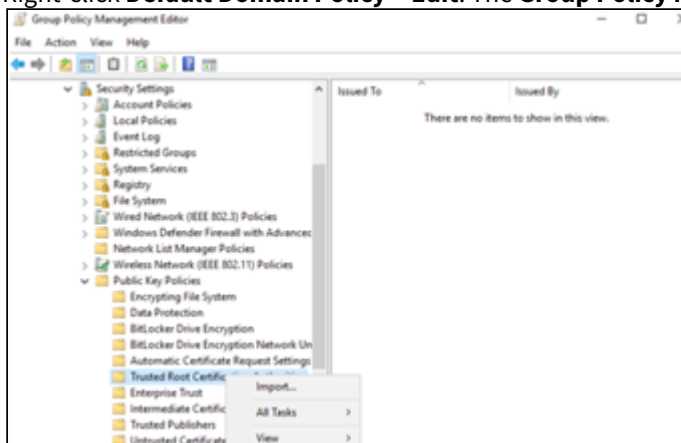
To import the CA certificate

1. Log in to the server hosting Active Directory.
2. Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**.

The **Group Policy Management** dialog box appears.

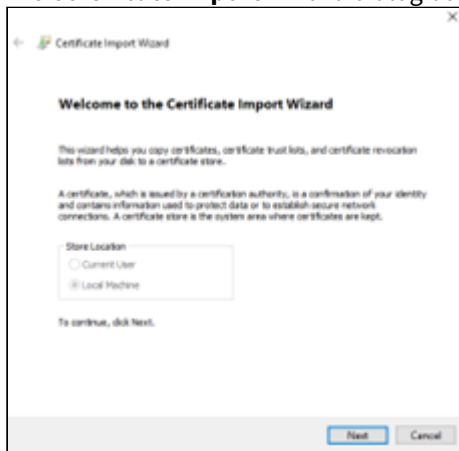


3. In the tree view, expand the Domain Controller you will modify.
4. Right-click **Default Domain Policy > Edit**. The **Group Policy Management Editor** dialog box appears.

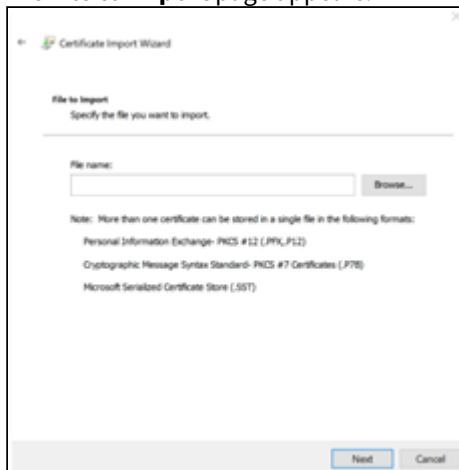


5. In the tree view, expand **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies**.

- Right-click **Trusted Root Certification Authorities** and select **Import**. The **Certificate Import Wizard** dialog box appears.



- Click **Next**. The **File to Import** page appears.



- Click **Browse** and select the root certificate of the CA that will issue certificates for the enrollment service.
- Click **Next**. The **Certificate Store** page appears.
- The **Certificate Store** field is automatically set to **Trusted Root Certification Authorities**. Click **Next**. The **Completing the Certificate Import Wizard** page appears.
- Click **Finish**.

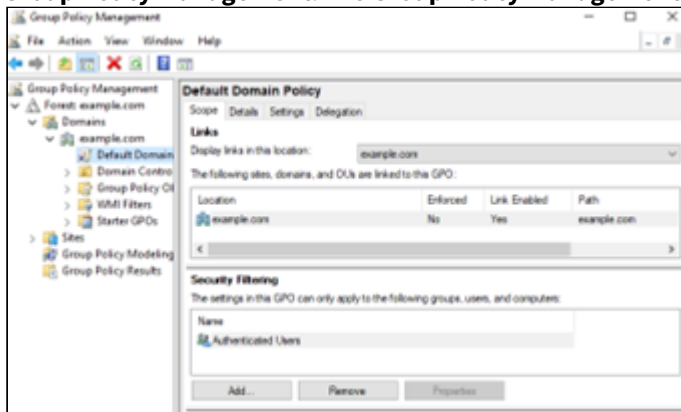
Configuring the Certificate Enrollment Policy Web Service for Windows domain endpoints

To work with Certificate Enrollment Gateway, Windows domain endpoints need the Certificate Enrollment Policy Web Service URL. Complete the following procedure to add the Certificate Enrollment Policy Web Service URL for Windows domain endpoints.

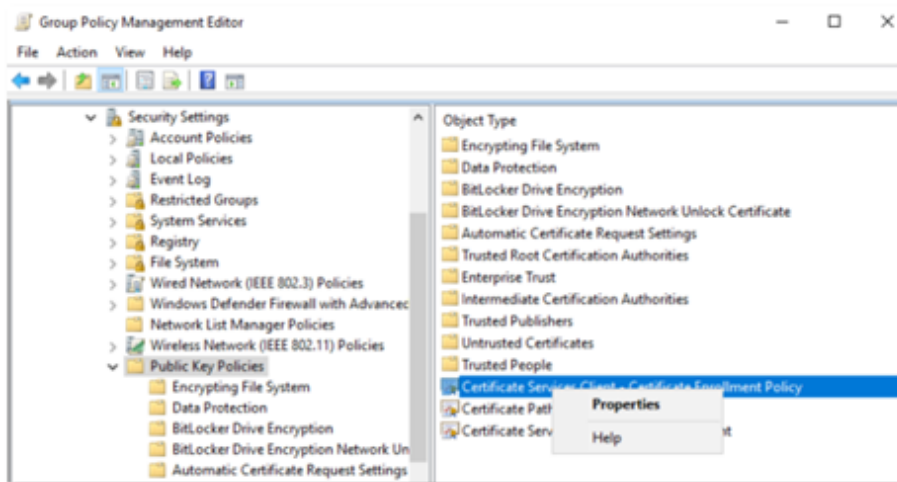
To configure the Certificate Enrollment Policy Web Service for Windows domain endpoints

- Log in to the server hosting Active Directory.

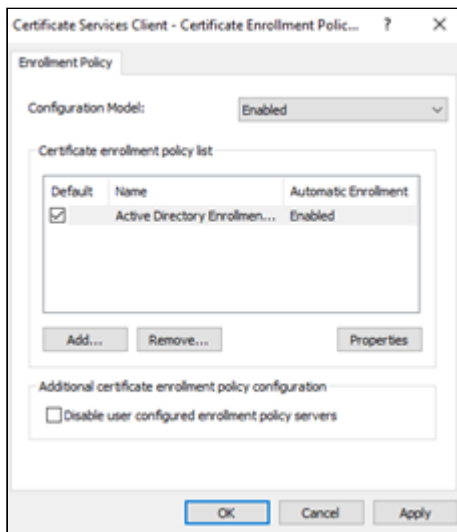
- Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**. The **Group Policy Management** dialog box appears.



- In the tree view, expand the Domain Controller you will modify.
- Right-click **Default Domain Policy > Edit**. The **Group Policy Management Editor** dialog box appears.

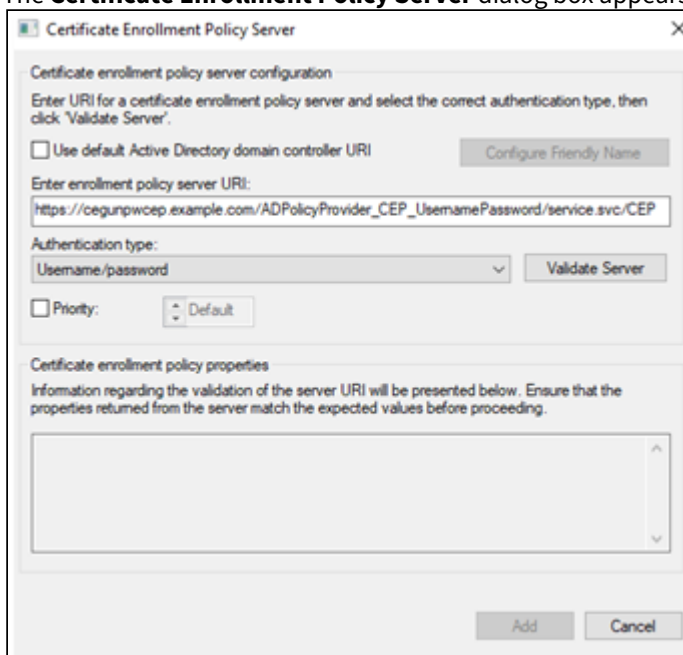


- In the tree view, expand **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies**.
- In the content pane, right-click **Certificate Services Client - Certificate Enrollment Policy > Properties**. The **Certificate Services Client - Certificate Enrollment Policy Properties** dialog box appears.



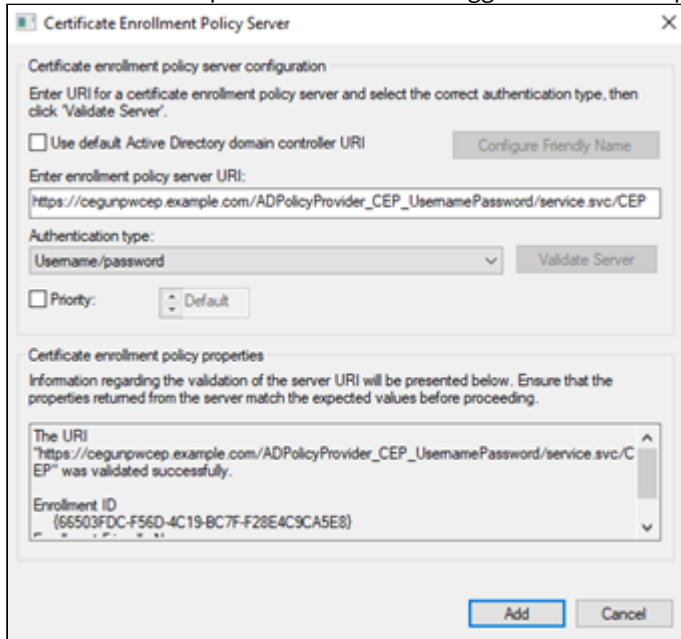
7. In the **Configuration Model** drop-down list, select **Enabled**.
8. If you are not installing WSTEP along with an existing Microsoft CA, select **Active Directory Enrollment** in the **Certificate enrollment policy list** pane, and then click **Remove**.
9. Click **Add**.

The **Certificate Enrollment Policy Server** dialog box appears.

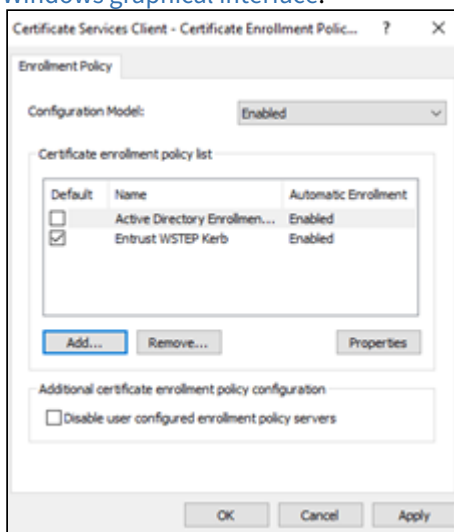


10. In the **Enter enrollment policy server URI** field, enter the Certificate Enrollment Policy Web Service URL that you obtained earlier.
11. In the **Authentication type** drop-down list, select the same authentication mode that you configured earlier in [Selecting the authentication mode of the CEP Web Service using the Windows graphical interface](#).

- Click **Validate Server**. If the selected authentication type is **Username/password**, you will be prompted for the username and password of the user logged in to the computer.



- Click **Add**. The **Certificate enrollment policy list** pane should display the friendly name of the Certificate Enrollment Policy Web Service that you specified earlier in [Assigning a friendly name to the CEP Web Service using the Windows graphical interface](#).



- In the **Certificate enrollment policy list** pane, select the checkbox for the Certificate Enrollment Policy Web Service you just added to make it the default Certificate Enrollment Policy.
- Click **OK**.

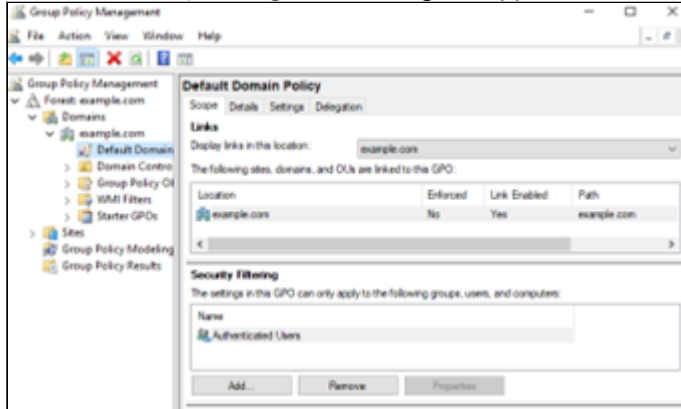
Configuring the Certificate Enrollment Policy Web Service for Windows users

To work with Certificate Enrollment Gateway, Windows domain endpoints need the Certificate Enrollment Policy Web Service URL. Complete the following procedure to add the Certificate Enrollment Policy Web Service URL to Windows domain endpoints.

To configure the Certificate Enrollment Policy Web Service for Windows users

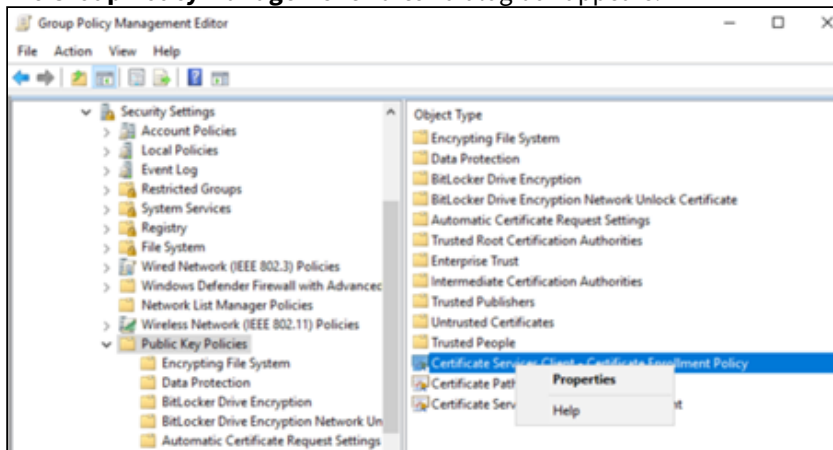
1. Log in to the server hosting Active Directory.
2. Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**.

The **Group Policy Management** dialog box appears.



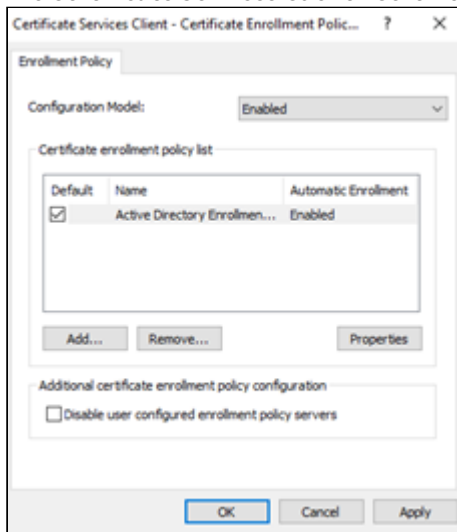
3. In the tree view, expand the Domain Controller you will modify.
4. Right-click **Default Domain Policy > Edit**.

The **Group Policy Management Editor** dialog box appears.

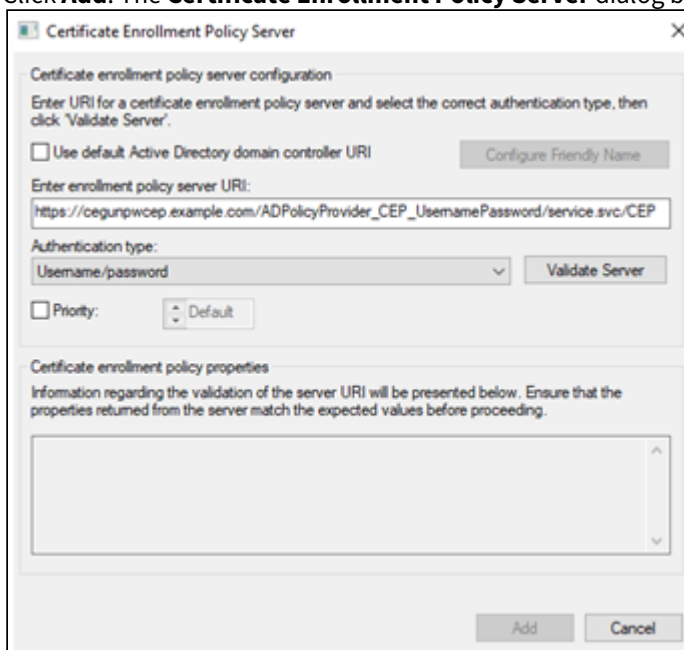


5. In the tree view, expand **User Configuration > Policies > Windows Settings > Security Settings > Public Key Policies**.

- In the content pane, right-click **Certificate Services Client - Certificate Enrollment Policy > Properties**. The **Certificate Services Client - Certificate Enrollment Policy Properties** dialog box appears.

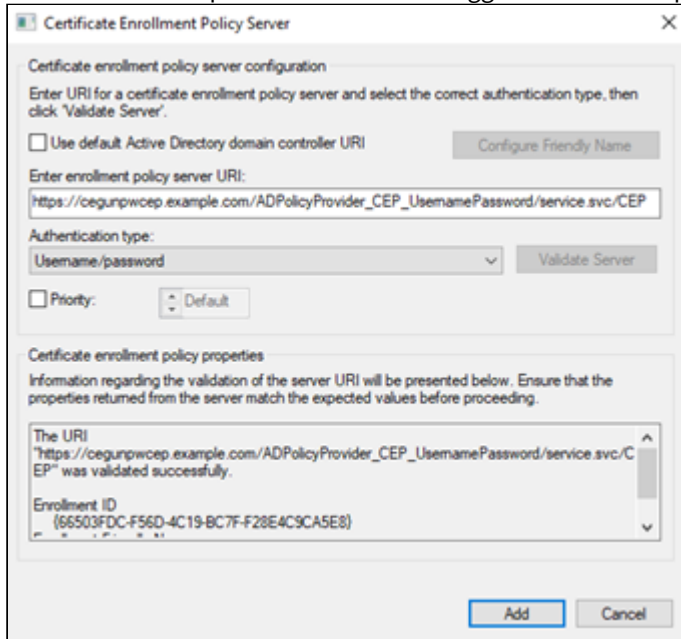


- In the **Configuration Model** drop-down list, select **Enabled**.
- If you are not installing WSTEP along with an existing Microsoft CA, select **Active Directory Enrollment** in the **Certificate enrollment policy list** pane, and then click **Remove**.
- Click **Add**. The **Certificate Enrollment Policy Server** dialog box appears.

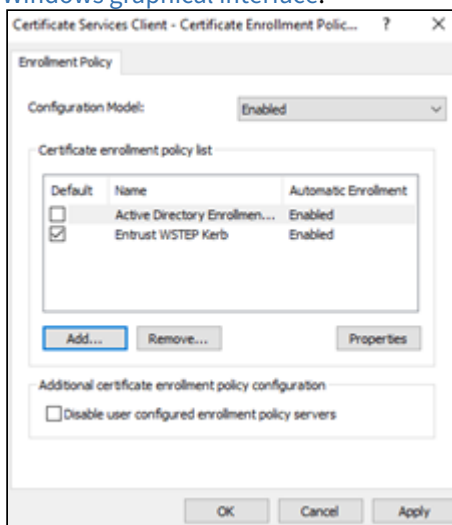


- In the **Enter enrollment policy server URI** field, enter the Certificate Enrollment Policy Web Service URL that you obtained earlier.
- In the **Authentication type** drop-down list, select the same authentication mode that you configured earlier in [Selecting the authentication mode of the CEP Web Service using the Windows graphical interface](#).

- Click **Validate Server**. If the selected authentication type is **Username/password**, you will be prompted for the username and password of the user logged in to the computer.



- Click **Add**. The **Certificate enrollment policy list** pane should display the friendly name of the Certificate Enrollment Policy Web Service that you specified earlier in [Assigning a friendly name to the CEP Web Service using the Windows graphical interface](#).



- In the **Certificate enrollment policy list** pane, select the checkbox for the Certificate Enrollment Policy Web Service you just added to make it the default Certificate Enrollment Policy.
- Click **OK**.

Enabling certificate auto-enrollment for computers and domain controllers

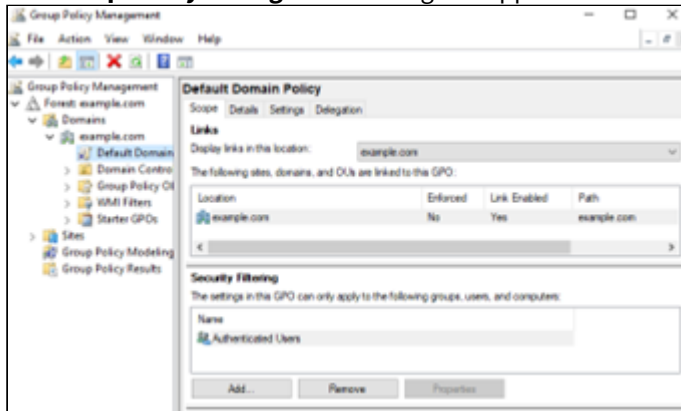
In the Windows domain, enable the certificate auto-enrollment for computers and domain controllers.

To enable certificate auto-enrollment for computers and domain controllers

- Log in to the server hosting Active Directory.

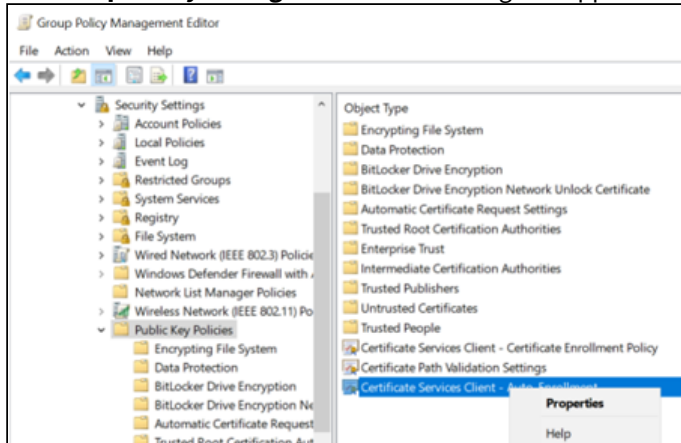
- Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**.

The **Group Policy Management** dialog box appears.



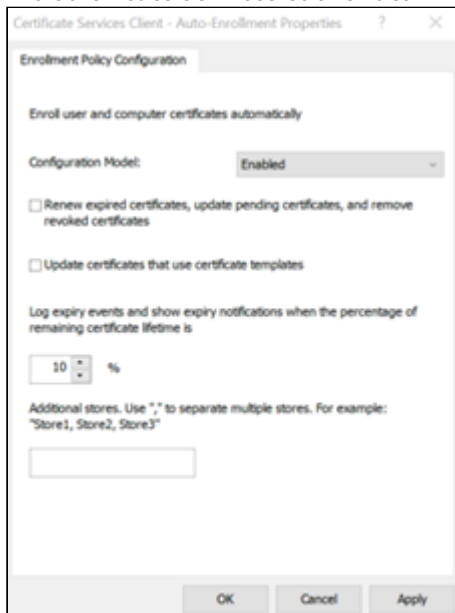
- In the tree view, expand the Domain Controller you will modify.
- Right-click **Default Domain Policy > Edit**.

The **Group Policy Management Editor** dialog box appears.



- Expand to **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies**.

- In the content pane, right-click **Certificate Services Client Auto Enrollment > Properties**. The **Certificate Services Client Auto Enrollment Properties** dialog box appears.



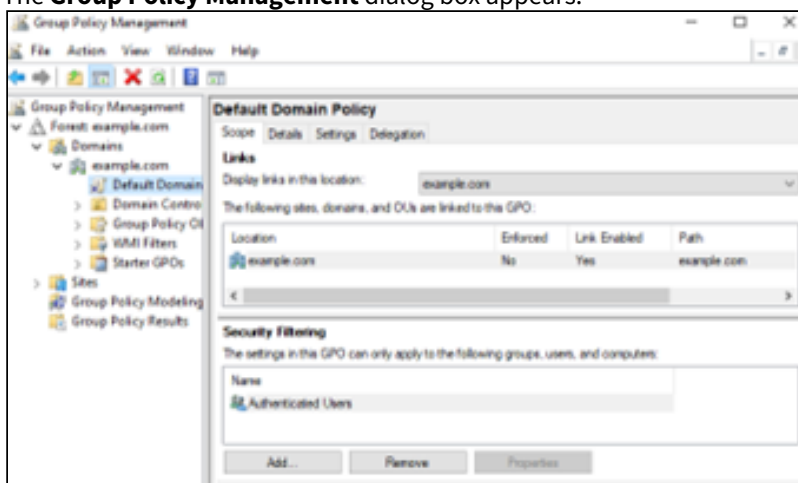
- In the **Configuration Model** drop-down list, select **Enabled**.
- Select **Renew expired certificates, update pending certificates, and remove revoked certificates**.
- Select **Update certificates that use certificate templates**.
- Click **OK**.

Enabling certificate auto-enrollment for users

In the Windows Domain, enable the certificate auto-enrollment for users.

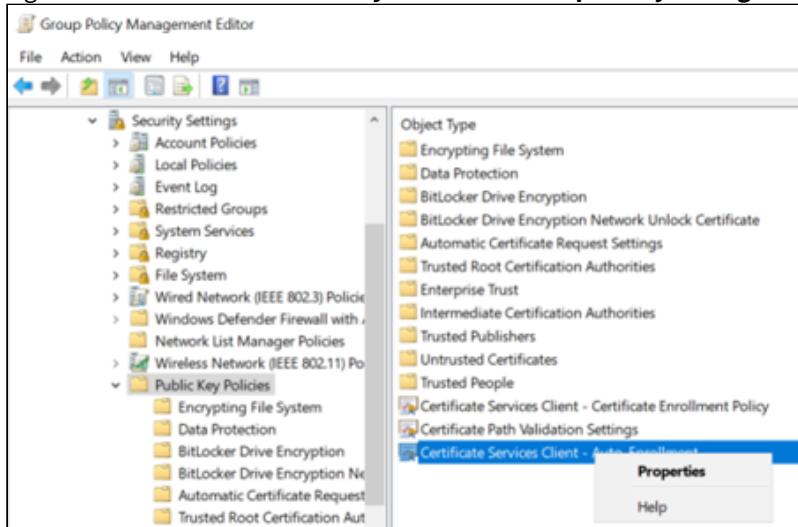
To enable the certificate auto-enrollment for users

- Log in to the server hosting Active Directory.
- Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**. The **Group Policy Management** dialog box appears.

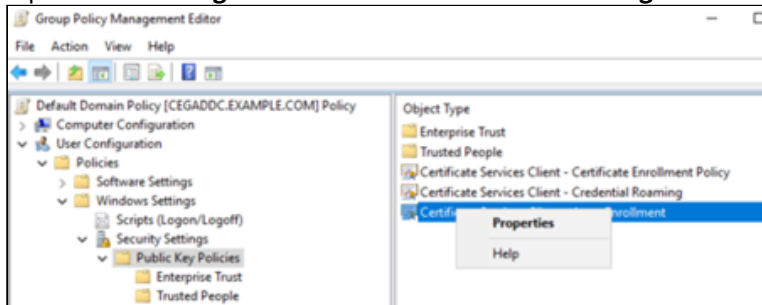


- In the tree view, expand the Domain Controller you will modify.

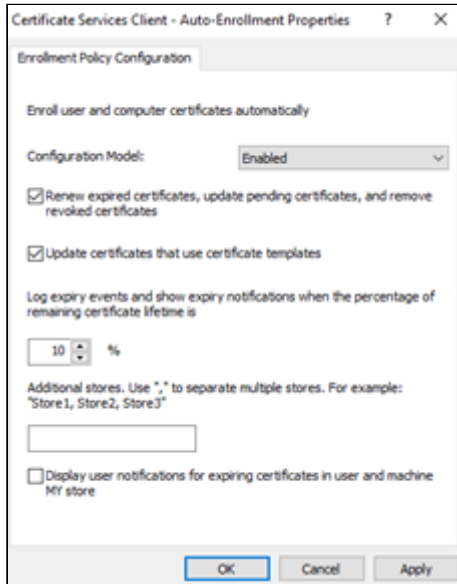
- Right-click **Default Domain Policy > Edit**. The **Group Policy Management Editor** dialog box appears.



- Expand **User Configuration > Policies > Windows Settings > Security Settings > Public Key Policies**.



- In the content pane, right-click **Certificate Services Client Auto Enrollment > Properties**. The **Certificate Services Client Auto Enrollment Properties** dialog box appears.



- In the **Configuration Model** drop-down list, select **Enabled**.
- Select **Renew expired certificates, update pending certificates, and remove revoked certificates**.
- Select **Update certificates that use certificate templates**.
- Click **OK**.

Configuring non-domain endpoints

Enrollment endpoints outside the Windows domain require the manual configuration described in this section.

- [Configuring the enrollment policy in non-domain endpoints](#)
- [Importing the root CA certificate into non-domain endpoints](#)

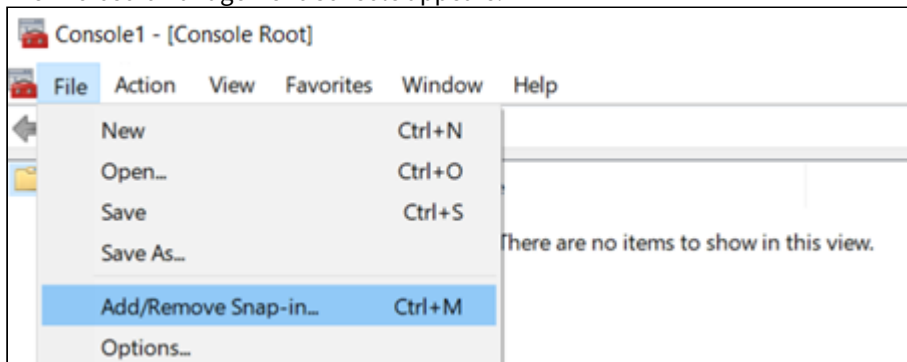
Configuring the enrollment policy in non-domain endpoints

In the enrollment endpoints outside the Windows Domain, add the enrollment policy as described in the following procedure.

To configure the enrollment policy in non-domain endpoints

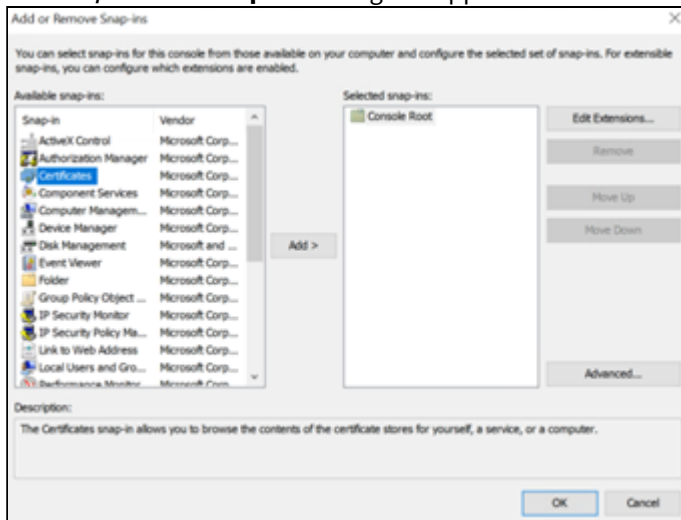
1. Log in to the non-domain endpoint.
2. Run `mmc.exe`.

The Microsoft Management Console appears.



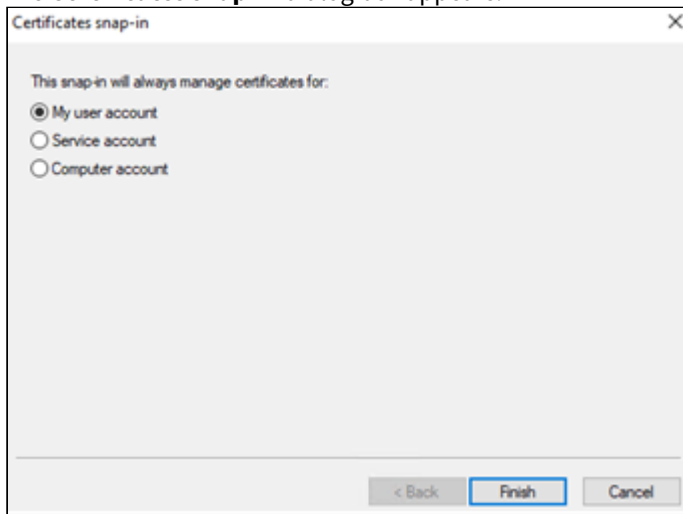
3. Select **File > Add/Remove Snap-in**.

The **Add/Remove Snap-ins** dialog box appears.

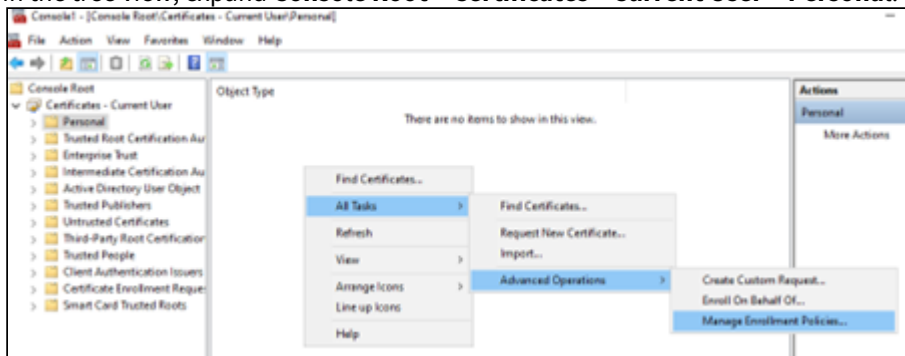


4. In the **Available snap-ins** list, select **Certificates**.

5. Click **Add**.
The **Certificates snap-in** dialog box appears.



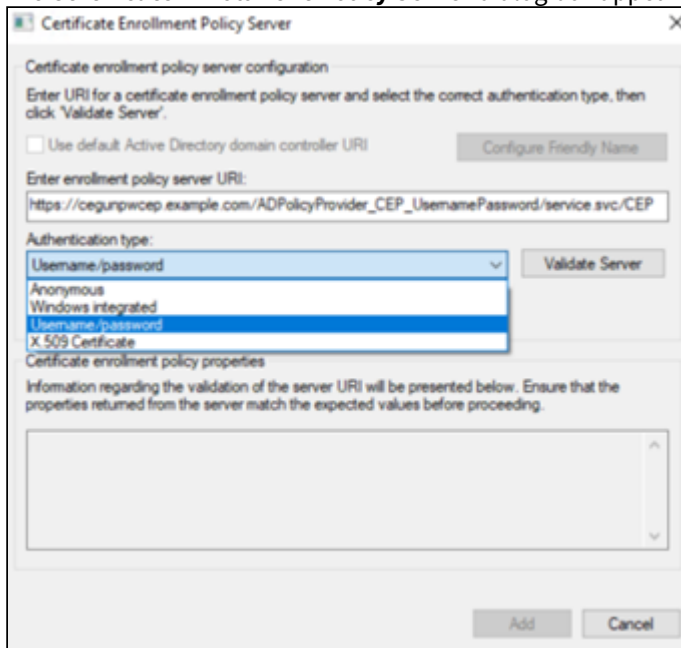
6. Select **My user account**.
7. Click **Finish** to close the **Certificates snap-in** dialog box.
8. Click **OK** to close the **Add or Remove Snap-ins** dialog box.
9. In the tree view, expand **Console Root > Certificates – Current User > Personal**.



- Right-click **Personal > All Tasks > Advanced Operations > Manage Enrollment Policies**. The **Manage Enrollment Policies** dialog box appears.



- Click **Add**. The **Certificate Enrollment Policy Server** dialog box appears.



- In the **Enter enrollment policy server URI** field, enter the URL of the Certificate Enrollment Policy Web Service that you obtained earlier in [Obtaining the URL of the Certificate Enrollment Policy Web Service](#).
- In the **Authentication Type** drop-down list, select **Username/Password**.
- Click **Validate Server**.
- When prompted, authenticate with your Windows user name and password.
- Click **Add** to add the URL and close the **Certificate Enrollment Policy Server** dialog box.
- Click **OK**.

Importing the root CA certificate into non-domain endpoints

In the enrollment endpoints outside the Windows Domain, import the certificate of the CA that will issue certificates for the enrollment service.

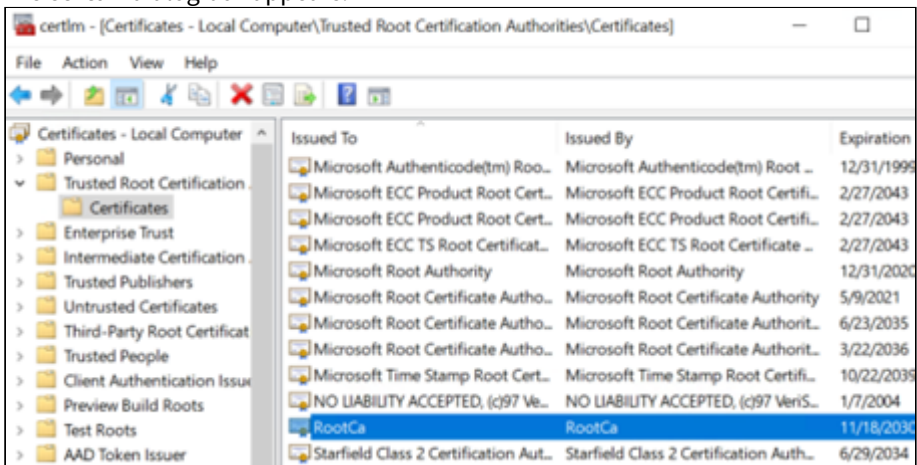
To import the root CA certificate in non-domain endpoints

1. Log in to the non-domain endpoint.
2. Open a Command Prompt window. Select **Start > Windows System > Command Prompt**.
3. Enter the following command.

```
certutil -addstore Root <cert_path>
```

Where `<cert_path>` is the full path and file name of the CA certificate file.

4. Open the Certificate Manager snap-in. Select **Start > Run**, then enter `certlm.msc`. The **certlm** dialog box appears.



5. In the tree view, expand **Certificates – Local Computer > Trusted Root Certification Authorities > Certificates**.
6. In the content pane, verify that the root CA certificate you imported appears in the list of trusted root CA certificates.

Configuring the TLS certificate of the Windows endpoints

On the server hosting the Certificate Enrollment Policy Web Service, the TLS certificate installed on Microsoft IIS is irrelevant to Certificate Enrollment Gateway. Instead, what matters is that the issuing certificate chain is trusted by all devices on the domain, along with any non-domain WSTEP client.

- If you are integrating Certificate Enrollment Gateway with an existing Windows domain, this domain already has trusted TLS certificates, and you can skip this section.
- If you are integrating a new Windows domain, follow the steps below to install the TLS certificate chain.

This section contains the following topics:

- [Obtaining the CA certificates](#)
- [Installing the CA certificates in the Active Directory domain](#)

Obtaining the CA certificates

If you used TLS Bootstrapping feature when you deployed the CEG Service, the CA certificate chain will be a `certcerts.p7b` file, located in the directory where you exported the Certificate Enrollment Gateway configuration.

For CAs hosted by Entrust PKI as a Service, you should have obtain the CA certificates from the Entrust Certificate Services portal.

Copy this file from the CEG Service host to your machine.

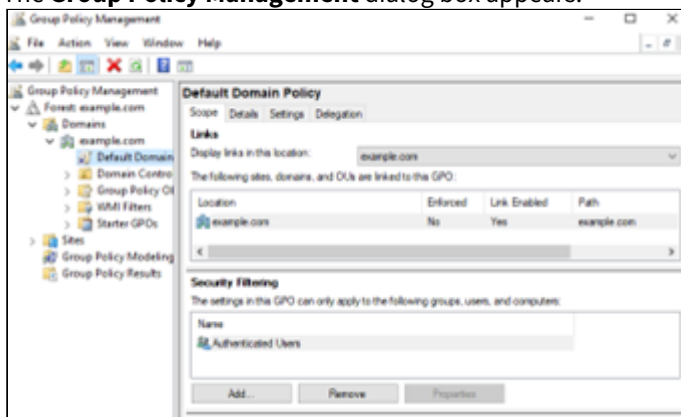
Installing the CA certificates in the Active Directory domain

In the Active Directory Domain Controller, install all certificates in the CA certificate chain as trusted root certificates.

To install the CA certificates in the Active Directory Domain Controller

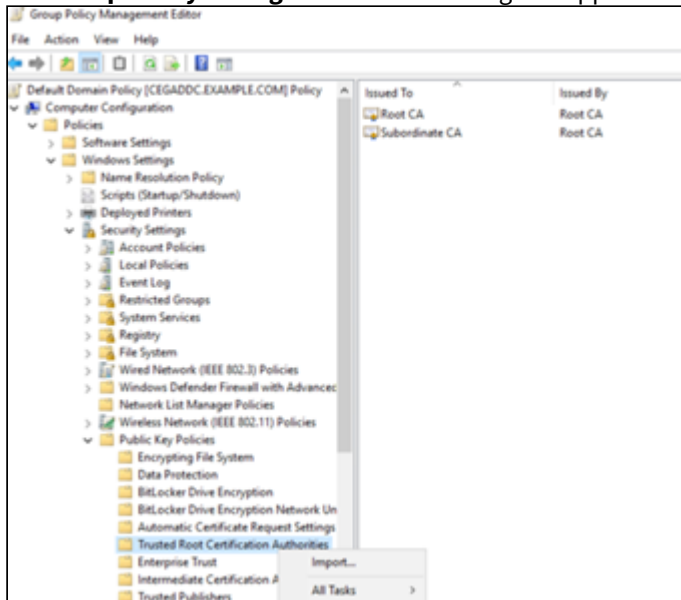
1. Log in to the server hosting Active Directory.
2. Open the Group Policy Management administrative tool. Select **Start > Windows Administrative Tools > Group Policy Management**.

The **Group Policy Management** dialog box appears.



3. In the tree view, expand the Domain Controller you will modify.
4. Right-click **Default Domain Policy > Edit**.

The **Group Policy Management Editor** dialog box appears.



5. In the tree view, expand **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies > Trusted Root Certification Authorities**.
6. Right-click **Trusted Root Certification Authorities > Import**.

7. Select the Security Manager CA certificates or the CA certificates file you obtained earlier in [Obtaining the CA certificates](#).

Starting up Certificate Hub

See below for starting up Certificate Hub.


- [Certificate Hub overview](#)
- [Preparing the Certificate Hub database](#)
- [Configuring and deploying Certificate Hub](#)
- [Managing certificates with the Certificate Hub console](#)
- [Backing up and restoring the database](#)
- [Certificate Hub error reference](#)

See [Browsing logs with Grafana](#) for how to browse Certificate Hub logs.

Certificate Hub overview

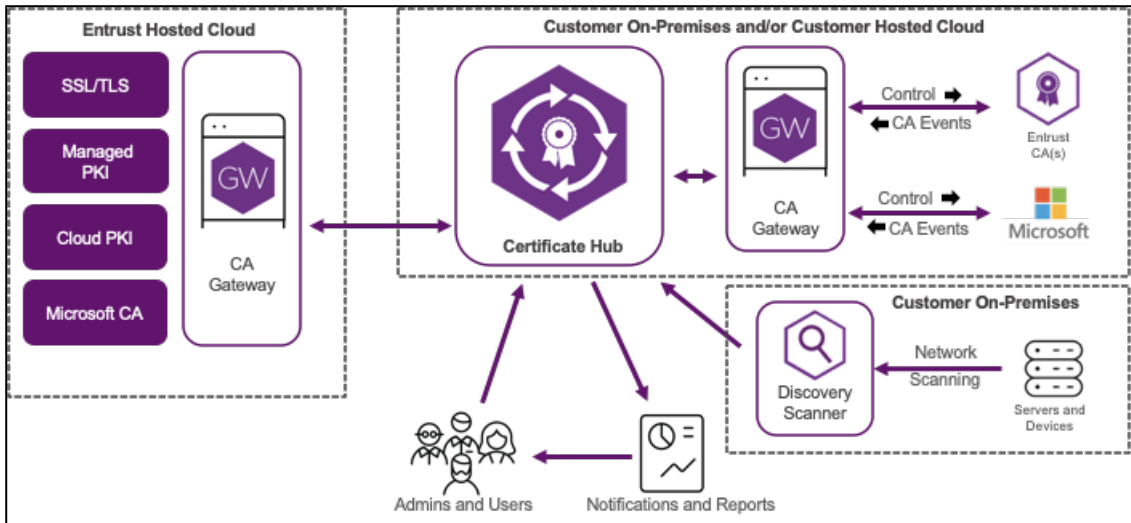
Certificate Hub has three sets of capabilities:

- The **find capabilities** inventory certificates across your organization (through network discovery) and automated certificate import (from CA databases and cloud services).
- The **control capabilities** centrally manage policy, issuance & access to public and private certificates regardless of vendor. Perform manual operations as necessary to issue, renew, and revoke certificates.
- The **automation capabilities** push keys and certificates to endpoints, with fully managed rotation and certificate profile management.
- The **report capabilities** provide organizational, issue notifications, and reports to remind certificate owners of actions they need to take.

 Administrators can customize Certificate Hub to meet enterprise needs like access permissions, system metadata, notifications, or report branding.

The high-level architecture integrates the following main components.

- [Discovery Scanners](#)
- [Entrust CA Gateway](#)
- [Certificate Hub](#)



Discovery Scanners

Certificate Hub Discovery Scanners:

- Search your enterprise's networks or portions of networks for the most recent information about deployed certificates.
- Record each certificate's location, type, algorithms, and expiry, regardless of the certificate issuer.

Discovery Scanners are typically deployed on your premises, inside corporate firewalls, to access the internal private servers. However, only Discovery Scanners require this kind of deployment; you can deploy the other Certificate Hub components in a less restrictive environment.

When started, a Discovery Scanner:

1. Contacts Certificate Hub to get the policy and scan configuration.
2. Launches the Certificate Hub scheduling process for scanning.
3. Executes one or more configured scans according to the calendar schedule and priority.
4. Periodically polls Certificate Hub for any policy and or configuration updates.

i Discovery Scanners run a custom-built version of Nmap to scan ports, capture the returned SSL certificate chain, and transmit scan results to Certificate Hub for processing.

Entrust CA Gateway

Through Entrust CA Gateway, Entrust solutions obtain a direct feed of issued certificates from each supported Certificate Authority (CA). See the following table for the CA Gateway deployment required by each type of CA.

CA type	CA Gateway deployment
Certificate Authority running on PKI Hub	Create a Certificate Authority instance, as explained in Starting up Certificate Authorities , and select the built-in CA Gateway service of this CA.

CA type	CA Gateway deployment
External Certificate Authority	Start up the Entrust CA Gateway solution and connect it with the external CA as explained Starting up CA Gateway .

Certificate Hub

Certificate Hub is a container-based set of services amenable to either customer premises or commercial cloud hosting. Certificate Hub provides:

- An API interface to the companion Certificate Hub browser UI.
- The underlying certificate database.
- The necessary background processes.

Preparing the Certificate Hub database

When configuring Certificate Hub, you must provide an external, empty database that meets the following requirements.

- [DBMS version](#)
- [Database storage](#)
- [Database permissions](#)

DBMS version

The external Certificate Hub database must be hosted on the following Database Management System (DBMS).

DBMS	Version
PostgreSQL	15 or higher

Database storage

Calculate the required database storage based on the expected certificates and reports. For example, a 1G storage is enough for 25,000 certs and a few weeks of reports.

Data	Quantity	Bytes/Item	Total
Certificates	25,000 certificates	20 KB/certificate	500 MB
Reports	200 reports	1 MB/report	200 MB
			700 MB

Database permissions

To create an external database user with sufficient permissions, connect to PSQL using the default PostgreSQL user and execute the following commands.

```
CREATE USER ${POSTGRES_USER} WITH NOSUPERUSER CREATEDB ENCRYPTED PASSWORD '${POSTGRES_PWD}';
\c postgres ${POSTGRES_USER}
CREATE DATABASE certhub;
\c certhub ${POSTGRES_USER}
CREATE EXTENSION IF NOT EXISTS pg_trgm;
```

Where:

- `${POSTGRES_USER}` is the value of the [Database User Name](#) configuration setting.
- `${POSTGRES_PWD}` is the value of the [Database User Password](#) configuration setting.

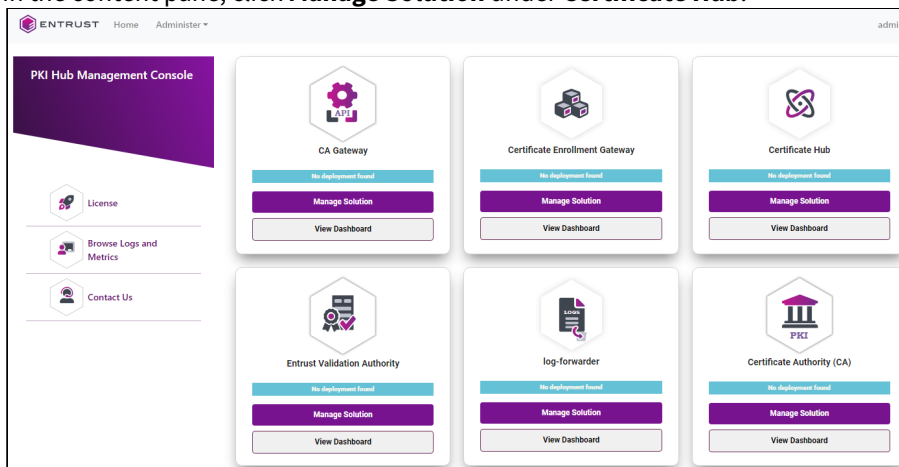
Configuring and deploying Certificate Hub

See below for configuring and deploying Certificate Hub with the Management Console.

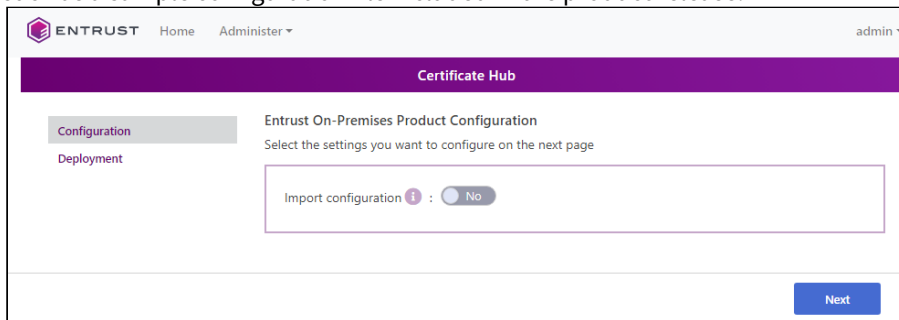
i Repeat the following steps each time a configuration update is required. Do not forget to click **Deploy** to make the changes effective.

To configure and deploy Certificate Hub with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).
2. In the content pane, click **Manage Solution** under **Certificate Hub**.



3. Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



4. Click **Next**.
5. Configure the solution settings described in the following sections.
 - [Certificate Hub Hostname](#)
 - [Discovery Scanner version](#)
 - [Initial Administrator Username](#)
 - [Initial Administrator Password](#)
 - [Initial Administrator Email](#)
 - [Database Storage Capacity](#)
 - [Name of the PostgreSQL Database](#)
 - [Database User Name](#)
 - [Database User Password](#)
 - [Host of the PostgreSQL database](#)
 - [External database port](#)
 - [SSLMode for the PostgreSQL external database](#)
 - [CA Certificate\(s\)](#)
6. Click **Validate** to validate the configured settings.
7. Correct any detected configuration error until the **Validate** option displays no warnings.
8. Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
9. Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
10. Click **Deploy**.

Certificate Hub Hostname

The base hostname or IP address routing to the Certificate Hub application.

Mandatory: No. This value defaults to the hostname or IP address hosting the appliance.

Discovery Scanner version

The username of the initial Certificate Hub administrator.

Mandatory: Yes.

Initial Administrator Username

The username of the initial Certificate Hub administrator.

Mandatory: Yes.

Initial Administrator Password

A temporary password for the initial Certificate Hub administrator.

 After the first login, users must create a new password meeting a set of password strength requirements.

Mandatory: Yes.

Initial Administrator Email

The email address of the initial Certificate Hub administrator.

Mandatory: Yes.

Database Storage Capacity

The persistent volume or capacity of the internal database in gigabytes. For example `10Gi`.

Mandatory: When using the default internal database.

Name of the PostgreSQL Database

The name of an external PostgreSQL database.

Mandatory: Yes.


Database User Name

The user name of the external PostgreSQL database.

Mandatory: Yes.

Database User Password

The user password of the external PostgreSQL database.

 Do not include special characters such as "#", "!", or "*" in the user password.

Mandatory: Yes.

Host of the PostgreSQL database

The hostname or IP address of the external PostgreSQL database.

Mandatory: Yes.

External database port

The connection port with the external PostgreSQL database.

Mandatory: Yes.

SSLMode for the PostgreSQL external database

The SSL mode for connecting with the external PostgreSQL database. Supported values are:

- require
- verify-ca
- verify-full

See <https://www.postgresql.org/docs/current/libpq-ssl.html> for a description of each mode.

 Any of the supported PostgreSQL modes requires enabling SSL.

Mandatory: Yes.

CA Certificate(s)

A file containing the root CA certificate for connecting to an external PostgreSQL database. Click **Select Files** to import this file.

Mandatory: When the [SSLMode for the PostgreSQL external database](#) value is one of the following.

- verify-ca
- verify-full

Managing certificates with the Certificate Hub console

After deploying Certificate Hub, you can log into the application console to manage the certificate lifecycle.

To manage certificates with the Certificate Hub console

1. Open a web browser in the following URL.

```
https://<host>/certhub
```


Where `<host>` is the value of the [Certificate Hub Hostname](#) configuration parameter.

2. Login with the credentials set in the [Initial Administrator Username](#) and [Initial Administrator Password](#) configuration parameters.
3. In the top-right corner of the web console, click on your username.
4. Select **Help** to display the online guide describing all the certificate management options.

Backing up and restoring the database

See below for backing up and restoring the Certificate Hub database.

- [Installing the dbctl.sh script](#)
- [Backing up the database](#)
- [Restoring the database](#)

 To backup and restore the database encryption key, the below operations use the `dbctl.sh` script included in the distribution.

Installing the dbctl.sh script

Install the `dbctl.sh` script in a folder containing the configuration of the original installation.

To install the dbctl.sh script

1. Run the `clusterctl solution config export` command to export the Certificate Hub configuration in the `<dir_path>` directory – for example:

```
sudo clusterctl solution config export -n certhub --path <dir_path>
```

2. Download the Database Management Scripts compressed file as explained in [Downloading the Entrust PKI Hub image](#).
3. Extract the `dbctl.sh` script in the `<dir_path>` directory.

Backing up the database

See below for backing up the Certificate Hub database.

- [Vacuuming the database](#)
- [Backing up the database contents](#)
- [Backing up the database encryption key](#)

 Back up the databases regularly to restore your data in case of disaster recovery.

Vacuuming the database

Run this command once before any backup to release orphaned pages from the database and decrease its size.

```
sudo dbctl.sh vacuum -n certhub
```

Backing up the database contents

Back up the Certificate Hub database using the tools provided by the DBMS.

Backing up the database encryption key

Run the following command to back up the database encryption key.

```
sudo dbctl.sh backup -n certhub
```

Restoring the database

To restore the Certificate Hub database, follow the steps below in the same Certificate Hub version used when [Backing up the database](#).

- [Restoring the configuration](#)
- [Restoring the database contents](#)
- [Restoring the database encryption key](#)
- [Completing the database restoration](#)

Restoring the configuration

Reapply the following configurations.

- [Name of the PostgreSQL Database](#)

- [External database port](#)
- [Database User Password](#)
- [CA Certificate\(s\)](#)
- [SSLMode for the PostgreSQL external database](#)
- [Database User Name](#)

Restoring the database contents

Restore the Certificate Hub database using the Database Management System (DBMS) tools.

Restoring the database encryption key

Run the following command to restore the database encryption key.

```
sudo dbctl.sh restore -n certhub --backup-file <backup-file>
```

Where `<backup-file>` is the path of the backup file.

i Before running this command, you can ignore or delete the `user-creation` and `role-update` jobs in ERROR state.

Completing the database restoration

Redeploy Certificate Hub to make effective the restoration of the database encryption key.

Certificate Hub error reference

When executed, Certificate Hub can print the following errors.

- [Certificate Hub authentication and authorization errors](#)
- [Certificate Hub administration errors](#)
- [Certificate Hub automation errors](#)
- [Certificate Hub control errors](#)
- [Certificate Hub certificate errors](#)

i See the Certificate Hub user guide for how to browse the audit logs.

Certificate Hub authentication and authorization errors

Certificate Hub throws the following authentication and authorization errors.

Code	Message
ERR_1006	Failed to hash the password for user: <code><Username></code>
ERR_1010	<code>hasPermission</code> unexpectedly invoked for <code><Permission></code>

Code	Message
ERR_1011	The util command must have a <code>--cmd</code> argument.
ERR_1012	Unknown command <code><Command></code>
ERR_1013	<code>--username</code> , <code>--password</code> , and <code>--email</code> must be supplied to the <code>createUser</code> command.
ERR_1014	Unexpected crypto error:
ERR_1015	Error creating default cert expiry rule for initial user:
ERR_1016	<code>--username</code> and <code>--role</code> must be provided.
ERR_1017	Unexpected crypto exception:
ERR_1040	Unexpected parsing error while loading auth request:
ERR_1041	Unexpected parsing error while saving auth request:
ERR_1042	Unexpected parsing error while removing auth request:
ERR_1046	Could not find password auth provider entry.
ERR_1047	Failed to hash the password for user: <code><Username></code>
ERR_1048	Cannot update non-existent user. User must have existing id.
ERR_1049	Login denied. Tenant id not found for user <code><Username></code> .
ERR_1056	More than one LDAP auth provider registration found (<code><Number of registrations></code>). Unexpected behavior may result!
ERR_1057	More than one PASSWORD auth provider registration found (<code><Number of registrations></code>). Unexpected behavior may result!
ERR_1076	Unable to create keystore: <code><CA></code>

Code	Message
ERR_1077	Cryptography issue when creating user.
ERR_1078	Cryptographic error processing password.
ERR_1079	Unable to initialize SSLContext for LDAPS
ERR_1080	More than one LDAP auth provider registration present. Unexpected results may occur.
ERR_1081	LDAP authentication error.
ERR_1082	Unexpected exception during LDAP lookup.
ERR_1083	Error closing LDAP context.
ERR_1084	Could not find Active Directory user.
ERR_1085	Error creating the daemon user:
ERR_1086	Error creating the initial user:

Certificate Hub administration errors

Certificate Hub throws the following administration errors.

Code	Message
ERR_1100	Internal error occurred
ERR_1101	Error parsing license : <Error message>
ERR_1102	Error parsing license: Epm client could not parse license
ERR_1103	Error parsing license : <Error message>
ERR_1104	Error parsing license: Epm client could not parse license

Code	Message
ERR_1105	Order Number of <Order number> uploaded license doesn't match the existing license <Customer contact reference>
ERR_1106	License revision <Revision> already uploaded.
ERR_1107	Uploaded license revision <Uploaded revision> is outdated. Current license revision : <Current revision> .
ERR_1108	Failed to create the license expiry schedule
ERR_1109	Failed to send email for license consumption
ERR_1110	Failed to send email for license expiry
ERR_1111	Failed to check the license expiry schedule
ERR_1112	Failed to delete existing license expiry schedule
ERR_1113	Failed to create the license expiry schedule
ERR_1114	Invalid plugin name: <Plugin name>
ERR_1115	Error executing plugin options for plugin: <Plugin name>
ERR_1116	Error loading plugin jar <JAR file name> . Plugin will not be loaded!
ERR_1117	Error loading plugin classloader.
ERR_1118	Plugin <Canonical name> is missing a language bundle. Plugin will not be loaded!
ERR_1119	Plugin <Canonical name> has invalid language bundle. No messages section found. Plugin will not be loaded!
ERR_1120	Plugin <Canonical name> has invalid language bundle. No languages found. Plugin will not be loaded!

Code	Message
ERR_1121	Plugin <code><Canonical name></code> has an invalid language bundle. Language <code><Key></code> is an invalid map. Plugin will not be loaded!
ERR_1122	Plugin <code><Canonical name></code> has an invalid language bundle. Language <code><Name></code> , key <code><Key></code> is invalid (<code><Value></code>). Plugin will not be loaded!
ERR_1123	Error initializing plugins! No <code><Plugin class name></code> plugins will be loaded until invalid plugin is removed!
ERR_1124	updatePlugin: Error converting global options to Json string from list
ERR_1125	validatePluginStateUpdate : cannot deactivate plugins that don't require license
ERR_1126	validatePluginStateUpdate : cannot deactivate plugin <code><Name></code> as its in use by destination : <code><Label></code>
ERR_1127	validatePluginStateUpdate : cannot deactivate plugin <code><Name></code> as its in use by source : <code><Label></code>
ERR_1128	addPlugin: Error converting global options to Json string from list
ERR_1129	Error converting global options to list from <code>Json byte[]</code>
ERR_1130	addPlugin: Error converting global options to list from <code>Json byte[]</code>
ERR_1131	Error fetching language bundle, Plugin <code><Plugin name></code> not found
ERR_1132	Failed to add an entry to the keystore: <code><TBU></code>
ERR_1133	Plugin update failed, plugin ID <code><Plugin ID></code>
ERR_1149	Failed importing multiple addresses.
ERR_1150	Failed importing single addresses.

Code	Message
ERR_1153	Failed to check the events retention schedule: <Error>
ERR_1154	Failed to create the events retention schedule: <Error>
ERR_1199	Unhandled exception caught

Certificate Hub automation errors

Certificate Hub throws the following automation errors.

Code	Message
ERR_1207	Failed to mapping existing source plugin options.
ERR_1208	Failed to process existing source plugin options.
ERR_1209	Failed to migrate existing source plugin options.
ERR_1214	Failed to send email for report <Report name> , schedule id: <Schedule ID> . Error:
ERR_1215	Failed to generate missing report: <Report ID>
ERR_1216	Failed to generate missing schedule: <Schedule ID>
ERR_1217	Failed to return report: <Report ID> . Error: <Error>
ERR_1218	Error while retrieving report data:
ERR_1219	Error while generating report:
ERR_1220	User <Username> does not have permission to edit or delete report <Report name>
ERR_1221	Error while generating report: <Error>

Code	Message
ERR_1222	User <Username> does not have permission to access artifact <Artifact ID>
ERR_1223	User <Username> does not have permission to access execution <Execution ID>
ERR_1224	Failed to check the reports retention schedule: <Error>
ERR_1225	Failed to create the reports retention schedule: <Error>
ERR_1230	Field ' <Name> ' value ' <Value> ' cannot be parsed as <Type> . Field will be treated as a String.
ERR_1231	Unexpected exception while processing rule. RULE WILL BE SKIPPED!
ERR_1232	Expiry notification is dropped for certificate <Certificate name> . The address field <Address field> is empty.
ERR_1233	Expiry notification is dropped for certificate <Certificate name> . The address field <Address field> is not referring to a text custom field.
ERR_1234	Action plugins not currently supported. THIS ACTION WILL BE SKIPPED!
ERR_1235	Exception while executing rule. RULE WILL BE SKIPPED!
ERR_1236	Error running rules engine for certificate renewal rule.
ERR_1237	Execution of action failed.
ERR_1238	FAILED processing conditions. RULE WILL BE SKIPPED!
ERR_1239	I/O issue while parsing conditions. RULE WILL BE SKIPPED!
ERR_1240	Error running rules engine for event.

Code	Message
ERR_1241	Could not parse plugin config, ACTION WILL BE SKIPPED: <Plugin config>
ERR_1242	FAILED to create the expiration rules schedule! Expiry notifications will not be sent!
ERR_1243	Error while processing event rule conditions. RULE WILL BE SKIPPED!
ERR_1244	Only NOTIFICATION actions are supported! ACTION WILL BE SKIPPED!
ERR_1254	Unexpected IOException while formatting the certificate. Error:
ERR_1255	Unexpected IOException while formatting the certificate chain. Error:
ERR_1256	Unexpected IOException while formatting the certificate. Error:
ERR_1260	FAILED to create the key manager scan schedule! Key managers will not be scanned!
ERR_1261	Error encountered while scanning key manager.
ERR_1262	Error encountered while scanning source.
ERR_1271	User <User ID> does not have permission to view, edit or delete destination <Label>
ERR_1272	Error verifying destination config <Label>
ERR_1273	Error verifying destination config for plugin <Plugin name>
ERR_1274	Error while generating report.
ERR_1275	Failed to retrieve schedule runtimes for <Schedule name>
ERR_1276	Failed to parse schedule runtimes for <Schedule name>
ERR_1280	Failed processing conditions for renewal success. RULE WILL BE SKIPPED!

Code	Message
ERR_1281	I/O issue while parsing conditions for renewal success. RULE WILL BE SKIPPED!
ERR_1282	Failed processing conditions for renewal failure. RULE WILL BE SKIPPED!
ERR_1283	I/O issue while parsing conditions for renewal failure. RULE WILL BE SKIPPED!
ERR_1289	Expiry notification is dropped for certificate <code><Certificate name></code> . The address field <code><Address field></code> is empty.
ERR_1290	Error running rules engine for certificate renewal rule.
ERR_1291	Failed processing rule. RULE WILL BE SKIPPED!
ERR_1292	I/O issue while running rule. RULE WILL BE SKIPPED!
ERR_1293	Expiry notification is dropped for certificate <code><Certificate name></code> . The custom field <code><Custom field></code> is empty.
ERR_1294	Expiry notification is dropped for certificate <code><Certificate name></code> . The address field <code><Address field></code> is not referring to a text custom field.
ERR_1295	Expiry notification is dropped for certificate <code><Certificate name></code> . The address field <code><Address field></code> is empty.
ERR_1296	Error running rules engine for certificate renewal rule.
ERR_1299	Execution of rule action failed.

Certificate Hub control errors

Certificate Hub throws the following control errors.

Code	Message
ERR_1302	Error getting authority capabilities from CAGW

Code	Message
ERR_1303	Failed to check the domain sync trigger
ERR_1304	Unable to add domain sync for authority
ERR_1305	Internal error contacting CAGW.
ERR_1306	Error while reading XML stream from upload.
ERR_1307	Unexpected exception while pushing certificate:
ERR_1308	HTTP Error while uploading certificate: <Error> :\n <Response body>
ERR_1309	Error while uploading certificate: <Error>
ERR_1310	Unable to parse properties for domain: <Domain name>
ERR_1311	User <User ID> doesn't have access to authority <Authority ID>
ERR_1312	Internal error contacting CAGW.
ERR_1313	Unable to get profiles for authority
ERR_1314	Unable to get the subject DN for authority
ERR_1315	Unable to get the Capabilities for authority
ERR_1316	Unexpected error contacting CAGW: <Error>
ERR_1330	User <User ID> does not have permission to view, edit or delete key manager <Key manager>
ERR_1331	Error verifying key manager config <Key manager label>
ERR_1332	saveOrUpdateKeyManager: Error converting plugin options to Json string from list

Code	Message
ERR_1333	Error converting plugin options to list from Json byte[]
ERR_1334	Error verifying key manager config for plugin <Plugin name>
ERR_1349	Failed to sync domains, Error from CAGW: <Error>
ERR_1350	Unexpected response received from CAGW
ERR_1351	Internal error contacting CAGW
ERR_1352	Unexpected response received from CAGW: <Error>
ERR_1353	Unexpected response received from CAGW: <Error>
ERR_1354	Unexpected response received from CAGW: <Error>
ERR_1355	Error configuring the SSL client connection to the CAGW APIs.
ERR_1356	Error configuring the SSL client connection to the CAGW APIs.
ERR_1357	Error configuring the SSL client connection to the CAGW APIs
ERR_1358	Error configuring the SSL client connection to the CAGW APIs.
ERR_1359	Error configuring the SSL client connection to the CAGW APIs.
ERR_1362	Error parsing authority certificate validity period: <Certificate validity period>
ERR_1363	Error parsing authority certificate validity period: <Certificate validity period>
ERR_1374	Error response from CAGW: <Error>
ERR_1375	Unable to parse properties for domain: <Domain name>

Code	Message
ERR_1376	Internal error contacting CAGW: <Error>
ERR_1377	Internal error contacting CAGW.
ERR_1378	Internal error contacting CAGW.
ERR_1379	Internal error contacting CAGW while responding to an authority request.
ERR_1380	Failed to create the authority domain sync schedule for authority <Authority ID>
ERR_1381	Failed to delete the authority domain sync schedule for authority <Authority ID>
ERR_1382	Certificate Authority <Authority ID> not found
ERR_1383	Unable to parse plugin options for authority <Authority ID> :
ERR_1384	Error response from CAGW while getting domain: <Domain name>
ERR_1385	Failed to get domain. Error from CAGW: <Error>
ERR_1386	Failed to submit domain, Error from CAGW: <Error>
ERR_1387	Unable to fetch whois record from server <Server name> . Error:
ERR_1388	Unable to close whois client connection with server <Server name> . Error:
ERR_1389	Unable to fetch whois record from default host. Error:
ERR_1390	Unable to close whois client connection with default server. Error:
ERR_1392	Error on DNS lookup : <Error>
ERR_1394	Failed to submit domain, Error from CAGW: <Error>

Code	Message
ERR_1397	Certificate Authority <Authority ID> not found
ERR_1398	Unable to parse plugin options for authority <Authority ID>
ERR_1399	Unable to import/update domain id <Domain ID> due to Json parsing error from authority <Authority ID>

Certificate Hub certificate errors

Certificate Hub throws the following certificate errors.

Code	Message
ERR_1426	Renewal failed. Missing certificate id.
ERR_1427	Failed auto renewal for certificate <Certificate ID> .
ERR_1428	Automated renewal failed for certificate <Certificate ID> due to certificate processing error
ERR_1430	Automated renewal failed for certificate <Certificate ID> due to destination errors: <List of errors>
ERR_1431	Failed to find the renewal daemon user for auto renewal
ERR_1432	Failed to create the renewal schedule for cert <Certificate serial> : <Error>
ERR_1433	Failed to check the renewal schedule <Error>
ERR_1434	Failed to create the renewal schedule <Error>
ERR_1435	Adding definition for custom field with duplicate display order : <Label> of type <Type> at position <Display order>

Code	Message
ERR_1436	Deleting definition for custom field with Id : <Metadata ID> failed as it is in use by <Certificates using metadata> certificates
ERR_1437	Updating definition for custom field with duplicate display order : <Label> of type <Type> at position <Display order>
ERR_1438	Updating definition for custom field with Id : <Metadata ID> failed as it is in use by <Certificates using metadata> certificates
ERR_1439	Updating definition for custom field with duplicate display order : <Metadata values>
ERR_1440	Other certificate custom field definitions exists with same display order <List>
ERR_1441	Updating definition for custom field with Id : <Metadata ID> failed as one of its value <List> is in use by <Certificates> certificates
ERR_1442	Error parsing the value <Value> for custom field <Metadata ID>
ERR_1443	Unsupported Operator <Operator> for custom field Id: <Metadata ID>
ERR_1450	Could not unarchive certificate because entitlement limit reached.
ERR_1452	Error response from CAGW <Error>
ERR_1453	Error exporting a certificate: <Error>
ERR_1454	Failed to parse certificate <Certificate name> stored in DB. Error: <Error>
ERR_1455	Certificate Chain is not available for export
ERR_1456	Error while exporting certificate: <Error>

Code	Message
ERR_1457	Error saving chain to keystore for export of: <Certificate name>
ERR_1458	Error adding P12 to response stream
ERR_1459	Unable to parse response from CAGW to export certificate for : <Certificate name> .Error: <Error>
ERR_1460	Certificate can not be exported since the issuing Authority is not known
ERR_1461	Certificate Authority not found
ERR_1462	Error adding P12 to response stream
ERR_1463	Unexpected response received from CAGW when exporting a certificate
ERR_1464	Internal error contacting CAGW
ERR_1465	Failed to export certificate for <Certificate name> . Error from CAGW: <Error>
ERR_1466	Failed to export certificate for <Certificate name> with serial number <Certificate serial number> . Certificate key is not backed up.
ERR_1467	Unable to parse response from CAGW to export certificate for : <Certificate name> .Error: <Error>
ERR_1468	Export private key is not supported for export type <Type> You can uncheck \\\"Include Private Key\\\" and try again, however, your exported certificate will not have the private key
ERR_1469	Export certificate chain is not supported for export type <Type> You can uncheck \\\"Include Certificate Chain\\\" and try again, however, your exported certificate will not have certificate chain
ERR_1470	Public certificate must be requested for export type <Type>

Code	Message
ERR_1471	At least one of public certificate, certificate chain or private key must be requested for export type <code><Type></code>
ERR_1472	At least one of public certificate, certificate chain or private key must be requested for export type <code><Type></code>
ERR_1473	Unable to revoke the authority <code><Authority name></code>
ERR_1474	Unable to unhold the authority <code><Authority name></code>
ERR_1477	Error building certificate query with filter : <code><Filter></code> . Error <code><Error></code>
ERR_1478	Error fetching certificates with predicate : <code><Predicate></code> . Error <code><Error></code>
ERR_1479	Certificate Bulk Edit Error: 'certificatesFilter' missing from request body
ERR_1480	Certificate Bulk Edit Error: If 'clearOutAccessTags' is set, 'accessTags' must be empty.
ERR_1481	Certificate Bulk Edit Error: No updated values provided
ERR_1482	Certificate Bulk Edit Error building certificate query with filter : <code><Filter></code> , Error: <code><Error></code> .
ERR_1483	Certificate Bulk Edit Error building certificate query with filter : <code><Filter></code> , Error: <code><Error></code> .
ERR_1484	Certificate Bulk Edit Error updating certificates with filter : <code><Filter></code> , Error: <code><Error></code>
ERR_1486	Certificate unhold error : Could not find certificate with id: <code><Certificate ID></code> .
ERR_1487	Certificate unhold error : No Authority Id associated with this certificate: <code><Certificate ID></code> .

Code	Message
ERR_1488	Certificate unhold error : Cannot unhold certificate <Certificate ID> . Authority is not active : <Authority ID> .
ERR_1489	Certificate unhold error : Cannot unhold certificate <Certificate ID> . No external id found.
ERR_1490	Issue certificate error : Subject DN is required for CSR.
ERR_1491	Issue certificate error : CAGW failed to create certificate for authority <Authority ID>
ERR_1492	Issue certificate error : CAGW Failed to create certificate: <Key manager ID> .
ERR_1493	Issue certificate error : Subject DN is required for CSR.
ERR_1494	Issue certificate error : Subject DN is required for CSR.
ERR_1495	Issue certificate error : Subject DN is required for CSR.
ERR_1496	Issue certificate error : Subject DN is required for CSR.
ERR_1497	Failed to save certificate: <Error>
ERR_1498	Failed to upload certificate to the key manager <Key manager ID> . Error <Error>
ERR_1499	Certificate revoke error : No Authority Id associated with this certificate. <Certificate ID>
ERR_1500	Certificate revoke error : Cannot revoke certificate <Certificate ID> . Authority <Authority ID> is not active.
ERR_1501	Certificate revoke error : Cannot revoke certificate <Certificate ID> . No external id found.
ERR_1502	Failed to apply service-level filters on query <Filter>

Code	Message
ERR_1503	Failed to apply service-level filters on query <Filter>
ERR_1504	Failed to apply service-level filters on query <Filter>
ERR_1505	Could not find certificate with id <Certificate ID>
ERR_1506	Could not find certificate with id <Certificate ID>
ERR_1508	Failed to issue a certificate from authority <Authority name> . Error <Error>
ERR_1509	Failed to parse the X509 certificate <Certificate body> \n Message: <Error>
ERR_1510	Unable to find certificate <Certificate ID>
ERR_1511	Failed to process the certificate <Certificate import request body> \n Message: <Error>
ERR_1512	Failed to process the certificate <Certificate body> \n External ID: <Certificate External ID> \n Message: <Error>
ERR_1513	Failed to apply service-level filters on query <Filter>
ERR_1514	Failed to run the certificate count query: <Error>
ERR_1521	Error verifying source config <Source Label>
ERR_1522	Error verifying source config for plugin <Plugin name>
ERR_1523	addOrUpdateSource: Error converting plugin options to Json string from list
ERR_1524	Error scheduling source sync, sources will not be scanned!
ERR_1525	Error creating certificate from certificate request


Code	Message
ERR_1526	Failed to send new external certificate request notification to approver(s). Error: <Notification message>
ERR_1527	Failed to send external certificate request cancellation notification to requestor. Error: <Notification message>
ERR_1528	Failed to send certificate request approval notification to requestor. Error: <Notification message>
ERR_1529	Failed to send certificate request rejection notification to requestor. Error: <Notification message>
ERR_1530	Failed to send new certificate request notification to internal requestor. Error: <Notification message>
ERR_1531	Failed to send new internal certificate request notification to approver(s). Error: <Notification message>
ERR_1533	Failed to send new certificate request notification to external requestor. Error: <Notification message>
ERR_1534	CSR key algorithm <CSR key algorithm> does not match the required key algorithm <Allowed key algorithm>
ERR_1536	CSR key algorithm keysize <CSR key size> does not meet minimum public key size required: <Allowed key size>
ERR_1537	Invalid certificate signing request provided
ERR_1540	Failed to send new certificate request notification to external requestor. SMTP Notification Plugin not found
ERR_1541	Failed to send certificate request cancellation notification to external requestor. SMTP Notification Plugin not found
ERR_1542	Failed to send new external certificate request notification to approver(s). SMTP Notification Plugin not found

Code	Message
ERR_1543	Failed to send new certificate request notification to certhub admin. SMTP Notification Plugin not found
ERR_1544	Failed to send new internal certificate request notification to approver(s). SMTP Notification Plugin not found
ERR_1545	Failed to send notification for certificate request cancellation. SMTP Notification Plugin not found
ERR_1546	Failed to send notification for certificate request approval. SMTP Notification Plugin not found
ERR_2010	Found invalid certificate with name <code><Certificate name></code> .
ERR_2011	Unexpected exception while processing certificate.
ERR_2012	Error processing certificate.
ERR_2013	Error creating certificate factory.
ERR_2015	Failed to parse certificate <code><Certificate name></code> stored in DB. Error: <code><Error message></code>

Starting up Timestamping Authority

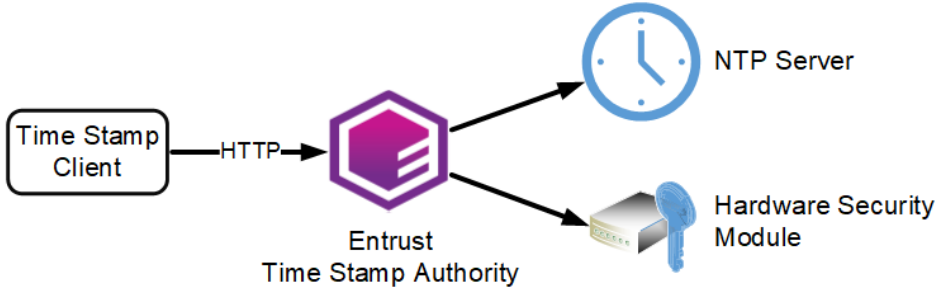
See below for starting up Validation Authority.

- [Timestamping Authority overview](#)
- [Loading the HSM configuration on Timestamping Authority](#)
- [Verifying port access for Timestamping Authority](#)
- [Configuring Authority Security Manager for Timestamping Authority](#)
- [Generating a timestamping certificate and key pair](#)
- [Configuring and deploying Timestamping Authority](#)
- [Testing the timestamping service](#)
- [Browsing Timestamping Authority logs](#)
- [tsactl reference](#)
- [Troubleshooting Timestamping Authority](#)

 As explained in [Manually starting starting the chrony service](#), you must manually start the `chrony` service after restarting a node.

Timestamping Authority overview

Timestamping Authority responds to timestamp requests to prove the existence of certain data before a given time. To respond these requests, Timestamping Authority connects with different components.



In this architecture:

- Multiple clients send requests to the timestamping service of Timestamping Authority.
- One or several Hardware Security Modules (HSMs) manage the timestamp signing key.
- One or several Network Time Protocol (NTP) servers provide an accurate and reliable time source.

Loading the HSM configuration on Timestamping Authority

See the following table for the required command for loading the configuration of the Hardware Security Module (HSM).

HSM	Command
Entrust nShield HSM	<code>tsactl import-nshield</code>
Thales HSM	<code>tsactl import-thales</code>

See also:

- [HSM requirements](#) for the list of supported HSMs.
- [Configuring an nShield HSM](#) for the additional steps required by Entrust nShield HSMs.

i Skip this step if you use software cryptography (not recommended).

Verifying port access for Timestamping Authority

In addition to the listed described in [Required open ports](#), ensure no network restriction blocks access to the following ports.

- [Incoming traffic](#)
- [Outgoing traffic](#)

i Timestamping Authority deployment automatically opens these ports in the firewall of the machines hosting Entrust PKI Hub.

Incoming traffic

The Timestamping Authority deployment automatically opens the following ports for incoming traffic in the firewall of the host machines.

Target Port	Protocol	Source	Target Service
80	TCP/HTTP	Timestamp client	Timestamp responder
323	NTP	TSA clock service	chrony NTP client

Outgoing traffic

The Timestamping Authority deployment automatically opens the following ports for outgoing traffic in the firewall of the host machines.

Target Port	Protocol	Source	Target Service
1792	NTLS	TSA	Luna Network HSM (if any)
9000-9004	TCP/HTTPS	TSA	nShield HSM (if any)

Configuring Authority Security Manager for Timestamping Authority

If you will use Authority Security Manager for [Generating a timestamping certificate and key pair](#), you may need to create a new certificate type. Otherwise, you can skip this section.

i To issue the Timestamping Authority certificate with Entrust Authority Security Manager, you may need to create a new certificate type. In the latest Entrust Authority Security Manager 10.0.x releases, a Time-Stamp Authority (TimeStamp_1K) certificate type may already be predefined in the certificate specifications. This certificate type includes the proper certificate extensions for signing timestamp responses.

The following procedures describe how to create the Time-Stamp Authority (TimeStamp_1K) certificate type if it does not already exist.

To add the Time-Stamp Authority certificate type to Entrust Authority Security Manager

1. Log in to Entrust Authority Security Manager Administration.
2. Select **File > Certificate Specifications > Export** and export the certificate specifications.
3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```
TimeStamp_1k=enterprise,Time-Stamp Authority,Time-Stamp Authority certificate
-no directory entry
```

5. Add the following lines to the `[Extension Definitions]` section.

```

;-----
;- Cert Type: TimeStamp_1k
;- This cert type needs to be mapped to cert def policy enforcing:
; - Certificate lifetime:
; - Exclude privateKeyUsagePeriod: 1
; - Exclude basicConstraints: 1
; - Exclude entrustVersInfo: 1
;-----
[TimeStamp_1k Certificate Definitions]
1=Verification
;
[TimeStamp_1k Verification Extensions]
;Key Usage: Digital Signature
keyusage=2.5.29.15,n,m,BitString,1
;Extended Key Usage: Time Stamping
extkeyusage=2.5.29.37,c,o,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.8.
; Certificate Policies: DER encode the <Policy-OID>
; Policy-OID=<Policy-OID> - This OID is optional, the customer might not have a
policy OID.
;certificatepolicies=2.5.29.32,n,o,DER,<DER encoded value of the above OID>
; AuthorityInfo Access:
; - OCSP server URL: <OCSP-HTTP-URL>
; - Issuing CA certificate URL: <CA-Cert-HTTP-URL>
;aia=1.3.6.1.5.5.7.1.1,n,m,DER,<DER encoded value of the above two URLs>
;

```

6. (Optional.) You can add a certificatePolicies extension to the certificate type. The certificatePolicies extension contains policy information, such as how your CA operates and the intended purpose of the issued certificate. Typically, different certificate policies will relate to different applications which may use the certified key. The Certificate Policies extension contains a sequence of one or more policy information terms. Each policy information term consists of an object identifier (OID) and optional qualifiers. In an end entity certificate, the policy information terms indicate the policy under which the certificate has been issued, and the purposes for which the certificate may be used. To add a certificatePolicies extension to the certificate type:
 - a. DER-encode a list of one or more policy OIDs. Entrust provides an entDerEncoder utility for Security Manager that you can use to DER-encode data for certificate extensions. For instructions about using the entDerEncoder utility, see the Security Manager documentation.
 - b. Uncomment the `certificatepolicies=` entry and replace `<DER encoded value of the above OID>` with the DER-encoded value you obtained in the previous step.
7. (Optional.) You can add an authorityInformationAccess extension to the certificate type. The Authority Information Access (AIA) certificate extension indicates how to access information and services for the CA that issued the certificate. Information and services may include online validation services and CA policy data. To add a certificatePolicies extension to the certificate type:
 - a. DER-encode the HTTP URL of the CA certificate. Entrust provides an entDerEncoder utility for Security Manager that you can use to DER-encode data for certificate extensions. For instructions about using the entDerEncoder utility, see the Security Manager documentation.
 - b. Uncomment the `aia=` entry and replace `<DER encoded value of the above URL>` with the DER-encoded value you obtained in the previous step.
8. Add the following lines to the `[Advanced Settings]` section.

```
[TimeStamp_1k Advanced]
noBasicConstraints=1
noPrivateKeyUsage=1
noEntrustVersInfo=1
cdpLdapDnLast=1
noUserInDirectory=1
;noCRLDistPoints=1
```

9. Save and close the file.
10. Select **File > Certificate Specifications > Import** and import the certificate specifications back into Entrust Authority Security Manager.

Generating a timestamping certificate and key pair

Each TSA issuer in Timestamping Authority needs a certificate to sign timestamping responses. You can:

- Use a different certificate for every TSA issuer.
- Share a certificate among multiple TSA issuers.


See below for generating a timestamping certificate and key pair.

- [Generating a timestamping key pair](#)
- [Issuing a timestamping certificate](#)

Generating a timestamping key pair

To generate the timestamping key pair, run the `tsactl create-key` command in any Entrust PKI Hub node. The command will output a CSR that you can use to generate the certificate – for example:

```
$ sudo tsactl create-key -k RSA2048 -s "CN=TSA" -o /tmp/certreq.txt -t mytoken -v
thales
Created key with id 4a00a4617d1afd5ad626955132dd0d396a69ed24
CSR:
-----BEGIN CERTIFICATE REQUEST-----
MIICqDCCAZACAQAwMzExMC8GA1UEAxMoNGEwMGE0NjE3ZDFhZmQ1YWQ2MjY5NTUx
...
etTv+pac+nJKW8fw
-----END CERTIFICATE REQUEST-----
```

 As explained in [tsactl create-csr](#) you can create a certificate request for a key that already exists on the HSM.

Issuing a timestamping certificate

Use your CA to process the CSR obtained in [Generating a timestamping key pair](#) and issue a certificate with the following extension values.

Extension	Value
Key Usage	digitalSignature
Extended Key Usage	timeStamping


See below for how to issue this certificate with Entrust products.

- [Issuing a timestamping certificate with Entrust Security Manager](#)
- [Issuing a timestamping certificate with the Certificate Authorities solution](#)

Issuing a timestamping certificate with Entrust Security Manager

If you are using Entrust Authority Security Manager, you can run one of the following applications to issue the timestamping certificate:

- The CSR Enrollment Services (CSRES) provided by the Entrust Administration Services.
- The Profile Creation Utility included in products such as Entrust Administration Services or available as a separate download with Entrust CA Gateway.

 See [Configuring Authority Security Manager for Timestamping Authority](#) for instructions on configuring Entrust Authority Security Manager.

Issuing a timestamping certificate with the Certificate Authorities solution

Follow the steps below to issue a timestamping certificate using the Certificate Authorities solution provided by PKI Hub.

- [Creating a Certificate Authority to issue timestamping certificates](#)
- [Creating a timestamping certificate request](#)
- [Processing the timestamping certificate request](#)

Creating a Certificate Authority to issue timestamping certificates

Follow the steps described in [Starting up Certificate Authorities](#) to create:

- A root Certificate Authority.
- An issuing Certificate Authority with at least one signature profile – for example, the `wstep-digital-signature` profile described in [Active Directory \(WSTEP\) certificate profiles](#).

Creating a timestamping certificate request

Create a CA Gateway certificate enrollment request – for example:

```

1  {
2    "csr": "MIIDVzCCAb8...",
3    "profileId": "wstep-digital-signature",
4    "requiredFormat": {
5      "format": "X509"
6    },
7    "optionalCertificateRequestDetails": {
8      "extensions": [

```



```

9      {
10         "oid": "2.5.29.37",
11         "critical": true,
12         "value": "MAoGCCsGAQUFBwMI"
13     }
14 ],
15     "validity_period": "2024-11-06T13:00Z/2026-07-06T13:00:00Z"
16 }
17 }

```

See below for the values required by each request field.

Field	Line	Value
csr	2	The base64-encoded Certificate Signing Request previously generated in Generating a timestamping key pair as a single line.
extensions	8	The same fields and values as in the above example (to select the timestamping extended key usage).
validity_period	15	The validity period for the issued certificate. The expiry date in this period cannot exceed the expiry date of the issuing CA certificate.

Processing the timestamping certificate request

See below for processing the timestamping certificate request and obtaining the issued certificate.

To process the timestamping certificate request

1. Use a REST client to process the request as explained in [Issuing certificates with a REST client](#).
2. Edit the REST response and copy the base64-encoded string in the `body` field.
3. Save the string in a file with the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` delimiters – for example:

```

-----BEGIN CERTIFICATE-----
MIIICDCCBlIqAwIBAgITMwEf/Fvr7NDwanyeRAAAAR/8WzANBgkqhkiG9w0BAQwF
ADBdMQswCQYDVQQGEwJVUzEeMBwGA1UEChMVTWljcm9zb2Z0IENvcnBvcnF0aW9u
...
Ud085g==
-----END CERTIFICATE-----

```

4. Use this certificate file as the [TSA certificate](#) when configuring the solution.

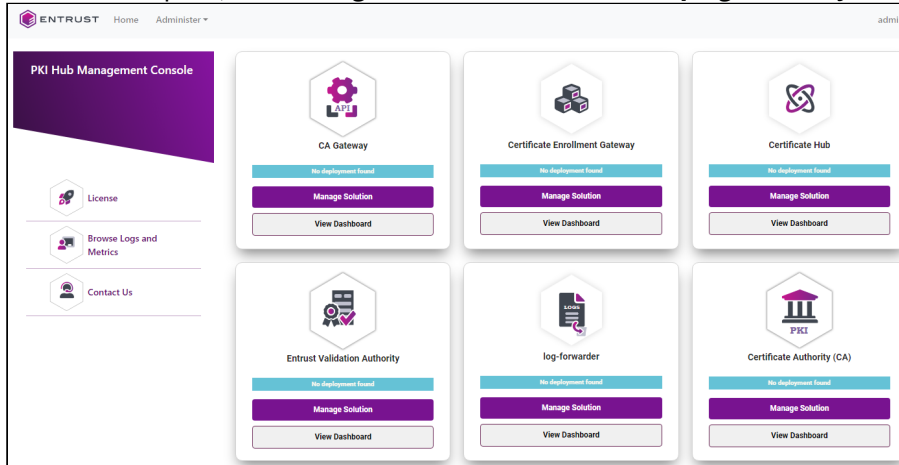
Configuring and deploying Timestamping Authority

See below for configuring and deploying Timestamping Authority with the Management Console.

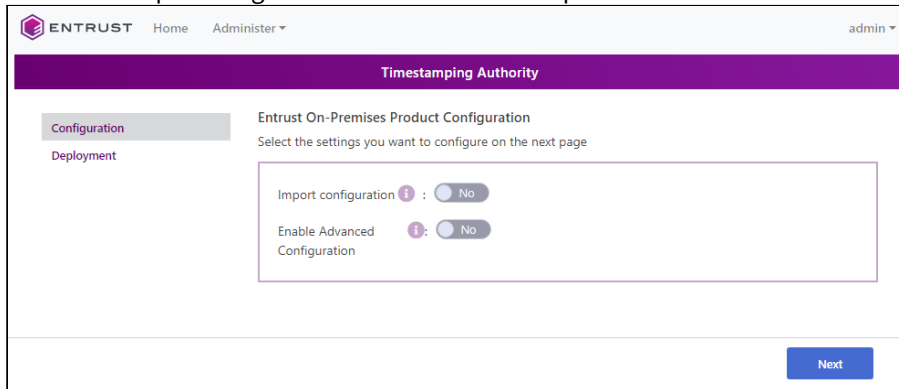
i Repeat the following steps each time a configuration update is required. Do not forget to click **Deploy** to make the changes effective.

To configure and deploy Timestamping Authority with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).
2. In the content pane, click **Manage Solution** under **Timestamping Authority**.



3. Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



4. Active the **Enable Advanced Configuration** if you want to configure the full set of configuration parameters supported by the solution.
5. Click **Next**.
6. Configure the solution settings described in the following sections.
 - [Hsm](#)
 - [Tsa Server](#)
 - [Clock service](#)
 - [Tsa issuers](#)
7. Click **Validate** to validate the configured settings.
8. Correct any detected configuration error until the **Validate** option displays no warnings.
9. Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
10. Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
11. Click **Deploy**.

Hsm

Select the **Hsm** tab of the **Configuration** page to configure the connection with the HSM (Hardware Security Module).

- [Vendor](#)
- [Token Label](#)
- [HSM PIN](#)
- [Number of sessions](#)


Mandatory: Yes.

JSON data type: Object.

Vendor

The vendor of the security module.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: Yes.

JSON data type: String ("none", "nshield" or "thales").

Token Label

The label of the HSM token that contains the private key for timestamp signing.

Mandatory: When the [Vendor](#) value is `nshield` or `thales`.

JSON data type: String. Remove this parameter from the JSON file if [Vendor](#) is `none`.

HSM PIN

The PIN (Personal Identification Number) of the HSM (Hardware Security Module).

Mandatory: When the [Vendor](#) value is `nshield` or `thales`.

JSON data type: The string identifier of a secret holding the PIN value. See the Entrust PKI Hub for how to set secret values with the following command.

```
clusterctl solution secret set
```

Remove this parameter from the JSON file if [Vendor](#) is `none`.

Number of sessions

The maximum number of concurrent PKCS #11 sessions on the HSM.

Mandatory: When the **Vendor** value is `nshield` or `thales`.

JSON data type: Integer. Remove this parameter from the JSON file if **Vendor** is `none`.

Tsa Server

Select the **Tsa server** tab of the **Configuration** page to configure the below timestamping server settings.

- [Read timeout](#)
- [Write timeout](#)
- [Idle timeout](#)
- [Max header bytes](#)
- [Max body bytes](#)
- [Graceful timeout](#)
- [Listen limit](#)
- [Keep alive](#)

Mandatory: No.

JSON data type: Object.

Read timeout

The maximum allowed period for reading an entire request, including the body. When this period expires, the request gets the following response.

```
Code=503,Reason=Service Unavailable
```

Mandatory: No. This optional value defaults to 60 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Write timeout

The maximum period allowed for writing a response. When this period expires, the request gets the following response.

```
Code=503,Reason=Service Unavailable
```

Mandatory: No. This optional value defaults to 60 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Idle timeout

The maximum period to wait for the next request when keep-alives are enabled.

Mandatory: No. This optional value defaults to 10 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Max header bytes

The maximum number of bytes allowed for keys and values in the request header, including the request line.

Mandatory: No. This optional value defaults to 1024.

JSON data type: Integer.

Max body bytes

The maximum number of bytes allowed in the request body.

Mandatory: No. This optional value defaults to 8192.

JSON data type: Integer.

Graceful timeout

The grace period before shutting down the server.

Mandatory: No. This optional value defaults to 15 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Listen limit


The maximum number of outstanding requests.

Mandatory: No. This optional value defaults to 0 (no limit).

JSON data type: Integer.

Keep alive

The TCP keep-alive timeouts on accepted connections.

 When this period expires, the server prunes dead TCP connections.

Mandatory: No. This optional value defaults to 3 minutes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Clock service

Select the **Clock server** tab of the **Configuration** page to configure the clock service settings.

- [Maximum allowed error](#)
- [Poll interval](#)
- [Connection timeout](#)

Mandatory: Yes.

JSON data type: Object.

Maximum allowed error

The maximum allowed time difference between the Timestamping Authority clock and the `chrony` client. When the difference exceeds the selected period, The Timestamping Authority solution:

1. Considers that the clock is not valid (bad clock).
2. Logs an error when trying to timestamp data.

See below for the main fields on the recorded error log.

Error log field	Value
level	info
msg	ProcessTimeStampRequest.Failed
tsa-log.tsa-pkistatus-string	TimeNotAvailableTSError
cause.cause.cause.msg	BadClock

Mandatory: Yes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Poll interval

The period between successive connections to the `crony` client for checking the clock health.

Mandatory: Yes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Connection timeout

The maximum period allowed for establishing a connection with the `chrony` client.

Mandatory: No. This optional value defaults to 1 second.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Tsa issuers

Select the **TSAs issuers** tab of the **Configuration** page to configure at least one timestamp issuer.

- [Issuer ID](#)
- [Log timestamp response](#)
- [TSA certificate](#)
- [CA chain](#)
- [TST profile](#)

Mandatory: Yes.

JSON data type: Object.

Issuer ID

A user-defined identifier for the configured timestamp issuer. The clients of this issuer will send requests to the following URL.

```
http://<host>/tsa/<issuerID>
```

Mandatory: Yes.

JSON data type: String.

Log timestamp response

Check this box to log the encoded timestamp response under the following tag.


```
tsa-timestampresponse-encoded
```

Mandatory: No. This optional value defaults to `false` (unchecked).

JSON data type: Boolean.

TSA certificate

The certificate described in [Generating a timestamping certificate and key pair](#). Click **Select Files** to import this certificate from file.

 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: Yes.

JSON data type: The string representation of a file path.

CA chain

Click **Select Files** to import the Certificate Authority certificates that will be included in the timestamp responses.

 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: Yes.

JSON data type: A list of `certFile` parameters, each one containing the String representation of a file path. For example:

```
"caCertChain": [  
  {  
    "certFile": "../data/tsa-rootCA-cert.pem"  
    "certFile": "../data/tsa-subCA-cert.pem"  
  }  
],
```

TST profile

Configure the timestamping policy for generating timestamp responses.

- [Accuracy](#)
- [Allowed hash algorithms](#)
- [Ordering](#)
- [Policy ID](#)
- [Qualified timestamp extension](#)
- [Serial number length](#)
- [Signature digest algorithm](#)

Mandatory: Yes.

JSON data type: Object.

Accuracy

The allowed deviation from the `genTime` generation time of the timestamp response.


Mandatory: No. When omitted, this optional value is not present in the `TSTInfo` response field.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Allowed hash algorithms

The list of allowed hash algorithms. Supported values are:

- sha1
- sha224
- sha256
- sha384
- sha512

 The Timestamping Authority solution will not timestamp data hashed with other algorithms.

Mandatory: No. This optional value defaults to the following list.

- sha224
- sha256
- sha384
- sha512

JSON data type: A list of algorithm identifiers – for example:

```
"allowedHashAlgs": ["sha224", "sha256", "sha384", "sha512"]
```

Ordering

Check this box to sort timestamps based on the `genTime` field, regardless of the accuracy of the `genTime` generation time in the timestamp response.

Mandatory: No. This optional value defaults to `false` (unchecked).

JSON data type: Boolean.

Policy ID

The identifier of the timestamping policy.

Mandatory: Yes.

JSON data type: String.

Qualified timestamp extension

Check this box to add a `qcStatements` statement with the `esi4-qtstStatement-1` extension as required for qualified electronic timestamps.

Mandatory: No. This optional value defaults to `false` (unchecked).

JSON data type: Boolean.

Serial number length

The allowed length in bytes of the timestamp serial numbers, as a number in the 8-20 range.

Mandatory: No. This optional value defaults to 8 bytes.

JSON data type: Integer.

Signature digest algorithm

The algorithm for hashing the timestamped data.

Mandatory: No. This optional value defaults to "sha256".

JSON data type: String.

Testing the timestamping service

To test the Timestamping Authority service, you can send timestamp requests as follows.

- [Creating the timestamp request](#)
- [Validating the timestamp request](#)
- [Sending the request to Timestamping Authority](#)
- [Parsing the timestamp response](#)
- [Verifying the response against the data](#)
- [Verifying the response against the request](#)

Creating the timestamp request

Create the timestamp request. For example:

```
openssl ts -query -data data.txt -sha256 -cert -out tsareq.tsq
```

See below for a description of each option.

Parameter	Description
-data <file>	Create a timestamp request for the data in the <code><file></code> file.
-sha256	Use the SHA256 algorithm to hash the data.
-cert	Add to the response the certificate described in Issuing a timestamping certificate .
-out <file>	Save the generated request in the <code><file></code> file.

The command saves the request in the `tsareq.tsq` file.

Validating the timestamp request

Validate the generated request.

```
openssl ts -query -in tsareq.tsq -text
```

Sending the request to Timestamping Authority

Send the timestamp request to Timestamping Authority. For example:

```
curl -H "Content-Type: application/timestamp-query" -H "Accept: application/timestamp-reply" --data-binary "@tsareq.tsq" http://<machine>/tsa/<issuerID> --output tsaresp.tsr
```

Where:

- `<machine>` is the IP address or domain name of the Entrust PKI Hub node hosting Timestamping Authority.
- `<issuerID>` is the value of the [Issuer ID](#) configuration parameter.

The command saves the response in the `tsaresp.tsr` file.

Parsing the timestamp response

Parse the timestamp response to validate the format.

```
openssl ts -reply -in tsaresp.tsr -text
```

Verifying the response against the data

Verify the timestamp response against the original data.

```
openssl ts -verify -in tsaresp.tsr -CAfile ca.pem -data data.txt
```

Where `ca.pem` is the Certificate Authority certificate for validating the certificate described in [Issuing a timestamping certificate](#).

Verifying the response against the request

Verify the response against the timestamp request.

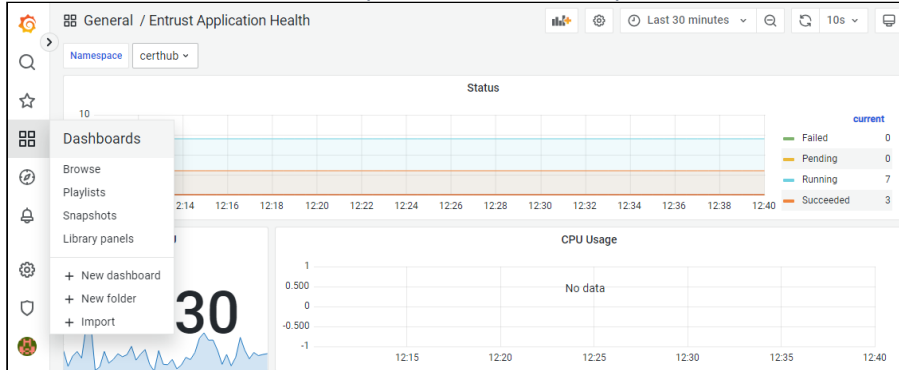
```
openssl ts -verify -in tsaresp.tsr -CAfile ca.pem -queryfile tsareq.tsq
```

Browsing Timestamping Authority logs

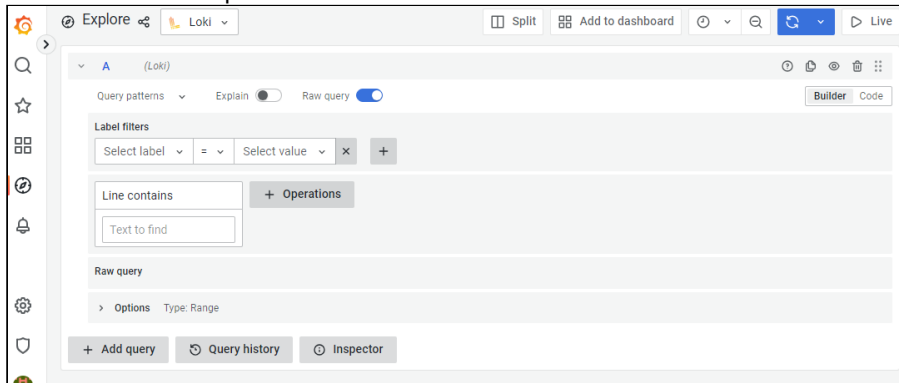
Entrust PKI Hub provides a Grafana web portal for browsing logs and metrics.

To browse Timestamping Authority logs with Grafana

1. Log into Grafana as explained in [Browsing logs with Grafana](#).
2. In the left sidebar of the Grafana portal, click the four squares icon and select **Dashboards**.



1. Select **Loki** in the top menu.



2. Select a time frame in the top menu.
3. Add the following log filters described in the below sections.
 - [Auditing the executed tsactl commands](#)
 - [Checking the clock status](#)
 - [Querying timestamping request logs](#)
4. Click **Run query**.

Auditing the executed tsactl commands

Add the following filters under **Label filters** to audit the execution of the `tsactl` command line tool.

Select label	Select value	Query output
filename	/var/log/entrust/tsa/tsactl.log	A record of all the executed <code>tsactl</code> commands

Checking the clock status

Add the following filters under **Label filters** to query clock status service logs.

Select label	Select value	Query output
namespace	tsa	Timestamping Authority logs.
app	clockstatus	Logs of the clock status service.

Use the **Line contains** field to filter time checking logs.

Line contains	Query output
"GetClockData.	Logs for time checking operations.

Querying timestamping request logs

Add the following filters under **Label filters** to query timestamping request logs.

Select label	Select value	Query output
namespace	tsa	Timestamping Authority logs.
app	tsa	Logs of the timestamping service.

Use the **Line contains** fields to add filters like the following.

Line contains	Query output
"ProcessTimeStampRequest.	Logs for all timestamping requests.
"ProcessTimeStampRequest.Failed"	Logs for failed timestamping requests.
"endpoint":"<issuerID>"	Logs for timestamping requests received by the <issuerID> timestamp issuer, where <issuerID> is the value of the Issuer ID configuration value.

tsactl reference

See the following sections for the commands supported by the `tsactl` command-line tool.

- [tsactl check clock](#)
- [tsactl check hsm](#)
- [tsactl create-csr](#)
- [tsactl create-key](#)
- [tsactl delete-key](#)

- [tsactl export-nshield](#)
- [tsactl import-nshield](#)
- [tsactl import-thales](#)
- [tsactl list-keys](#)
- [tsactl stop](#)

✘ The commands described in the following sections require passwordless `sudo` permissions.

tsactl check clock

Checks the host connection with the time source.

```
tsactl check clock
```

For example:

```
$ sudo ./tsactl check clock
Starting Pod...                               Done
Checking clock status...                       Done
```

tsactl check hsm

Checks the HSM connectivity.

```
tsactl check hsm [-l <level>] [-p <pin>] [-t <token>] [-v <vendor>]
```

For example:

```
$ sudo ./tsactl check hsm
Obtaining loaded secrets and configuration... Done
Starting PKCS #11 Manager...                   Done

Slot Id ->                                     0
Label ->                                       pking203
Serial Number ->                               1433959427612
Model ->                                       LunaSA 7.2.1
Firmware Version ->                           7.0.3
Configuration ->                               Luna User Partition With S0 (PED) Signing With
Cloning Mode
Slot Description ->                            Net Token Slot
FM HW Status ->                                FM Ready

Slot Id ->                                     1
Label ->                                       pking202
Serial Number ->                               1433964084224
Model ->                                       LunaSA 7.2.1
Firmware Version ->                           7.0.3
```

```
Configuration -> Luna User Partition With SO (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status -> FM Ready

Current Slot Id: 0


Passing HSM checks... Done
```

See below for a description of each option.

- `-l <level>`
- `-p <pin>`
- `-t <token>`
- `-v <vendor>`

`-l <level>`

Debug the nShield HSM with the `<level>` level, where `<level>` is a `CKNFAST_DEBUG` variable level. When not using an nShield HSM, the command ignores this option.

 See the nShield documentation for details on the `CKNFAST_DEBUG` configuration parameter.

Mandatory: No. This optional parameter defaults to 0.

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

`-t <token>`

Select the HSM token with the `<token>` label.

Mandatory: No. When omitting this option, the command uses the value of the `Token label` configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

`-v <vendor>`

Check an HSM of the `<vendor>` vendor, where `<vendor>` is either:

- `nshield`
- `thales`

Mandatory : When omitting this option, the command assumes the value of the `Vendor` configuration parameter and throws an error if not set.

tsactl create-csr

Generates a new certificate signing request (CSR) for a key pair previously generated with the [tsactl create-key](#) command.

```
tsactl create-csr -k <key_id> [-s <subject>] [-o <csr>] [-p <pin>] [-t <token>] [-v <vendor>] [-y]
```

For example:


```
$ sudo ./tsactl create-csr -k 7ce798c13a411bc1da4a9f983ed6d44fb4d7ed1a -s
"CN=97357462, O=Entrust, C=ES"
Obtaining loaded secrets and configuration... Done
Starting PKCS #11 Manager... Done
Using token with label mytoken
CSR:
----BEGIN CERTIFICATE REQUEST-----
MIIBIDCBxQIBADAzMTEwLWYDVQQDEyg3Y2U30ThjMTNhNDExYmMxZGE0YTlmOTgz
...
9xMajw==
----END CERTIFICATE REQUEST-----
```

See below for a description of each option.

- `-k <key_id>`
- `-s <subject>`
- `-o <csr>`
- `-p <pin>`
- `-t <token>`
- `-v <vendor>`
- `-y`

`-k <key_id>`


Select the key with the `<key_id>` identifier.

 Run the [evactl list-keys](#) command to get the key identifiers.

Mandatory: Yes.

`-s <subject>`

Use `<subject>` as the Subject of the certificate request. Where `<subject>` is a full Distinguished Name (DN) or Relative Distinguished Name (RDN).

 The DN attributes must be in capital letters for the Timestamping Authority solution to recognize the Subject.

For example:

```
CN=Example User,O=Example,C=US
```

```
CN=Example User
```

Mandatory: No. When omitting this option, the Subject in the generated certificate request defaults to the following:

```
CN=<key_id>
```

Where `<key_id>` is the key identifier.

`-o <csr>`

Save the certificate signing request (CSR) in a file with the `<csr>` path.

Mandatory: No. When omitting this option, the command prints the CSR to the standard output.

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

`-t <token>`

Select the HSM token with the `<token>` label.


Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

`-v <vendor>`

Use the `<vendor>` security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

✘ The command will raise an error if you omit this option and the configuration is not loaded.

-y

Skip the confirmation prompt.

tsactl create-key

Generates the key pair and the certificate signing request (CSR) of the certificate for signing TST responses.

```
tsactl create-key -k <key_type> [-s <subject>] [-o <csr>] [-p <pin>] [-t <token>] [-v <vendor>] [-y]
```

For example:

```
$ sudo ./tsactl create-key -k RSA2048 -s "CN=97357462, O=Entrust, C=ES"
Obtaining loaded secrets and configuration... Done
Starting PKCS #11 Manager... Done
Using token with label mytoken
Created key with id 4a00a4617d1afd5ad626955132dd0d396a69ed24
CSR:
-----BEGIN CERTIFICATE REQUEST-----
MIICqDCCAQAwMzExMC8GA1UEAxMoNGEwMGE0NjE3ZDFhZmQ1YWQ2MjY5NTUx
...
etTv+pac+nJKW8fw
-----END CERTIFICATE REQUEST-----
```

See below for a description of each option.

- -k <key_type>
- -s <subject>
- -o <csr>
- -p <pin>
- -t <token>
- -v <vendor>
- -y

-k <key_type>

Create a key of the <key_type> type, where <key_type> is one of the following.


- RSA2048
- RSA3072
- RSA4096
- ECDSAP256
- ECDSAP384
- ECDSAP521

Mandatory: Yes.

-s <subject>

Use <subject> as the Subject of the certificate request. Where <subject> is either:

- A full Distinguished Name (DN)
- A Relative Distinguished Name (RDN).

 The DN attributes must be in capital letters for the Subject to be recognized.

For example:

```
CN=Example User,O=Example,C=US
```

```
CN=Example User
```

Mandatory: No. When omitting this option, the Subject in the generated certificate request defaults to the following:

```
CN=<key_id>
```

Where <key_id> is the key identifier.

-o <csr>

Save the certificate signing request (CSR) in a file with the <csr> path.

Mandatory: No. When omitting this option, the command prints the CSR to the standard output.

-p <pin>


Authenticate in the HSM with the <pin> PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

-t <token>

Select the HSM token with the <token> label.


Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


-v <vendor>

Use the <vendor> security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

-y

Skip the confirmation prompt.

tsactl delete-key

Deletes a key.

```
tsactl delete-key -k <key-id> [-p <pin>] [-t <token>] [-v <vendor>] [-y]
```

For example:


```
$ sudo ./tsactl delete-key -k c403e0abae421c73625666dcff26dacf184eddd4 -y
Obtaining loaded secrets and configuration... Done
Starting PKCS #11 Manager... Done
Using token with label pking203
Deleted public key with id c403e0abae421c73625666dcff26dacf184eddd4
Deleted private key with id c403e0abae421c73625666dcff26dacf184eddd4
```

See below for a description of each option.

- -k <key_id>
- -p <pin>
- -t <token>
- -v <vendor>
- -y

-k <key_id>

Select the key with the `<key_id>` identifier.

 Run the `evactl list-keys` command to get the key identifiers.

Mandatory: Yes.

-p <pin>


Authenticate in the HSM with the <pin> PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

-t <token>

Select the HSM token with the <token> label.


Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


-v <vendor>

Use the <vendor> security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

-y

Skip the confirmation prompt.

tsactl export-nshield

Saves a copy of the nShield Security World keys and configuration currently loaded in TSA. You can later import it with [tsactl import-nshield](#), even in a different deployment of TSA.

```
tsactl export-nshield -o <output-directory> [-t]
```

For example:

```
$ sudo ./tsactl export-nshield -o /opt/nfast/copy-kmdata
```

See below for a description of each option.

- `-o <output-directory>`
- `-t`

`-o <output-directory>`

Save the configuration in the `<output-directory>` folder.

Mandatory: Yes.

`-t`

Save the configuration in the following compressed file.

```
<output-directory>/kmdata.tar.gz
```

Mandatory. No. When omitting this option, the command does not compress the configuration in the `<output-directory>` folder.

tsactl import-nshield

Imports the nShield Security World configuration so the Timestamping Authority solution can use the keys managed by the nShield HSM.

```
tsactl import-nshield -f <kmdata> [-y]
```

For example:

```
$ sudo ./tsactl import-nshield -f ./kmdata
```

See below for a description of each option.


- `-f <kmdata>`
- `-y, --yes`

Unable to render include or excerpt-include. Could not retrieve page.

`-f <kmdata>`

Import the `<kmdata>` configuration, where `<kmdata>` is one of the following.

- The path of the nShield `kmdata` folder.
- The path of a backup folder generated with the `tsactl export-nshield` command.
- The path of a `tar.gz` backup file generated with the `tsactl export-nshield` command.

 See [Loading the HSM configuration](#) for considerations on this configuration.

Mandatory: Yes.

-y, --yes

Skip the confirmation prompt.

tsactl import-thales

Imports the configuration of a Thales HSM. Use the following syntax to import this configuration from a ZIP file.

```
tsactl import-thales -d <package_path> [-y]
```

Use the following syntax to import this configuration from a Chrystoki file.

```
tsactl import-thales -c <cert_dir> -k <chrystoki> [-y]
```

For example:

```
$ sudo ./tsactl import-thales -c ./tsa-thales-config/cert -k ./tsa-thales-config/Chrystoki.conf -y
Saving Thales configuration... Done
Warning: tsa is already deployed! To apply the changes, tsa needs to be redeployed using the tsactl deploy command.
```

See below for a description of each option.

- -c <cert_dir>
- -d <package_path>
- -k <chrystoki>
- -y

⚠ Changes will be effective when deploying (or redeploying) the solution with the Management Console or the `clusterctl deploy` command.

-c <cert_dir>

Import the client and server certificates for the Luna Network or DPoD authentication. Where `<cert_dir>` is the path of a `cert` directory with the following contents.


```
├── cert
│   ├── client
│   │   ├── <clientKey>.pem
│   │   └── <clientCert>.pem
│   └── server
│       └── <caCert>.pem
```

See below for a description of each field.

Value	Description
<clientKey>	The file name of a PEM file containing the client's private key.
<clientCert>	The file name of a PEM file containing the client's certificate.
<caCert>	The file name of a PEM file containing the CA certificate for validating the server's certificate.

After running the command, verify the `Chrystoki.conf` file includes the following configuration.

```
ClientPrivKeyFile = /usr/safenet/lunaclient/cert/client/<clientKey>.pem;
ClientCertFile = /usr/safenet/lunaclient/cert/client/<clientCert>.pem;
ServerCAFile = /usr/safenet/lunaclient/cert/server/<caCert>.pem;
```

 Do not modify any other path in the `Chrystoki.conf` file.

Mandatory: Yes.

`-d <package_path>`

Use the `<package_path>` DPoD configuration package, where `<package_path>` is the path of the ZIP package file.

Mandatory: Yes.

`-k <chrystoki>`

Import the `<chrystoki>` configuration of the Luna Network or DPoD client, where `<chrystoki>` is the path of the `Chrystoky.conf` file.

Mandatory: Yes.

`-y`

Skip the confirmation prompt.

tsactl list-keys

Lists the keys in the PKCS #11 token.

```
tsactl list-keys [-p <pin>] [-t <token>] [-v <vendor>]
```

For example:

```
$ sudo ./tsactl list-keys
Obtaining loaded secrets and configuration... Done
Starting PKCS #11 Manager... Done
```

```
Using token with label pking203
Public Key Object; RSA 2048 bits
Label: 305ecd78340acc3d906be370a01e7884
ID: 03b1dac1e383b8d3adea5a6a2c6200bde58ffb40
Usage: verify

Private Key Object; RSA 2048 bits
Label: F
ID: 0f
Usage: sign, unwrap

Public Key Object; RSA 2048 bits
Label: F
ID: 0f
Usage: verify, wrap

Private Key Object; RSA 2048 bits
Label: webserver-root1
ID: 103d6c94ea10b98ab37186cc1c4977eb
Usage: sign
```

See below for a description of each option.

- `-p <pin>`
- `-t <token>`
- `-v <vendor>`

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

`-t <token>`

Select the HSM token with the `<token>` label.

Mandatory: No. When omitting this option, the command uses the value of the `Token label` configuration parameter.


 The command will raise an error if you omit this option and the configuration is not loaded.

`-v <vendor>`


Use the `<vendor>` security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.

Vendor	Security module
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


tsactl stop

Stops a deployed Timestamping Authority solution.

```
tsactl stop
```

For example:

```
$ sudo ./tsactl stop
Stopping Virtual Services... Done
Stopping Services... Done
Stopping Deployments... Done
Stopping Stateful Sets... Done
Stopping Pods... Done
```

 To restart Timestamping Authority, redeploy the solution as explained in [Configuring and deploying Timestamping Authority](#).

Troubleshooting Timestamping Authority

See below for how to troubleshoot the main issues.

- [Connectivity issues](#)
- [Error: Another instance of tsactl is running](#)

Connectivity issues

As explained in [Timestamping Authority overview](#), the Timestamping Authority connects with:

- The system clock service (which is provided by the `chrony` connection to an NTP server).
- An HSM

To check the connection with these components, run the `tsactl check clock` and `tsactl check hsm` commands.

Error: Another instance of tsactl is running

You can encounter the following error when creating or deleting a key.

```
Error: Another instance of tsactl is running create-key or delete-key
```

In that case:

1. Ensure there is no other instance of `tsactl` performing any of those operations.
2. Re-run the command with the `FORCE_MUTEX_OPERATION` environment variable set to 1. For example:

```
sudo FORCE_MUTEX_OPERATION=1 ./tsactl create-key -k RSA2048
```

 Running a command with `FORCE_MUTEX_OPERATION` set to 1 can override the changes made by another `tsactl` running instance.

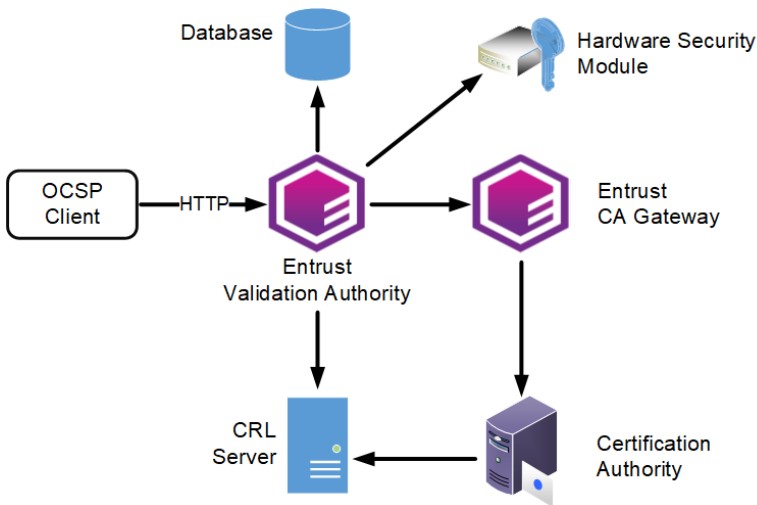
Starting up Entrust Validation Authority

See below for starting up the Entrust Validation Authority solution.

- [Entrust Validation Authority overview](#)
- [Loading the HSM configuration on Entrust Validation Authority](#)
- [Initializing the Entrust Validation Authority database](#)
- [Configuring a certificate information source for Entrust Validation Authority](#)
- [Verifying port access for Entrust Validation Authority](#)
- [Generating a VA certificate and key pair](#)
- [Configuring Entrust Authority Security Manager for Entrust Validation Authority](#)
- [Configuring and deploying Entrust Validation Authority](#)
- [Testing the OCSP Responder](#)
- [Browsing Entrust Validation Authority logs](#)
- [evactl reference](#)
- [Troubleshooting Entrust Validation Authority](#)

Entrust Validation Authority overview

The Entrust Validation Authority (EVA) solution responds to OCSP requests on the validation status of the certificates issued by a Certificate Authority. To respond to these requests, the Entrust Validation Authority solution connects with different components.



See below for a description of each component.

- [OCSP client](#)
- [Certificate Authority](#)
- [Certificate information source](#)
- [Hardware Security Module](#)
- [Database](#).

OCSP client

Multiple clients send OCSP requests to the OCSP Responder service of the Entrust Validation Authority solution.

Certificate Authority

The Entrust Validation Authority solution checks the status of certificates issued by one or multiple Certificate Authorities (CAs).

i As explained in [Starting up Certificate Authorities](#), the Certificate Authorities solution provides built-in CRL and OCSP services. Therefore, you do not need an Entrust Validation Authority for CAs created with the Certificate Authorities solution.


Certificate information source

Through Entrust CA Gateway, Entrust solutions obtain a direct feed of issued certificates from each supported Certificate Authority (CA). See the following table for the CA Gateway deployment required by each type of CA.

CA type	CA Gateway deployment
Certificate Authority running on PKI Hub	Create a Certificate Authority instance, as explained in Starting up Certificate Authorities , and select the built-in CA Gateway service of this CA.

CA type	CA Gateway deployment
External Certificate Authority	Start up the Entrust CA Gateway solution and connect it with the external CA as explained Starting up CA Gateway .

Alternatively, the Entrust Validation Authority solution can obtain revocation information from a full or "combined" CRL published in an LDAP or HTTP server.

 Entrust Validation Authority does not support partitioned CRLs.

Hardware Security Module

A Hardware Security Module (HSM) manages one or several OCSP signing keys.

Database.

A database stores the status of the certificates.


Loading the HSM configuration on Entrust Validation Authority

See the following table for the required command for loading the configuration of the Hardware Security Module (HSM).

HSM	Command
Entrust nShield HSM	<code>evactl import-nshield</code>
Thales HSM	<code>evactl import-thales</code>

See also:

- [HSM requirements](#) for the list of supported HSMs.
- [Configuring an nShield HSM](#) for the additional steps required by Entrust nShield HSMs.

 Skip this step if you use software cryptography (not recommended).

Initializing the Entrust Validation Authority database

Initialize the Entrust Validation Authority external database as explained in the following sections.

- [Database Management System requirements for Entrust Validation Authority](#)
- [Downloading the Entrust Validation Authority database scripts](#)
- [Setting the variables of the Entrust Validation Authority database scripts](#)
- [Running the Entrust Validation Authority database scripts](#)

Database Management System requirements for Entrust Validation Authority

Entrust Validation Authority is tested with the following Database Management System (DBMSs).

DBMS	Version
PostgreSQL	14.3 or 15.6
Oracle	21.3.0
SQL Server	2019 CU15

Downloading the Entrust Validation Authority database scripts

See below for instructions on downloading the Entrust Validation Authority database scripts.

To download the Entrust Validation Authority database scripts

1. Log in to the secure trustedcare.entrust.com portal with your customer credentials.
2. Select the **PRODUCTS** tab.
3. Click **PKI Hub**.
4. Select the product version.
5. In the **SOFTWARE DOWNLOADS** tab, download the compressed file containing the database scripts.
6. Extract the contents of the compressed file. In Linux operating system, run:

```
tar -xvf eva-database-scripts.tar.gz
```

Setting the variables of the Entrust Validation Authority database scripts

To run the database scripts, you will need to provide the following values, either in the execution command line or as environment variables.

Variable	Value
DBNAME	The database name.
HOSTNAME	The name of the host to connect to.
USERNAME	The name of a DBMS user with permission to create tables, create users, and grant user permissions.
PASSWORD	The password of the database user.
OCSRESPONDER_DB_PASSWORD	The password of the OCS Responder user with Read permissions on the certStatus and metadata tables.
OCSRESPONDER_DB_USER	The name of the OCS Responder user with Read permissions on the certStatus and metadata tables.

Variable	Value
STATUSFEEDER_DB_PASSWORD	The password of the Status Feeder user with Read and Write permissions on the certStatus and metadata tables.
STATUSFEEDER_DB_USER	The name of the Status Feeder user with Read and Write permissions on the certStatus and metadata tables.

Running the Entrust Validation Authority database scripts

See below for creating the Entrust Validation Authority database in the DBMS of your choice.

- [Creating the database on Oracle](#)
- [Creating the database on PostgreSQL](#)
- [Creating the database on SQL Server](#)

i The syntax of the below commands assumes a Linux operating system. Running these commands on a Windows machine may require a different syntax – for example, evaluating the `<var>` variables with the `%<var>%` syntax.

Creating the database on Oracle

Run the `certstatus_initial_schema.sql` and `metadata_initial_schema.sql` scripts to create the database objects on Oracle.

```
sqlplus "$USERNAME/$PASSWORD@tcp://$HOSTNAME/$DBNAME" @./  
certstatus_initial_schema.sql  
sqlplus "$USERNAME/$PASSWORD@tcp://$HOSTNAME/$DBNAME" @./metadata_initial_schema.sql
```

Run the `create_users.sql` script to create the database users on Oracle.

```
sqlplus "$USERNAME/$PASSWORD@tcp://$HOSTNAME/$DBNAME" @./create_users.sql $USERNAME  
$STATUSFEEDER_DB_USER $STATUSFEEDER_DB_PASSWORD $OCSPRESPONDER_DB_USER  
$OCSPRESPONDER_DB_PASSWORD
```

Creating the database on PostgreSQL

Run the `certstatus_initial_schema.sql` and `metadata_initial_schema.sql` scripts to create the database objects on PostgreSQL.

```
PGPASSWORD=$PASSWORD psql -d $DBNAME -U $USERNAME -h $HOSTNAME -v "ON_ERROR_STOP=1"  
-f ./certstatus_initial_schema.sql  
PGPASSWORD=$PASSWORD psql -d $DBNAME -U $USERNAME -h $HOSTNAME -v "ON_ERROR_STOP=1"  
-f ./metadata_initial_schema.sql
```

Run the `create_users.sql` script to create the database users on PostgreSQL.

```
PGPASSWORD=$PASSWORD psql -d $DBNAME -U $USERNAME -h $HOSTNAME \  
-v STATUSFEEDER_DB_USER=$STATUSFEEDER_DB_USER \  
-v OCSPRESPONDER_DB_USER=$OCSPRESPONDER_DB_USER \  
-v STATUSFEEDER_DB_PASSWORD=$STATUSFEEDER_DB_PASSWORD \  
-v OCSPRESPONDER_DB_PASSWORD=$OCSPRESPONDER_DB_PASSWORD \  
-v "ON_ERROR_STOP=1" -f ./create_users.sql
```

Creating the database on SQL Server

Run the `certstatus_initial_schema.sql` and `metadata_initial_schema.sql` scripts to create the database objects on SQL Server.

```
sqlcmd -S "$HOSTNAME" -U "$USERNAME" -P $PASSWORD -v DBNAME="$DBNAME" -i ./  
certstatus_initial_schema.sql  
sqlcmd -S "$HOSTNAME" -U "$USERNAME" -P $PASSWORD -v DBNAME="$DBNAME" -i ./  
metadata_initial_schema.sql
```

Run the `create_users.sql` script to create the database users on SQL Server.

```
sqlcmd \  
-S "$HOSTNAME" -U "$USERNAME" -P $PASSWORD \  
-v STATUSFEEDER_DB_USER="$STATUSFEEDER_DB_USER" \  
-v OCSPRESPONDER_DB_USER="$OCSPRESPONDER_DB_USER" \  
-v STATUSFEEDER_DB_PASSWORD="$STATUSFEEDER_DB_PASSWORD" \  
-v OCSPRESPONDER_DB_PASSWORD="$OCSPRESPONDER_DB_PASSWORD" \  
-v DBNAME="$DBNAME" \  
-i ./create_users.sql
```

Configuring a certificate information source for Entrust Validation Authority

You need one of the following sources of information on the certificate validity status.

- [Certificate Revocation List](#)
- [CA Gateway for Entrust Validation Authority](#)

i When [Configuring and deploying Entrust Validation Authority](#), the certificate information source is selected using the `Certificates Source` parameter.

Certificate Revocation List

Entrust Validation Authority supports obtaining certificate validity status information from a CRL (Certificate Revocation List) with the following configuration.

CRL setting	Supported value
CRL Encoding	DER
CRL host	HTTP or LDAP server
CRL type	Full or "combined" CRL (because Entrust Validation Authority does not support partitioned CRLs).
CRL issuer	A CA of any manufacturer.

CA Gateway for Entrust Validation Authority


When using CA Gateway as the source of certificate information, perform the steps below for Entrust Validation Authority to authenticate on CA Gateway.

- [Generating the CA Gateway client certificate](#)
- [Configuring the client certificate in CA Gateway](#)
- [Importing the CA Gateway client certificate](#)

Generating the CA Gateway client certificate


Use your CA to generate a PKCS #12 containing:

- A TLS client certificate for Entrust Validation Authority to authenticate on CA Gateway.
- The private key of the certificate.

 The PKCS #12 cannot contain more than one client certificate.


To generate the PKCS #12 with Entrust Security Manager:

- Select the **1-Key-Pair User (1-Key-Pair User with Dual Usage Key)** certificate type to generate a PKCS #12 with a single client certificate.
- Check the **Export PKCS #12** and **All exportable** options so the user can export the generated PKCS #12.

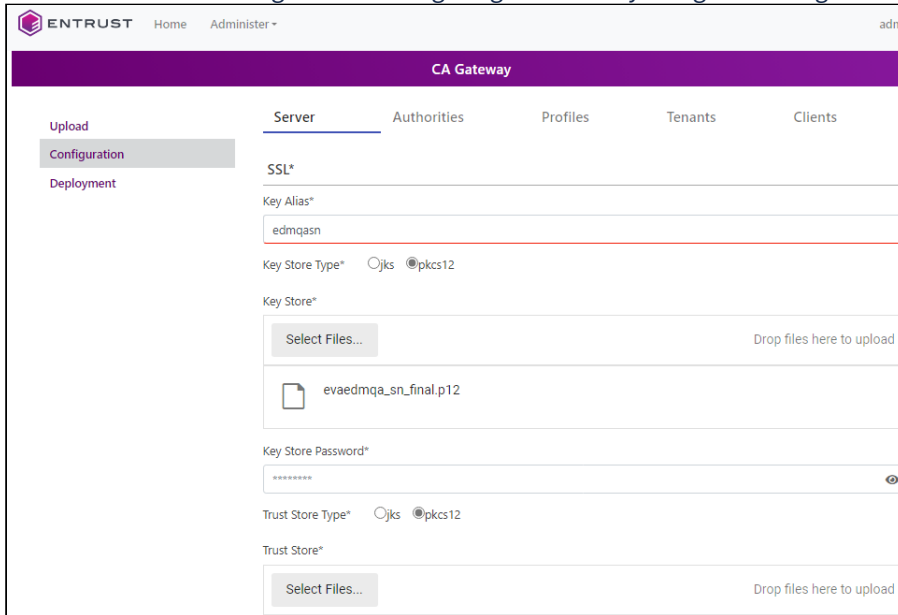
 See the Entrust Security Manager documentation for more detailed information.

Configuring the client certificate in CA Gateway

Configure the Entrust Validation Authority client certificate in the CA Gateway settings.

 Redeploy CA Gateway to make changes effective.

See the Entrust PKI Hub guide for configuring CA Gateway using the Management Console.



Alternatively, you can set the configuration using the `clusterctl` command-line tool.

1. Export the CA Gateway initial configuration with the `clusterctl solution config export` command described in the Entrust PKI Hub guide.
2. Perform the required updates in the `application.yml` configuration file.
3. Update the solution configuration with the `clusterctl solution config import` command described in the Entrust PKI Hub guide.

See the following sections for the required parameters.

- [Trust Store \(trust-store\)](#)
- [Subject DN \(subject-dn\)](#)

Trust Store (trust-store)

A PKCS #12 file containing:

- The CA certificates already included in the previous Trust Store (if any).
- The certificate of the CA that issued the certificate described in [Generating the CA Gateway client certificate](#).

When using the Management Console:

1. Select the **Server** tab.
2. click **Select Files** under the **Trust Store** field and import the file.

When using instead the `application.yml` configuration file, assign the path of this file to the following parameter.

```
server.ssl.trust-store
```

Subject DN (subject-dn)

The distinguished name (DN) of the certificate described in [Generating the CA Gateway client certificate](#). When using the Management Console:

1. Click the **Clients** tab.
2. Enter the DN in the **Subject DN** field.

When using instead the `application.yml` configuration file, assign this DN to the following parameter.

```
server.clients.subject-dn
```

Importing the CA Gateway client certificate

When using CA Gateway as the source of certificate information, run the `evactl import-p12` command to import the PKCS #12 described in [Generating the CA Gateway client certificate](#). For example:

```
$ sudo ./evactl import-p12 -f eva-cagw.p12
```

Verifying port access for Entrust Validation Authority

In addition to the ports listed in [Required open ports](#), ensure no network restriction blocks access to the following ports.

- [Incoming traffic](#)
- [Outgoing traffic](#)

i Entrust Validation Authority deployment automatically opens these ports in the firewall of the machines hosting Entrust PKI Hub.

Incoming traffic

The Entrust Validation Authority deployment automatically opens the following ports for incoming traffic in the firewall of the host machines.


Target Port	Protocol	Source	Target Service
80	TCP/HTTP	OCSP client	OCSP Responder

Outgoing traffic

The Entrust Validation Authority deployment automatically opens the following ports for outgoing traffic in the firewall of the host machines.

Target Port	Protocol	Source	Target Service
80	HTTP	CRL shim	HTTP Server

Target Port	Protocol	Source	Target Service
389	LDAP	CRL shim	LDAP Server
8444	TCP/HTTPS	CA Gateway shim	CA Gateway
1433	TCP/HTTPS	Status Feeder and OCSP Responder	SQL Server Database
1792	NTLS	OCSP Responder	Luna Network HSM
1521	TCP/HTTPS	Status Feeder and OCSP Responder	Oracle Database
5432	TCP/HTTPS	Status Feeder and OCSP Responder	PostgreSQL Database
9000-9004	TCP/HTTPS	OCSP Responder	nShield HSM

 You can modify these default ports in the configuration settings of the target services.

Generating a VA certificate and key pair

Each CA configured in Entrust Validation Authority needs a certificate to sign OCSP responses. You can:

- Use a different certificate for every CA.
- Share a certificate among multiple CAs.


Perform the steps below for every certificate you want to use.

- [Generating a VA key pair](#)
- [Issuing a VA certificate](#)

Generating a VA key pair

To generate a VA key pair, run the `evactl create-key` command in any Entrust PKI Hub node. The command will output a CSR that you can use to generate the VA certificate – for example:

```
$ sudo evactl create-key -k RSA2048 -s "CN=OCSP Server" -o /tmp/certreq.txt -t
mytoken -v thales
Created key with id 4a00a4617d1afd5ad626955132dd0d396a69ed24
CSR:
-----BEGIN CERTIFICATE REQUEST-----
MIICqDCCAQAQAwMzExMC8GA1UEAxMoNGEwMGE0NjE3ZDFhZmQ1YWQ2MjY5NTUx
...
etTv+pac+nJKW8fw
-----END CERTIFICATE REQUEST-----
```

 As explained in [evactl create-csr](#), you can create a certificate request for a key that already exists on the HSM.

Issuing a VA certificate

Use your CA to process the CSR obtained in [Generating a VA key pair](#) and issue a certificate with the following extension values.

Extension	Value
Key Usage	digitalSignature
Extended Key Usage	OCSPSigning


See below for how to issue this certificate with Entrust products.

- [Issuing an OCSP responder VA certificate Entrust Security Manager](#)
- [Issuing an OCSP responder VA certificate with the Certificate Authorities solution](#)

Issuing an OCSP responder VA certificate Entrust Security Manager

If you are using Entrust Authority Security Manager, you can run one of the following applications to issue the VA certificate:

- The CSR Enrollment Services (CSRES) provided by the Entrust Administration Services.
- The Profile Creation Utility included in products such as Entrust Administration Services or available as a separate download with Entrust CA Gateway.

 See [Adding the OCSP Server certificate type to Entrust Authority Security Manager](#) for how to configure Entrust Authority Security Manager.

Issuing an OCSP responder VA certificate with the Certificate Authorities solution

Follow the steps below to issue an OCSP responder VA certificate using the Certificate Authority solution provided by PKI Hub.

- [Creating a Certificate Authority to issue OCSP responder VA certificates](#)
- [Creating a timestamping certificate request](#)
- [Processing the timestamping certificate request](#)

⚠ The Certificate Authorities solution provides built-in CRL and OCSP capabilities. Therefore, you do not need the Entrust Validation Authority to check the validity status of certificates issued by CA instances managed by the Certificate Authorities solution.

Creating a Certificate Authority to issue OCSP responder VA certificates

Follow the steps described in [Starting up Certificate Authorities](#) to create:

- A root Certificate Authority.
- An issuing Certificate Authority with at least one signature profile – for example, the `wstep-digital-signature` profile described in [Active Directory \(WSTEP\) certificate profiles](#).

Creating a timestamping certificate request

Create a CA Gateway certificate enrollment request – for example:

```

1  {
2    "csr": "MIIDVzCCAb8...",
3    "profileId": "wstep-digital-signature",
4    "requiredFormat": {
5      "format": "X509"
6    },
7    "optionalCertificateRequestDetails": {
8      "extensions": [
9        {
10       "oid": "2.5.29.37",
11       "critical": true,
12       "value": "MAoGCCsGAQUFBwMJ"
13     }
14   ],
15   "validity_period": "2024-11-06T13:00Z/2026-07-06T13:00:00Z"
16 }
17 }
```

See below for the values required by each request field.

Field	Line	Value
csr	2	The base64-encoded Certificate Signing Request previously generated in Generating a VA key pair as a single line.
extensions	8	The same fields and values as in the above example (to select the OCSP signing extended key usage).
validity_period	15	The validity period for the issued certificate. The expiry date in this period cannot exceed the expiry date of the issuing CA certificate.

Processing the timestamping certificate request

See below for processing the timestamping certificate request and obtaining the issued certificate.

To process the timestamping certificate request

1. Use a REST client to process the request as explained in [Issuing certificates with a REST client](#).
2. Edit the REST response and copy the base64-encoded string in the `body` field.
3. Save the string in a file with the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` delimiters – for example:

```
-----BEGIN CERTIFICATE-----
MIIIIcDCCBljgAwIBAgITMwEf/Fvr7NDwanyeRAAAAR/8WzANBgkqhkiG9w0BAQwF
ADBdMQswCQYDVQQGEwJVUzEeMBwGA1UEChMVTWljcm9zb2Z0IENvcnBvcmlF0aW9u
...
Ud085g==
-----END CERTIFICATE-----
```

4. Use this certificate file as the [VA certificate](#) when configuring the solution.

Configuring Entrust Authority Security Manager for Entrust Validation Authority

Follow the steps below if Entrust Validation Authority will obtain certificate information from an Entrust CA Gateway instance integrated with Entrust Authority Security Manager.


- [Configuring the CA Gateway administrator role in Entrust Authority Security Manager](#)
- [Adding the OCSF Server certificate type to Entrust Authority Security Manager](#)

Configuring the CA Gateway administrator role in Entrust Authority Security Manager

When configuring a role for the Entrust CA Gateway administrator in Entrust Authority Security Manager, make sure the role has permissions to administer:


- All roles
- All certificate types

See the Entrust CA Gateway deployment guide for creating and configuring the CA Gateway administrator role in Entrust Authority Security Manager.

 If the Entrust CA Gateway administrator role does not have enough permission, Entrust Validation Authority will only receive information on a subset of the issued certificates.

Adding the OCSF Server certificate type to Entrust Authority Security Manager

If not already added, add the OCSF Server (OCSP_1K) certificate type you will later use for [Generating a VA certificate and key pair](#).

 In the latest Entrust Authority Security Manager 10.0.x releases, an OCSF Server (OCSP_1K) certificate type may already be predefined in the certificate specifications. This certificate type includes the proper certificate extensions for signing OCSF responses.

To add the OCSF Server certificate type to Entrust Authority Security Manager

1. Log in to Entrust Authority Security Manager Administration.
2. Select **File > Certificate Specifications > Export** and export the certificate specifications.
3. Open the certificate specifications file in a text editor.
4. Add the following lines to the `[Certificate Types]` section.

```
OCSP_1K=enterprise,OCSP server,OCSP server certificate -no directory entry
```

5. Add the following lines to the `[Extension Definitions]` section.

```
-----  
;- Cert Type: OCSP_1K  
;- This cert type needs to be mapped to cert def policy enforcing:  
; - Certificate lifetime:  
; - Exclude privateKeyUsagePeriod: 1  
; - Exclude basicConstraints: 1  
; - Exclude entrustVersInfo: 1  
; - Exclude CDP: 1  
-----  
[OCSP_1K Certificate Definitions]  
1=Verification  
;  
[OCSP_1K Verification Extensions]  
;Key Usage: Digital Signature  
keyusage=2.5.29.15,n,m,BitString,1  
;Extended Key Usage: OCSP Signing  
extkeyusage=2.5.29.37,n,o,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.9  
ocspnocheck=1.3.6.1.5.5.7.48.1.5,n,o,DER,0500  
; Certificate Policies: DER encode the <Policy-OID>  
; Policy-OID=<Policy-OID> - This OID is optional, the customer might not have a  
policy OID.  
;certificatepolicies=2.5.29.32,n,o,DER,<DER encoded value of the above OID>  
; AuthorityInfo Access:  
; - Issuing CA certificate URL: <CA-Cert-HTTP-URL>  
;aia=1.3.6.1.5.5.7.1.1,n,m,DER,<DER encoded value of the above URL>  
;
```

6. (Optional.) You can add a `certificatePolicies` extension to the certificate type. The `certificatePolicies` extension contains policy information, such as how your CA operates and the intended purpose of the issued certificate. Typically, different certificate policies will relate to different applications which may use the certified key. The Certificate Policies extension contains a sequence of one or more policy information terms. Each policy information term consists of an object identifier (OID) and optional qualifiers. In an end entity certificate, the policy information terms indicate the policy under which the certificate has been issued, and the purposes for which the certificate may be used. To add a `certificatePolicies` extension to the certificate type:
 - a. DER-encode a list of one or more policy OIDs. Entrust provides an `entDerEncoder` utility for Security Manager that you can use to DER-encode data for certificate extensions. For instructions about using the `entDerEncoder` utility, see the Security Manager documentation.
 - b. Uncomment the `certificatepolicies=` entry and replace `<DER encoded value of the above OID>` with the DER-encoded value you obtained in the previous step.


7. (Optional.) You can add an `authorityInformationAccess` extension to the certificate type. The Authority Information Access (AIA) certificate extension indicates how to access information and services for the CA that issued the certificate. Information and services may include online validation services and CA policy data. To add a `certificatePolicies` extension to the certificate type:
 - a. DER-encode the HTTP URL of the CA certificate. Entrust provides an `entDerEncoder` utility for Security Manager that you can use to DER-encode data for certificate extensions. For instructions about using the `entDerEncoder` utility, see the Security Manager documentation.
 - b. Uncomment the `aia=` entry and replace `<DER encoded value of the above URL>` with the DER-encoded value you obtained in the previous step.
8. Add the following lines to the `[Advanced Settings]` section.

```
[OCSP_1K Advanced]
noBasicConstraints=1
noPrivateKeyUsage=1
noEntrustVersInfo=1
;cdpLdapDnLast=1
noUserInDirectory=1
noCRLDistPoints=1
```

9. Save and close the file.
10. Select **File > Certificate Specifications > Import** and import the certificate specifications back into Entrust Authority Security Manager.
11. In the tree view, select **Security Policy > Certificate Categories > Enterprise > Certificate Types > OCSP Responder (OCSP Responder Certificates) > Verification**.
12. Click the **Certificate definition Policy** field, and then select **Verification_p10 Policy** from the drop-down list.

Configuring and deploying Entrust Validation Authority

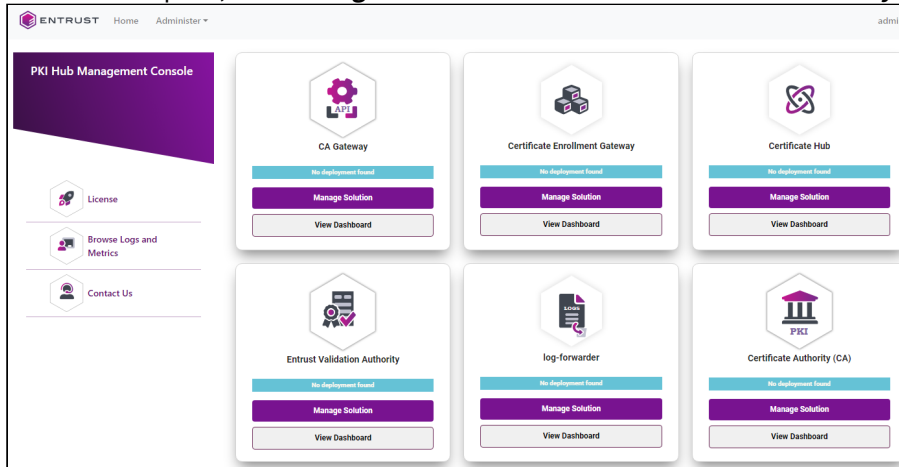
See below for configuring and deploying Entrust Validation Authority with the Management Console.

 Repeat the following steps each time a configuration update is required. Do not forget to click **Deploy** to make the changes effective.

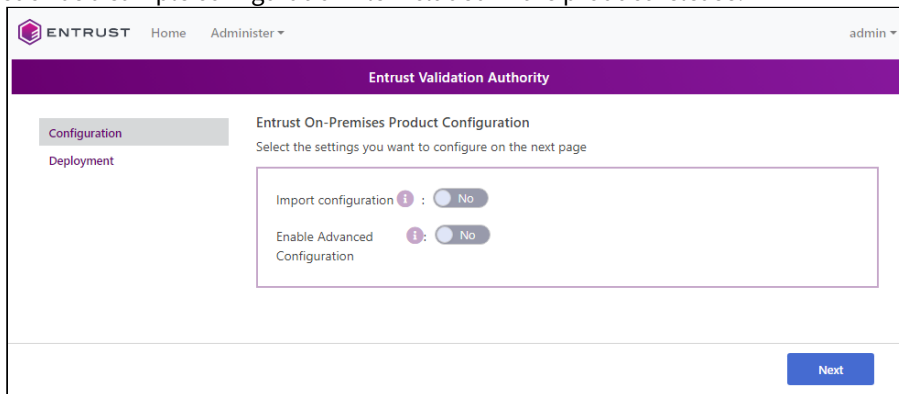
To configure and deploy Entrust Validation Authority with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).

- In the content pane, click **Manage Solution** under **Entrust Validation Authority**.



- Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.



- Active the **Enable Advanced Configuration** if you want to configure the full set of configuration parameters supported by the solution.
- Click **Next**.
- Configure the solution settings described in the following sections.
 - [Database](#)
 - [Hsm](#)
 - [OCSP Responder-Server](#)
 - [LDAP Servers](#)
 - [Certificate Authorities](#)
- Click **Validate** to validate the configured settings.
- Correct any detected configuration error until the **Validate** option displays no warnings.
- Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
- Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
- Click **Deploy**.

Database

Select the **Database** tab of the **Configuration** page to configure the database connection

- [Connection timeout](#)
- [Database name](#)

- Driver
- Host
- JDBC URL
- Max connections
- OCSF Responder password
- OCSF Responder User
- Port
- SSL mode
- SSL validation certificate
- Status Feeder password
- Status Feeder User

Mandatory: Yes

JSON data type: Object.

Connection timeout

The timeout for database connections.

Mandatory: No. This optional value defaults to 5 seconds.


JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Database name

The value assigned to the `DBNAME` parameter when [Initializing the Entrust Validation Authority database](#).

Mandatory: database connection settings support the following combinations.

Driver	Database name	Host	Port	JDBC URL
oracle	✓	✓	✓	✗
oracle	✗	✗	✗	✓
postgres	✓	✓	✓	✗
sqlserver	✓	✓	✓	✗

 Remove any unnecessary parameters.

JSON data type: String

Driver

The driver for connecting to the database.

Driver	Database Management System
oracle	Oracle SQL

Driver	Database Management System
postgres	PostgreSQL
sqlserver	Microsoft SQL Server

Mandatory: Yes.


JSON data type: String.

Host

The IP address or hostname of the database host.

Mandatory: database connection settings support the following combinations.

Driver	Database name	Host	Port	JDBC URL
oracle	✓	✓	✓	✗
oracle	✗	✗	✗	✓
postgres	✓	✓	✓	✗
sqlserver	✓	✓	✓	✗

 Remove any unnecessary parameters.

JSON data type: String.

JDBC URL

The JDBC URL to connect to the database when **Driver** is `oracle`. For example, to connect with a database hosted in multiple Oracle nodes:


```
(DESCRIPTION=(ADDRESS_LIST=(LOAD_BALANCE=yes) (FAILOVER=on) (ADDRESS=(PROTOCOL=TCP) (HOST=host1.domain.com) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP) (HOST=host2.domain.com) (PORT=1521))) (CONNECT_DATA=(SERVICE_NAME=ServiceName)))
```

Do not include the following prefix in the parameter value:

```
jdbc:oracle:thin:@
```

Mandatory: database connection settings support the following combinations.


Driver	Database name	Host	Port	JDBC URL
oracle	✓	✓	✓	✗
oracle	✗	✗	✗	✓
postgres	✓	✓	✓	✗
sqlserver	✓	✓	✓	✗

 Remove any unnecessary parameters.

JSON data type: String.

Max connections

The number of maximum concurrent database connections. as an integer equal to or greater than 1.

 Since three internal services of Entrust Validation Authority utilize this value, the database must support at least three times the maximum concurrent connections set by this parameter.

Mandatory: Yes.

JSON data type: Integer.

OCSF Responder password

The password assigned to the `OCSFPRESPONDER_DB_PASSWORD` parameter when [Initializing the Entrust Validation Authority database](#).

Mandatory: Yes.

JSON data type: The string identifier of a secret holding the PIN value.

 Run the `clusterctl solution secret set` command to set secret values manually.

OCSF Responder User

The value assigned to the `OCSFPRESPONDER_DB_USER` parameter when [Initializing the Entrust Validation Authority database](#).

Mandatory: Yes.


JSON data type: String.

Port

The TCP port where the database listens.

Mandatory: database connection settings support the following combinations.

Driver	Database name	Host	Port	JDBC URL
oracle	✓	✓	✓	✗
oracle	✗	✗	✗	✓
postgres	✓	✓	✓	✗
sqlserver	✓	✓	✓	✗

 Remove any unnecessary parameters.

JSON data type: Integer.

SSL mode

The security mode for the TCP/IP connections with the database. Select:


- **enable** for trying only SSL connections and verify that:
 - The server certificate is issued by the CA selected in [SSL validation certificate](#).
 - The server hostname matches the hostname in the certificate.
- **disable** for trying only non-SSL connections.

Mandatory: Yes.

JSON data type: String ("enable" or "disable").

SSL validation certificate

The certificate for validating the database TLS certificate of the database server. Click **Select Files** to import this certificate from file.

 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: When [SSL mode](#) is **enable**

JSON data type: The String representation of a file path.

Status Feeder password

The password assigned to the `STATUSFEEDER_DB_PASSWORD` parameter when [Initializing the Entrust Validation Authority database](#).

Mandatory: Yes.

JSON data type: The string identifier of a secret holding the PIN value.

 Run the `clusterctl solution secret set` command to set secret values manually.

Status Feeder User

The user identifier assigned to the `STATUSFEEDER_DB_USER` parameter when [Initializing the Entrust Validation Authority database](#).

Mandatory: Yes.

JSON data type: String.

Hsm

Select the **Hsm** tab of the **Configuration** page to configure the connection with the HSM (Hardware Security Module).

- [Vendor](#)
- [Token label](#)
- [HSM PIN](#)
- [Number of sessions](#)

Mandatory: Yes.

JSON data type: Object.

Vendor

The vendor of the security module.

Vendor	Security module
none	Built-in software PKCS #11 module. Omit the other HSM configuration settings when selecting this module.
nshield	nShield HSM. See Entrust Validation Authority HSM requirements for the supported models.
thales	Thales HSM. See Entrust Validation Authority HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: Yes.

JSON data type: String.

Token label

The label of the token that contains the private keys of the OCSP server certificate.

 See [Generating a VA certificate and key pair](#) for how to generate the OCSP server certificate keys.

Mandatory: When the `Vendor - tsa` value is `nshield` or `thales`.

JSON data type: String. Remove this parameter from the JSON file if `Vendor - tsa` is `none`.

HSM PIN

The PIN (Personal Identification Number) of the HSM (Hardware Security Module).

Mandatory: When the `Vendor - tsa` value is `nshield` or `thales`.

JSON data type: The string identifier of a secret holding the PIN value. See the Entrust PKI Hub for how to set secret values with the following command.

```
clusterctl solution secret set
```

Number of sessions

The maximum number of concurrent PKCS #11 sessions on the HSM.

Mandatory: When the `Vendor - tsa` value is `nshield` or `thales`.

JSON data type: Integer.

OCSP Responder-Server

Select the **OCSP Responder-Server** tab of the **Configuration** page to configure optional OCSP responder settings.

- [Read timeout](#)
- [Write timeout](#)
- [Idle timeout](#)
- [Max header bytes](#)
- [Max body bytes](#)
- [Graceful timeout](#)
- [Listen limit](#)
- [Keep alive](#)
- [Response Profile ID](#)
- [HTTP Error](#)

Mandatory: No

JSON data type: Object.

Read timeout

The maximum allowed period for reading an entire request, including the body. When this period expires, the request gets the following response.

```
Code=503,Reason=Service Unavailable
```

Mandatory: No. This optional value defaults to 60 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Write timeout

The maximum period allowed for writing a response. When this period expires, the request gets the following response.

```
Code=503,Reason=Service Unavailable
```

Mandatory: No. This optional value defaults to 60 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Idle timeout

The maximum period to wait for the next request when keep-alives are enabled.

Mandatory: No. This optional value defaults to 10 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Max header bytes

The maximum number of bytes allowed for keys and values in the request header, including the request line.

Mandatory: No. This optional value defaults to 1024.

JSON data type: Integer.

Max body bytes

The maximum number of bytes allowed in the request body.

Mandatory: No. This optional value defaults to 8192.

JSON data type: Integer.

Graceful timeout

The grace period before shutting down the server.

Mandatory: No. This optional value defaults to 15 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Listen limit


The maximum number of outstanding requests.

Mandatory: No. This optional value defaults to 0 (no limit).

JSON data type: Integer.

Keep alive

The TCP keep-alive timeouts on accepted connections.

 When this period expires, the server prunes dead TCP connections.

Mandatory: No. This optional value defaults to 3 minutes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Response Profile ID



The identifier of the profile for generating OCSP responses.



Mandatory: No. This optional value defaults to the **basic** identifier of the only supported profile. This profile:

1. Sets `byKey` as responder identifier.
2. If present in the request, copies the `id-pkix-ocsp-nonce` extension value in the response.
3. Signs the response with the SHA-256 algorithm.

JSON data type: The "basic" String.

HTTP Error

The HTTP error returned in the OCSP response body for failed requests. See the table below for the value returned when enabling  or disabling  this parameter.

Request type		
Invalid request	HTTP 400	HTTP 200
Valid request that could not be processed	HTTP 404	HTTP 200

Mandatory: No. This optional value defaults to `false` (disabled).

JSON data type: Boolean.

LDAP Servers

In the **LDAP Servers** tab, add the following parameter for each LDAP server hosting a CRL (Certificate Revocation List).

- [Choose a key name](#)


Mandatory: No.

JSON data type: Dictionary.

Choose a key name

Type the unique identifier of an LDAP server in the "Choose a key name" text field, press **Enter** and configure the following settings.

- [URL](#)
- [Username](#)
- [Password](#)

 The selected identifier must match the value assigned to the [LDAP Server ID](#) parameter.

Mandatory: Yes.

JSON data type: The String representation of a dictionary key. For example, `myLdapServer` in:

```
"ldapServers": {
  "myLdapServer": {
    "url": "ldap://127.0.0.1",
    "userName": "myUsername",
    "passwordID": "myPasswordID"
```

```
}  
}
```

URL

The URL of the LDAP server in the following format:

```
ldap://<host>:<port>
```

For example:

```
ldap://ldap.example.com:389
```

Mandatory: Yes.

JSON data type: String.

Username

The user name for binding to the LDAP server.

Mandatory: When the binding to the LDAP server is not anonymous,

JSON data type: String.

Password

The password of the [Username](#) user.

Mandatory: When the [Username](#) user requires password authentication.

JSON data type: The string identifier of a secret holding the PIN value.

 Run the `clusterctl solution secret set` command to set secret values manually.

Certificate Authorities

In the **Certificate Authorities** tab, add the following parameters for each Certificate Authority that will issue certificates validated by Entrust Validation Authority.

- [CA ID](#)
- [Certificates Source](#)
- [CA Gateway](#)
- [Certificate Revocation List](#)
- [Certificate Revocation List in HTTP server](#)
- [Certificate Revocation list in LDAP server](#)
- [Serial number list HTTP](#)
- [OCSP Responder](#)

Mandatory: Yes.

JSON data type: List.

CA ID

The identifier of the CA that issues the certificates. The values supported by this parameter depend on the [Certificates Source](#) value.

Certificates Source	CA ID
CAGW	The CA identifier in CA Gateway.
CRL	Any user-defined value that uniquely identifies the CA in Entrust Validation Authority.

Mandatory: Yes.

JSON data type: String.

Certificates Source

The source informing on the validity status of the issued certificates.

Value	Source
CAGW	An instance of Entrust CA Gateway.
CRL	A CRL (Certificate Revocation List) published on an HTTP or LDAP server.

Before changing this parameter value on a deployed Entrust Validation Authority:

1. Stop the solution with the `evactl stop` command.
2. Remove all the database records related to the CA.
3. Restart the solution with the `clusterctl deploy` command or the management console of Entrust PKI Hub.

Mandatory: Yes.

JSON data type: String ("CAGW" or "CRL").

CA Gateway

CA Gateway configuration parameters.

- [URL](#)
- [Wait to pull certs duration](#)
- [Wait on error duration](#)
- [Batch Size](#)
- [Timeout](#)
- [TLS client certificate](#)
- [TLS CA certificate](#)
- [Push by serial](#)

Mandatory: When [Certificates Source](#) is **CAGW**.

✘ When using CA Gateway to check the status of the certificates, Entrust Validation Authority has the known issues described in the release notes.

JSON data type: Object.

URL

The URL of the CA Gateway server. For example:

```
https://127.0.0.1/cagw
```

Mandatory: Yes.

JSON data type: String.

Wait to pull certs duration

The waiting time for Entrust Validation Authority while not receiving new events. When this period expires, Entrust Validation Authority sends a new request to CA Gateway.

Mandatory: Yes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Wait on error duration

The waiting time before retrying a failed connection with CA Gateway or the Status Feeder internal service.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Mandatory: No. This optional value defaults to "5s".

Batch Size

The maximum number of certificates to retrieve in every request to CA Gateway. Select an integer value starting from 1.

✘ The `cagw/v1/certificate-authorities/<caid>/certificate-events` endpoint of CA Gateway must support the selected value.

Mandatory: No. This optional value defaults to 50.

JSON data type: Integer.

Timeout

The timeout for connections with the CA Gateway server. When a connection attempt with the CA Gateway server exceeds this period:

1. The request fails.
2. Entrust Validation Authority tries another connection after the period configured in [Wait on error duration \(WaitOnErrorDuration\)](#)

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Mandatory: No. This optional value defaults to "5s".

TLS client certificate


The identifier of the certificate for connecting to CA Gateway. Run the `evactl list-certs` command to list the available certificate identifiers.

Mandatory: No. This optional parameter defaults to the latest client certificate imported as explained in [Importing the CA Gateway client certificate](#).

JSON data type: String.

TLS CA certificate

The CA certificate for validating the CA Gateway TLS server certificate. Click **Select Files** to import this certificate from file.

 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: Yes.

JSON data type: The String representation of a file path.

Push by serial

The certificate information pushed into the Entrust Validation Authority database.

- Check this box to push the certificate serial number
- Uncheck this box to push the whole DER encoding of the certificate

Mandatory: No. This optional value defaults to `false` (unchecked).

JSON data type: Boolean.

Certificate Revocation List

Configure the following Certificate Revocation List (CRL) configuration parameters.

- [Wait to pull certs duration](#)
- [Wait on error duration](#)
- [CRL warning time](#)
- [CRL Host Server](#)
- [Use SN Lists](#)

Mandatory: When [Certificates Source](#) is **CRL**.

JSON data type: Object.

Wait to pull certs duration

The period between:

- The last upload of the CRL data into the database.
- The next request to the CRL server.

When [Use SN Lists](#) is `true`, Entrust Validation Authority will pull the CRL and the serial number list.

Mandatory: Yes.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Wait on error duration

The waiting time before retrying a failed connection with the CRL server, the Status Feeder internal service or the serial number list server.

Mandatory: No. This optional value defaults to "5s".

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

CRL warning time

The period during which to enable the expiration warning for the last processed CRL. When the time remaining before the CRL expiration is shorter than this parameter value, the `CRLExpirationWarning` metric is set to 1.

Mandatory: No. This optional value defaults to 4 hours.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

CRL Host Server

The type of server hosting the CRL.

- HTTP
- LDAP

Mandatory: Yes.

JSON data type: String ("HTTP" or "LDAP").

Use SN Lists

`true` to use certificate serial number lists, `false` otherwise. Set this value to `true` when `Profile ID` is one of the following profile identifiers.

- SNListProfile
- SNListProfileWithArchiveCutOff

Entrust Validation Authority will pull a serial number list and return the following status for certificates.

Certificate status	The certificate is in the CRL	The certificate is in the SNL
good		✓
revoked	✓	✓
unknown		

Mandatory: Yes.

JSON data type: Boolean.

Certificate Revocation List in HTTP server

Configuration parameters of the HTTP server hosting the CRL.

- [CRL HTTP URL](#)
- [Connection timeout](#)

Mandatory: When [CRL Host Server](#) is **HTTP**.

JSON data type: Object.

CRL HTTP URL

The URL of a CRL hosted in an HTTP server – for example:

```
http://127.0.0.1/crl.crl
```

Mandatory: Yes.

JSON data type: String.

Connection timeout

The timeout for connections with the CRL server. When the connection attempt exceeds this value:

1. The request fails.
2. Entrust Validation Authority tries another connection after the [Wait on error duration - cagw](#).

Mandatory: No. This optional value defaults to 5 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

Certificate Revocation list in LDAP server

Configuration parameters of the LDAP server hosting the CRL.

- [LDAP Server ID](#)
- [Connection timeout](#)
- [CRL Entry Distinguished Name](#)
- [CRL Attribute Name](#)

Mandatory: When [CRL Host Server](#) is **LDAP**.

JSON data type: Object.

LDAP Server ID

The identifier of the LDAP server.

Mandatory: Yes.

JSON data type: String.

Connection timeout

The timeout for connections with the CRL server. When the connection attempt exceeds this value:

1. The request fails.
2. Entrust Validation Authority tries another connection after the period configured in [Wait on error duration - cagw](#).

Mandatory: No. This optional value defaults to 5 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

CRL Entry Distinguished Name

The Distinguished Name of the entry that contains the CRL in the LDAP server.

Mandatory: Yes.

JSON data type: String.

CRL Attribute Name

The name of the attribute that contains the CRL in the corresponding entry of the LDAP server.

Mandatory: No. This optional parameter defaults to:

```
certificateRevocationList;binary
```

JSON data type: String.

Serial number list HTTP

The configuration settings of the certificate serial number list hosted in an HTTP server.

- [Serial Number list URL](#)
- [Connection timeout](#)

JSON data type: Object.

Mandatory: When [Use SN Lists](#) is `true`.

Serial Number list URL

The URL of the HTTP server that hosts a list containing the serial numbers of all certificates issued by the CA.

Mandatory: Yes.

JSON data type: String.

Connection timeout

The timeout for connections with the HTTP server that hosts a list containing the serial numbers of all certificates issued by the CA. When the connection attempt exceeds this value:

1. The request fails.
2. Entrust Validation Authority tries another connection after the [Wait on error duration](#) (`WaitOnErrorDuration`).

Mandatory: No. This optional value defaults to 5 seconds.

JSON data type: A number followed by `ns`, `us`, `ms`, `s`, `m` or `h`. For example, "60s".

OCSP Responder

Configure the following parameters of the OCSP responder service provided by Entrust Validation Authority.

- [Profile ID](#)
- [CA certificate](#)
- [VA certificate](#)

Profile ID

The identifier of the profile for processing the certificate status before generating an OCSF response. See below for the response settings defined by each profile.

Profile identifier	nextUpdate	id-pkix-ocsp-archive-cutoff	Status if unknown	Revocation date
basic	—	—	revoked	Jan 1 00:00:00 1970 GMT
archiveCutOff	—	notBefore date of the CA certificate	revoked	Jan 1 00:00:00 1970 GMT
nextUpdate	thisUpdate + 8 hours	—	revoked	Jan 1 00:00:00 1970 GMT
archiveCutOffWithNextUpdate	thisUpdate + 8 hours	notBefore date of the CA certificate	revoked	Jan 1 00:00:00 1970 GMT
CRLProfile	—	—	good	—
CRLProfileWithArchiveCutOff	—	notBefore date of the CA certificate	good	—
SNListProfile	—	—	unknown	—
SNListProfileWithArchiveCutOff	—	notBefore date of the CA certificate	unknown	—

See the below for the [Certificates Source](#) and [Use SN Lists](#) values supported by each profile.

Profile identifier	Certificates Source	Use SN Lists
basic	CAGW	—
archiveCutOff	CAGW	—
nextUpdate	CAGW	—
archiveCutOffWithNextUpdate	CAGW	—
CRLProfile	CRL	false


Profile identifier	Certificates Source	Use SN Lists
CRLProfileWithArchiveCutOff	CRL	false
SNListProfile	CRL	true
SNListProfileWithArchiveCutOff	CRL	true

Mandatory: Yes.

JSON data type: String.

CA certificate

Click **Select Files** to import the certificate of the CA that issues the certificates validated by Entrust Validation authority.


 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: Yes.

JSON data type: The String representation of a file path.

VA certificate

The certificate described in [Generating a VA certificate and key pair](#). Click **Select Files** to import this certificate from file.

 Each certificate file must contain a certificate in PEM format and Base64 encoding.

Mandatory: Yes.

JSON data type: The String representation of a file path.

Testing the OCSP Responder

After deploying Entrust Validation Authority, you can test the OCSP Responder service as follows.

- [Testing the OCSP Responder with openssl](#)
- [Testing the OCSP Responder with the health check endpoint](#)

Testing the OCSP Responder with openssl

Run the following `openssl` command to test the OCSP Responder service.

```
openssl ocspl -issuer <ca_cert> -serial <sn> -url <url> -VAfile <va_cert>
```

For example:

```
$ openssl ocspl -issuer issuer.pem -serial 0x000000002439fa8f5fe6370bb20ccb2556da6991
-url http://10.1.141.37/eva -VAfile ./VAFile.pem
```

```
Response verify OK
0x000000002439fa8f5fe6370bb20ccb2556da6991: good
  This Update: Nov  7 18:52:34 2022 GMT
  Next Update: Nov  8 02:41:13 2022 GMT
```

See below for a description of each command option.

- `-issuer <ca_cert>`
- `-serial <sn>`
- `-url <url>`
- `-VAfile <va_cert>`

⚠ The OCSP Responder service reboots when losing connection with the HSM. Run the `evactl check all` command to check the HSM connection and other settings.

`-issuer <ca_cert>`

Validate the status of a certificate issued by the `<ca_cert>` CA. Where `<ca_cert>` is the file path of the CA certificate.

`-serial <sn>`

Validate the status of the certificate with the `<sn>` serial number.

`-url <url>`

Connect to the `<url>` Entrust Validation Authority service. Where `<url>` is an URL in the following format.

```
http://<host>/eva
```

Where `<host>` is the IP address or hostname of the host running Entrust Validation Authority.

`-VAfile <va_cert>`

Validate the response with the `<va_cert>` certificate. Where `<va_cert>` is the file path of the certificate:

- Generated as explained in [Generating a VA certificate and key pair](#).
- Assigned to the [VA certificate](#) configuration parameter.

Testing the OCSP Responder with the health check endpoint

Entrust Validation Authority exposes the following endpoint to check the health of the database and HSM connections.

```
http://<host>/eva/health
```

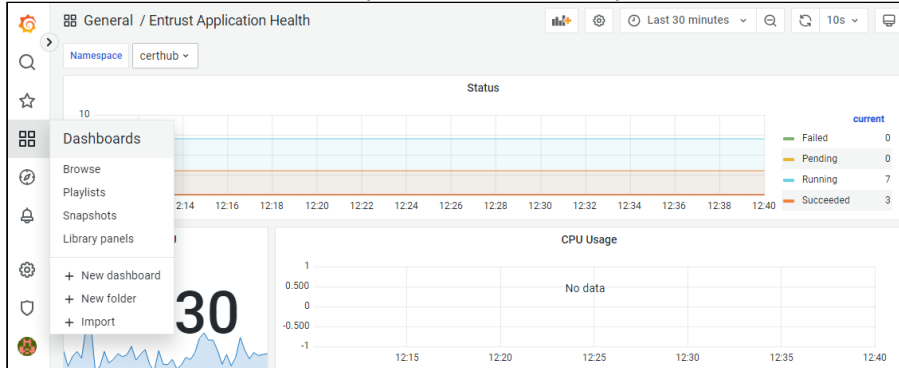
This endpoint returns an HTTP 503 response when the health check fails.

Browsing Entrust Validation Authority logs

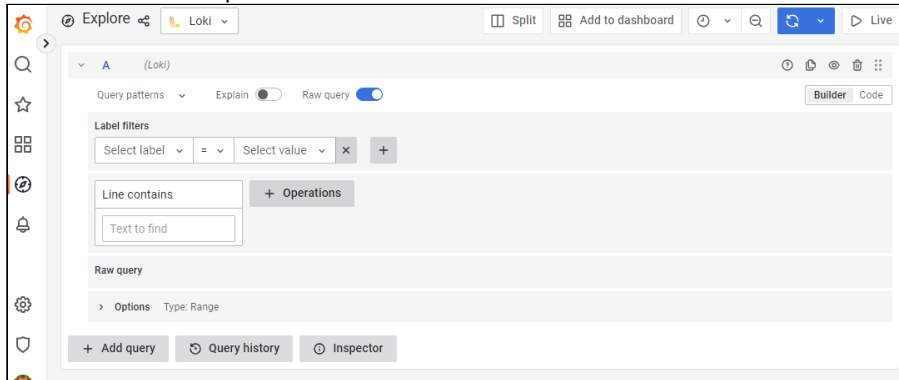
Entrust PKI Hub provides a Grafana web portal for browsing logs and metrics.

To browse Entrust Validation Authority logs with Grafana

1. Log into Grafana as explained in [Browsing logs with Grafana](#).
2. In the left sidebar of the Grafana portal, click the four squares icon and select **Dashboards**.



3. Select **Loki** in the top menu.



4. Select a time frame in the top menu.
 - [Querying Entrust Validation Authority logs](#)
 - [Querying OCSP request logs](#)
 - [Auditing the executed evactl commands](#)
5. Add the following log filters described in the below sections.
6. Click **Run query**.

Querying Entrust Validation Authority logs

Add the following filters under **Label filters** to query Entrust Validation Authority logs.

Select label	Select value	Query output
namespace	eva	Entrust Validation Authority logs.
app	<service>	Logs for the <service> service.

Querying OCSP request logs

Add the following under **Label filters** to query logs on OCSP request processing.

Select label	Select value	Query output
namespace	eva	Entrust Validation Authority logs.
app	eva-ocspresponder	Logs for the OCSP responder service.

Use the **Line contains** fields to add filters like the following.

Line contains	Query output
SendRevocationStatusRequest	Logs for OCSP requests.
SendRevocationStatusRequest.Failed	Logs for failed OCSP requests.
<sn>	Logs for requests on the certificate with the <sn> serial number.
<dn>	Logs for requests on the certificate issued by a CA with the <dn> distinguished name.

Auditing the executed evactl commands

Add the following filters under **Label filters** to audit the execution of the `evactl` command line tool.

Select label	Select value	Query output
filename	var/log/entrust/eva/evactl.log	A record of all the executed <code>evactl</code> commands

evactl reference

See the following sections for the commands supported by the `evactl` command-line tool.

- [evactl check all](#)
- [evactl check cert-source](#)
- [evactl check db](#)
- [evactl check hsm](#)
- [evactl create-csr](#)
- [evactl create-key](#)
- [evactl delete-key](#)
- [evactl enroll](#)
- [evactl export-nshield](#)
- [evactl import-nshield](#)
- [evactl import-p12](#)

- `evactl import-thales`
- `evactl list-certs`
- `evactl list-keys`
- `evactl load-oracle-wallet`
- `evactl reenroll`
- `evactl stop`

✘ The commands described in the following sections require passwordless `sudo` permissions.

evactl check all

Debugs all the components of the EVA deployment.

```
evactl check all [-c <tls_ca_path>] [-i <cert_id>] [-l <level>] [-p <pin>] [-v <vendor>] [-t <token>]
```

For example:

```
$ sudo ./evactl check all
Starting PKCS #11 Manager... Done
=====
CHECKING HSM
=====

Slot Id -> 0
Label -> pking203
Serial Number -> 1433959427612
Model -> LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration -> Luna User Partition With SO (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status -> FM Ready

Slot Id -> 1
Label -> pking202
Serial Number -> 1433964084224
Model -> LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration -> Luna User Partition With SO (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status -> FM Ready

Current Slot Id: 0

Passing HSM checks... Done
=====
```

```
CHECKING DB
=====
Starting Configurator...           Done
Checking DB user privileges...     Done
Checking DB tables...             Done

=====
CHECKING CAGW
=====

CAID: intminions~subordinate
  Checking CAGW is reachable...     Done
  Checking configured CA...         Done

=====

Tests passed successfully
```

See below for a description of each parameter.

- `-c <tls_ca_path>`
- `-i <cert_id>`
- `-l <level>`
- `-p <pin>`
- `-v <vendor>`
- `-t <token>`


`-c <tls_ca_path>`

Validate the TLS server certificate of CA Gateway with `<tls_ca_path>`. Where `<tls_ca_path>` is the path of a CA file in PEM format.

Mandatory: No. When omitting this option, the command uses the CA configured in [TLS CA certificate](#).

`-i <cert_id>`

Authenticate in CA Gateway with the `<cert_id>` certificate, where `<cert_id>` is a certificate identifier.


 Run the `evactl list-certs` command to list the available certificate identifiers.

Mandatory: No. This optional parameter defaults to the latest client certificate imported as explained in [Importing the CA Gateway client certificate](#).

 Run the `evactl list-certs` to command to check the latest imported certificate.

`-l <level>`

Debug the nShield HSM with the `<level>` level, where `<level>` is a `CKNFAST_DEBUG` variable level. When not using a nShield HSM, the command ignores this option.

 See the nShield documentation for details on the `CKNFAST_DEBUG` configuration parameter.

Mandatory: No. This optional parameter defaults to 0.

-p <pin>


Authenticate in the HSM with the <pin> PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.


-v <vendor>

Use the <vendor> security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).


Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

-t <token>

Select the HSM token with the <token> label.

Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

evactl check cert-source

Checks the certificate sources for every configured CA.

```
evactl check cert-source [-c <tls_ca_path>] [-i <cert_id>]
```

For example:

```
$ sudo ./evactl check cert-source
Starting Configurator... Done
```



```
CAID: ca~subordinate
  Checking CAGW is reachable...      Done
  Checking configured CA...          Done
```

```
Tests passed successfully
```

See below for a description of each option.

- `-c <tls_ca_path>`
- `-i <cert_id>`


`-c <tls_ca_path>`

Validate the TLS server certificate of CA Gateway with `<tls_ca_path>`. Where `<tls_ca_path>` is the path of a CA file in PEM format.

Mandatory: No. When omitting this option, the command uses the CA configured in [TLS CA certificate](#).

`-i <cert_id>`

Authenticate in CA Gateway with the `<cert_id>` certificate, where `<cert_id>` is a certificate identifier.

 Run the `evactl list-certs` command to list the available certificate identifiers.

Mandatory: No. This optional parameter defaults to the latest client certificate imported as explained in [Importing the CA Gateway client certificate](#).

 Run the `evactl list-certs` to command to check the latest imported certificate.

evactl check db

Checks the database connectivity, required users, user privileges, and tables.

```
evactl check db
```

For example:

```
$ sudo ./evactl check db
Starting Configurator...      Done
Checking DB user privileges... Done
Checking DB tables...         Done
```

evactl check hsm

Checks the HSM connectivity.

```
evactl check hsm [-l <level>] [-p <pin>] [-v <vendor>] [-t <token>]
```

For example:

```
$ sudo ./evactl check hsm
Starting PKCS #11 Manager... Done

Slot Id -> 0
Label -> pking203
Serial Number -> 1433959427612
Model -> LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration -> Luna User Partition With S0 (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status -> FM Ready

Slot Id -> 1
Label -> pking202
Serial Number -> 1433964084224
Model -> LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration -> Luna User Partition With S0 (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status -> FM Ready

Current Slot Id: 0

Passing HSM checks... Done
```

See below for a description of each option.

- `-l <level>`
- `-p <pin>`
- `-v <vendor>`
- `-t <token>`

`-l <level>`

Debug the nShield HSM with the `<level>` level, where `<level>` is a `CKNFAST_DEBUG` variable level. When not using a nShield HSM, the command ignores this option.



See the nShield documentation for details on the `CKNFAST_DEBUG` configuration parameter.

Mandatory: No. This optional parameter defaults to 0.

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.


-v <vendor>

Use the <vendor> security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).


Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

-t <token>

Select the HSM token with the <token> label.

Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

evactl create-csr

Checks the HSM connectivity.

```
evactl check hsm [-l <level>] [-p <pin>] [-v <vendor>] [-t <token>]
```

For example:

```
$ sudo ./evactl check hsm
Starting PKCS #11 Manager... Done

Slot Id -> 0
Label -> pking203
Serial Number -> 1433959427612
Model -> LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration -> Luna User Partition With S0 (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
```

```

FM HW Status ->      FM Ready

Slot Id ->          1
Label ->            pking202
Serial Number ->    1433964084224
Model ->            LunaSA 7.2.0
Firmware Version -> 7.0.3
Configuration ->    Luna User Partition With SO (PED) Signing With
Cloning Mode
Slot Description -> Net Token Slot
FM HW Status ->    FM Ready

Current Slot Id: 0

Passing HSM checks...           Done


```

See below for a description of each option.

- `-k <key_id>`
- `-s <subject>`
- `-o <csr>`
- `-p <pin>`
- `-t <token>`
- `-v <vendor>`
- `-y`

`-k <key_id>`


Select the key with the `<key_id>` identifier.

 Run the `evactl list-keys` command to get the key identifiers.

Mandatory: Yes.

`-s <subject>`

Use `<subject>` as the Subject of the certificate request. Where `<subject>` is a full Distinguished Name (DN) or Relative Distinguished Name (RDN).

 For Entrust Validation Authority to recognize the Subject, the DN attributes must be in capital letters.

For example:

```
CN=Example User,O=Example,C=US
```

```
CN=Example User
```

Mandatory: No. When omitting this option, the Subject in the generated certificate request defaults to the following:

```
CN=<key_id>
```

Where `<key_id>` is the key identifier.

```
-o <csr>
```

Save the certificate signing request (CSR) in a file with the `<csr>` path.

Mandatory: No. When omitting this option, the command prints the CSR to the standard output.

```
-p <pin>
```


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

```
-t <token>
```

Select the HSM token with the `<token>` label.


Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


```
-v <vendor>
```

Use the `<vendor>` security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

```
-y
```

Skip the confirmation prompt.

evactl create-key

Generates the key pair and the certificate signing request (CSR) of the certificate for signing OCSP responses.

```
evactl create-key -k <key_type> [-s <subject>] [-o <csr>] [-p <pin>] [-t <token>] [-v <vendor>] [-y]
```

For example:

```
$ sudo ./evactl create-key -k RSA2048 -s "CN=97357462, O=Entrust, C=ES"
Starting PKCS #11 pod... Done
Using token with label mytoken
Created key with id 4a00a4617d1afd5ad626955132dd0d396a69ed24
CSR:
-----BEGIN CERTIFICATE REQUEST-----
MIICqDCCAACAQAwMzExMC8GA1UEAxMoNGEwMGE0NjE3ZDFhZmQ1YWQ2MjY5NTUx
...
etTv+pac+nJKW8fw
-----END CERTIFICATE REQUEST-----
```

See below for a description of each option.

- -k <key_type>
- -s <subject>
- -o <csr>
- -p <pin>
- -t <token>
- -v <vendor>
- -y

-k <key_type>

Create a key of the <key_type> type, where <key_type> is one of the following.


- RSA2048
- RSA3072
- RSA4096
- ECDSAP256
- ECDSAP384
- ECDSAP521

Mandatory: Yes.

-s <subject>

Use <subject> as the Subject of the certificate request. Where <subject> is either:

- A full Distinguished Name (DN)
- A Relative Distinguished Name (RDN).

 The DN attributes must be in capital letters for the Subject to be recognized.

For example:

```
CN=Example User,O=Example,C=US
```

```
CN=Example User
```

Mandatory: No. When omitting this option, the Subject in the generated certificate request defaults to the following:

```
CN=<key_id>
```

Where `<key_id>` is the key identifier.

`-o <csr>`

Save the certificate signing request (CSR) in a file with the `<csr>` path.

Mandatory: No. When omitting this option, the command prints the CSR to the standard output.

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

`-t <token>`

Select the HSM token with the `<token>` label.


Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.

`-v <vendor>`

Use the `<vendor>` security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

✘ The command will raise an error if you omit this option and the configuration is not loaded.

-y

Skip the confirmation prompt.

evactl delete-key

Deletes a key.

```
evactl delete-key -k <key-id> [-p <pin>] [-t <token>] [-v <vendor>] [-y]
```

For example:

```
$ sudo ./evactl delete-key -k c403e0abae421c73625666dcff26dacf184eddd4 -y
Starting PKCS #11 Manager...                               Done
Using token with label pking203
Deleted public key with id c403e0abae421c73625666dcff26dacf184eddd4
Deleted private key with id c403e0abae421c73625666dcff26dacf184eddd4
```

See below for a description of each option.

- -k <key_id>
- -p <pin>
- -t <token>
- -v <vendor>
- -y

-k <key_id>

Select the key with the <key_id> identifier.

i Run the `evactl list-keys` command to get the key identifiers.

Mandatory: Yes.

-p <pin>

Authenticate in the HSM with the <pin> PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

-t <token>

Select the HSM token with the <token> label.

Mandatory: No. When omitting this option, the command uses the value of the [Token label](#) configuration parameter.

✘ The command will raise an error if you omit this option and the configuration is not loaded.

-v <vendor>

Use the <vendor> security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

⚠ It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the [Vendor](#) configuration parameter.

✘ The command will raise an error if you omit this option and the configuration is not loaded.

-y

Skip the confirmation prompt.

evactl enroll

```
evactl enroll -a <auth_code> -r <ref_number> -u <url> -l <ca_label> [-c <tls_ca_path>]
```

For example:

```
$ sudo ./evactl enroll -r 12473209 -a KNII-F4UH-8VX3 -u https://mycagateway.example.com:9443 -l my_issuing_ca
```

See below for a description of each option.

-a <auth_code>

Authenticate in the End Entity Enrollment server of CA Gateway with the <auth_code> authorization code.

Mandatory: Yes.

-c <tls_ca_path>

Validate the TLS server certificate of CA Gateway with <tls_ca_path>. Where <tls_ca_path> is the path of a CA file in PEM format.

Mandatory: No. When omitting this option, the command uses the CA configured in [TLS CA certificate](#).

-l <ca_label>

Use the <ca_label> CA, where <ca_label> is the label of a CA in the End Entity Enrollment server of CA Gateway.

Mandatory: Yes.

-r <ref_number>

Authenticate in the End Entity Enrollment server of CA Gateway with the <ref_number> reference_number.

Mandatory: Yes.

-u <url>

Select the End Entity Enrollment server of CA Gateway exposed in the <url> URL.

Mandatory: Yes.

evactl export-nshield

Saves a copy of the nShield Security World keys and configuration currently loaded in EVA. You can later import it with [evactl import-nshield](#), even in a different deployment of EVA.

```
evactl export-nshield -f <kmdata_dir> [-t]
```

For example:

```
$ sudo ./evactl export-nshield -f /opt/nfast/copy-kmdata
```

See below for a description of each option.

- -f <kmdata_dir>
- -t

-f <kmdata_dir>

Save the configuration in the <kmdata_dir> folder.

Mandatory: Yes.

-t

Save the configuration in the following compressed file.

```
<kmdata_dir>/kmdata.tar.gz
```

Mandatory: No. When omitting this option, the command does not compress the configuration in the <kmdata_dir> folder.

evactl import-nshield

Imports the nShield Security World configuration so Entrust Validation Authority can use the keys managed by the nShield HSM.


```
evactl import-nshield -f <kmdata> [-y]
```

For example:

```
$ sudo ./evactl import-nshield -f ./kmdata
```

See below for a description of each option.


- `-f <kmdata>`
- `-y`

 Changes will be effective when deploying (or redeploying) the solution with the Management Console or the `clusterctl deploy` command.

`-f <kmdata>`

Import the `<kmdata>` configuration, where `<kmdata>` is one of the following.

- The path of the nShield `kmdata` folder.
- The path of a backup folder generated with the `evactl export-nshield` command.
- The path of a `tar.gz` backup file generated with the `evactl export-nshield` command.

 See [Loading the HSM configuration](#) for considerations on this configuration.

Mandatory: Yes.

`-y`

Skip the confirmation prompt.

evactl import-p12

Sets the certificate and key to authenticate in CA Gateway.

```
evactl import-p12 -f <p12> [-p <pwd>]
```

For example:

```
$ sudo ./evactl import-p12 -f eva-cagw.p12 -p password
Starting Configurator...           Done
Importing P12...                   Done
```

See below for a description of each option.

- `-f <p12>`

- `-p <pwd>`

⚠ Changes will be effective when deploying (or redeploying) the solution with the Management Console or the `clusterctl deploy` command.

`-f <p12>`

The path of the PKCS #12 file containing the key and the certificate.

⚠ See [Importing the CA Gateway client certificate](#) for considerations on this file.

Mandatory: Yes.

`-p <pwd>`

The password of the PKCS #12.

Mandatory: No. When omitting this option, the command prompts for the password.

evactl import-thales

Imports the configuration of a Thales HSM. Use the following syntax to import this configuration from a ZIP file.

```
evactl import-thales -d <package_path> [-y]
```

Use the following syntax to import this configuration from a Chrystoki file.

```
evactl import-thales -c <cert_dir> -k <chrystoki> [-y]
```

For example:

```
$ sudo ./evactl import-thales -c ./eva-thales-config/cert -k ./eva-thales-config/Chrystoki.conf -y
Saving Thales configuration... Done
Warning: EVA is already deployed! To apply the changes, EVA needs to be redeployed using the evactl deploy command.
```

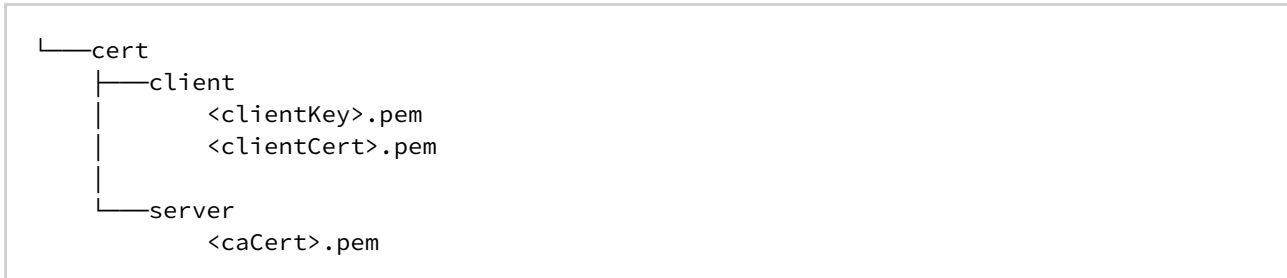
See below for a description of each option.

- `-c <cert_dir>`
- `-d <package_path>`
- `-k <chrystoki>`
- `-y`

⚠ Changes will be effective when deploying (or redeploying) the solution with the Management Console or the `clusterctl deploy` command.

`-c <cert_dir>`

Import the client and server certificates for the Luna Network or DPoD authentication. Where `<cert_dir>` is the path of a `cert` directory with the following contents.



See below for a description of each field.

Value	Description
<code><clientKey></code>	The file name of a PEM file containing the client's private key.
<code><clientCert></code>	The file name of a PEM file containing the client's certificate.
<code><caCert></code>	The file name of a PEM file containing the CA certificate for validating the server's certificate.

After running the command, verify the `Chrystoki.conf` file includes the following configuration.

```

ClientPrivKeyFile = /usr/safenet/lunaclient/cert/client/<clientKey>.pem;
ClientCertFile = /usr/safenet/lunaclient/cert/client/<clientCert>.pem;
ServerCAFile = /usr/safenet/lunaclient/cert/server/<caCert>.pem;

```

 Do not modify any other path in the `Chrystoki.conf` file.

Mandatory: Yes.

`-d <package_path>`

Use the `<package_path>` DPoD configuration package, where `<package_path>` is the path of the ZIP package file.

Mandatory: Yes.

`-k <chrystoki>`

Import the `<chrystoki>` configuration of the Luna Network or DPoD client, where `<chrystoki>` is the path of the `Chrystoky.conf` file.

Mandatory: Yes.


```

1MGRiNTI3ODNhN2NmMzU0N2EzYmYyYjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCvtsMW84yx
E5Ln6XMprAE6rvIApr0fevYqoaBI89gOdTyf6LSer0eSigVGr4TMyW6yhmfyzQKvBevP/
rK4pCB6mBoAiUTr5QF7HwBUJ0z4xbmiIxwhXIFcVNVXKnCekg3eHVS9DShdjy4U/
Vt6tXSZmHkI1rZ1uEJweBwVLGp8MHHU6Rot0mZFZ0ourxihrCd3siDHXu6hADp54wHjGDET7V2WVPMKvcVAW
e+TgzAJcVpYBw07fj4kqER2UmloT0PffjRtddTC9swxl30aC+UfmleAoBrSjWB+O6lw60bp5oL1gPgIUD9R5fZ
osqPNbVfS5gs6/
Vo5BqfFcGIGMxVTAgMBAAGjggFSMIIBTjAMBGNVHRMBAf8EAjAAMB0GA1UdDgQWBBSk2b3+ZE3ZyIe4bTi3ph
Adv6jB0zAfbgNVHSMEGDAWgBQ4XuxK8LobB2x+dGs0jaqQo0BHwzBLBggrBgEFBQcBAQQ/
MD0wOwYIKwYBBQUHMAGGL2h0dHA6Ly9vY3NwLmRldi5wa2lodWIuY29tL29jc3AvZW50cnVzdC9pbmRjYWd3M
EYGA1UdHwQ/
MD0wO6A5oDeGNWh0dHA6Ly9jcmwuZGV2LnBraWh1Yi5jb20vY3JsL2VudHJ1c3QvaW50Y2Fndy9jcmwuY3JsM
EQGA1UdEQ9MDuGOWlkcDovL2ludGNhZ3dpZHAVMDc3NjJlYzUzN2NhYmFiZjUwZGI1Mjc4M2E3Y2YzNTQ3YT
NiZjJiMzA0BgNVHQ8BAf8EBAMCB4AwEwYDVR0lBAwwCgYIKwYBBQUHAWIwDQYJKoZIhvcNAQELBQADggEBAFy
o2+QcxN4gN8XxhnErYQ3ET9kk5hrXUa+RGGcTLiegiNKX/
fxlG0V51QglvP4rFd12bnYCMqSQuLq0H08m0E3U7wmKZem40Ml0Ifjp94RyDHaMnp0W0v9e4C6I6Q6nv4CX6nr
9TDmpIKG32c0kKu7veSZaLDBVA/Wg+W2ox6yf3W8PbPpUbf6Ld6UC/
gu0hzBMLqw8H+lq4WQs9KWcFFF5+XMm4y1Q38HIL0b0DBVpeib0QOno41mc9+7w/
0W3ix+DDcuLIJEMaKg0ynXujl9Ga0wfm7qZdC+eJ1z0N3m2HBVfZWFEwaKZ8lm2ZqkuZnCyKJ430qgP2X2WF
axav+c=",
  "SN": "31:5A:FC:C9:6E:FE:18:20:42:6B:79:72:44:14:AD:DE",
  "DN": "subject=O = intcagwidp, OU = CAGW, serialNumber =
07762ec537cababf50db52783a7cf3547a3bf2b3"
}
]

```

evactl list-keys

Lists the keys in the PKCS #11 token.

```
evactl list-keys [-p <pin>] [-t <token>] [-v <vendor>]
```

For example:

```

$ sudo ./evactl list-keys
Starting PKCS #11 Manager... Done
Using token with label pking203
Public Key Object; RSA 2048 bits
  Label:      305ecd78340acc3d906be370a01e7884
  ID:        03b1dac1e383b8d3adea5a6a2c6200bde58ffb40
  Usage:     verify

Private Key Object; RSA 2048 bits
  Label:      F
  ID:        0f
  Usage:     sign, unwrap

Public Key Object; RSA 2048 bits
  Label:      F
  ID:        0f
  Usage:     verify, wrap

```

```
Private Key Object; RSA 2048 bits
Label:      webserver-root1
ID:        103d6c94ea10b98ab37186cc1c4977eb
Usage:     sign
```

See below for a description of each option.

- `-p <pin>`
- `-t <token>`
- `-v <vendor>`

`-p <pin>`


Authenticate in the HSM with the `<pin>` PIN.

Mandatory: No. When omitting this option, the command looks for the PIN in the application secrets. If not found, prompts the user for the PIN.

`-t <token>`

Select the HSM token with the `<token>` label.


Mandatory: No. When omitting this option, the command uses the value of the `Token label` configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


`-v <vendor>`

Use the `<vendor>` security module. See the following table for the supported values.

Vendor	Security module
none	Built-in software PKCS #11 module.
nshield	nShield HSM. See HSM requirements for the supported models.
thales	Thales HSM. See HSM requirements for the supported models.

 It is recommended to select a Hardware Security Module (HSM).

Mandatory: No. When omitting this option, the command assumes the value of the `Vendor` configuration parameter.

 The command will raise an error if you omit this option and the configuration is not loaded.


evactl load-oracle-wallet

Loads the Oracle wallet for validating the TLS server certificate of an Oracle DBMS. Overwrites the wallet previously loaded.

```
evactl load-oracle-wallet -f <wallet-folder> [-y]
```


For example:

```
sudo ./evactl load-oracle-wallet -f ./oracle-wallet
Loading Oracle Wallet... Done
```

 If the value of the [Driver](#) configuration parameter is not `oracle`, skip this command and select the certificate with the [SSL validation certificate](#) parameter instead.

`-f <wallet-folder>`

Load the the `<wallet-folder>` Oracle Wallet containing the certificate. Where `<wallet-folder>` is the path of the Oracle Wallet folder.

 See the Oracle documentation for how to generate an Oracle Wallet with the `orapki` command-line tool.

Mandatory: Yes.

`-y`

Skip the confirmation prompt.

evactl reenroll

Issues a new certificate for authenticating in CA Gateway (see [evactl enroll](#) for how to generate the first certificate).


```
evactl reenroll -l <ca_label> -u <url> [-c <tls_ca_path>] [-i <cert-id>]
```

For example:

```
$ sudo ./evactl reenroll -u https://mycagateway.example.com:9443/.well-known/est/
intcagwidp/simplereenroll -l intcagwidp
```

See below for a description of each option.

- `-c <tls_ca_path>`
- `-i <cert_id>`
- `-l <ca_label>`
- `-u <url>`

 Run this command before the current certificate expires.


-c <tls_ca_path>

Validate the TLS server certificate of CA Gateway with <tls_ca_path> . Where <tls_ca_path> is the path of a CA file in PEM format.

Mandatory: No. When omitting this option, the command uses the CA configured in [TLS CA certificate](#).

-i <cert_id>

Authenticate in CA Gateway with the <cert_id> certificate, where <cert_id> is a certificate identifier.

 Run the `evactl list-certs` command to list the available certificate identifiers.

Mandatory: No. This optional parameter defaults to the latest client certificate imported as explained in [Importing the CA Gateway client certificate](#).

 Run the `evactl list-certs` to command to check the latest imported certificate.

-l <ca_label>

Use the <ca_label> CA, where <ca_label> is the label of a CA in the End Entity Enrollment server of CA Gateway.

Mandatory: Yes.

-u <url>

Select the End Entity Enrollment server of CA Gateway exposed in the <url> URL.

Mandatory: Yes.


evactl stop

Stops a deployed Entrust Validation Authority.

```
evactl stop
```

For example:

```
$ sudo ./evactl stop
Stopping Virtual Services... Done
Stopping Services... Done
Stopping Deployments... Done
Stopping Stateful Sets... Done
Stopping Pods... Done
```

 To restart Entrust Validation Authority, run the `clusterctl deploy` command.

Troubleshooting Entrust Validation Authority

See below for how to troubleshoot the main issues.

- [Connectivity issues](#)
- [Error: Another instance of evactl is running](#)

Connectivity issues

As explained in [Entrust Validation Authority overview](#), Entrust Validation Authority connects to:

- A database.
- An HSM.
- A certificate status source (CA Gateway instance or CRL server).

To check the connection with these components, run the `evactl check all` command.

Error: Another instance of evactl is running

When trying to create or delete a key, you can encounter the following error.

```
Error: Another instance of evactl is running create-key or delete-key
```

When trying to enroll, reenroll or import a PKCS #12, you can encounter the following error.

```
Error: Another instance of evactl is running enroll, reenroll or import-p12
```

In both cases:

1. Make sure that there is no other instance of the `evactl` command line tool performing any of those operations.
2. Re-run the command with the `FORCE_MUTEX_OPERATION` environment variable set to 1. For example:

```
sudo FORCE_MUTEX_OPERATION=1 ./evactl create-key RSA2048
```

✘ Running a command with `FORCE_MUTEX_OPERATION` set to 1 can override the changes made by another `evactl` running instance.

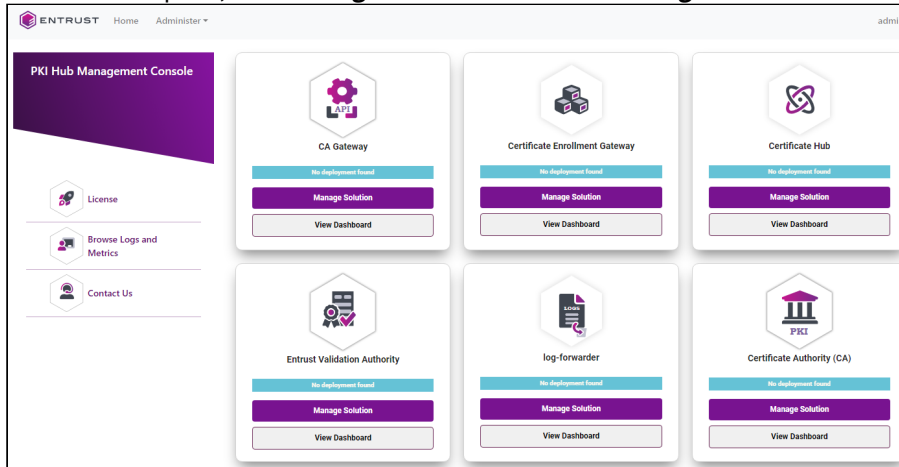
Starting up Entrust log-forwarder

Entrust PKI Hub provides the log-forwarder solution for forwarding logs to a Splunk SIEM (Security Information and Event Management) server.

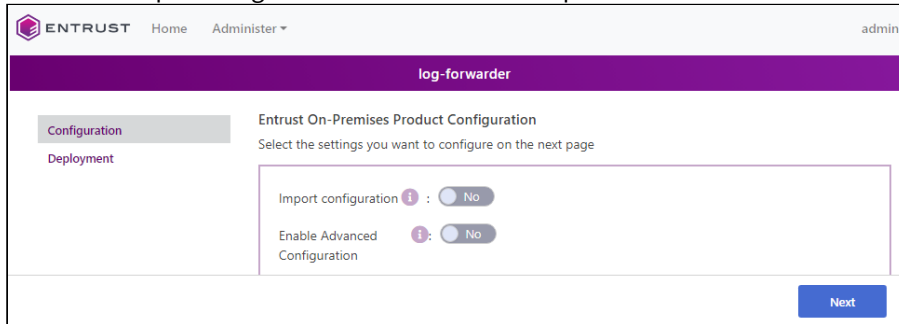
To configure and deploy Entrust log-forwarder with the Management Console

1. Login into the Management Console as explained in [Logging into the Management Console](#).

- In the content pane, click **Manage Solution** under **Entrust log-forwarder**.



- Activate the **Import configuration** toggle switch if you want to import configuration settings from a file, such as a sample configuration file included in the product release.




- Active the **Enable Advanced Configuration** if you want to configure the full set of configuration parameters supported by the solution.
- Click **Next**.
- Configure the solution settings described in the following sections.
 - Type
 - Host
 - Port
 - Token
 - TLS

- Click **Validate** to validate the configured settings.
- Correct any detected configuration error until the **Validate** option displays no warnings.
- Optionally, click the **Download** button to export the current configuration. You can later import this configuration with the already mentioned **Import configuration** toggle switch.
- Click **Submit** and wait while Entrust PKI Hub uploads the configuration and any attached file, such as a P12 file with authentication credentials.
- Click **Deploy**.

Type

The type of SIEM server. The current Entrust PKI Hub release only supports selecting **Splunk**.

 As explained in [SIEM requirements](#), the current Entrust PKI Hub release only supports the Splunk SIEM.

Mandatory: Yes


Host

The IP address or hostname of the external SIEM server.

Mandatory: Yes

Port

The port of the SIEM service.

 In the Splunk configuration, this port is the "HTTP Event Collector" port.

Mandatory: Yes

Token

A secret authentication token provided by the external SIEM service.

Mandatory: Yes

TLS

Configuration of the TLS security in communications with the external SIEM server.

Parameter	Value	Default
Enable	Mark this checkbox to use TLS security in the communications with the external SIEM server.	Disabled
Verify	Mark this checkbox to verify the TLS certificate of the external SIEM server.	Disabled
CA Certificate File	Click Select Files to import The CA certificate for validating the TLS certificate of the external SIEM server.	The system certificates

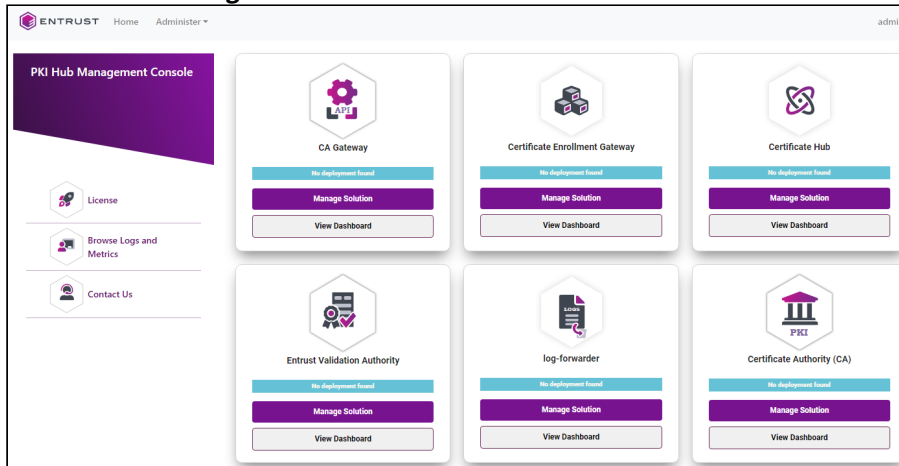
9 Browsing logs with Grafana

Entrust PKI Hub provides a Grafana portal to browse logs and metrics on the internal services and the installed Entrust solutions.

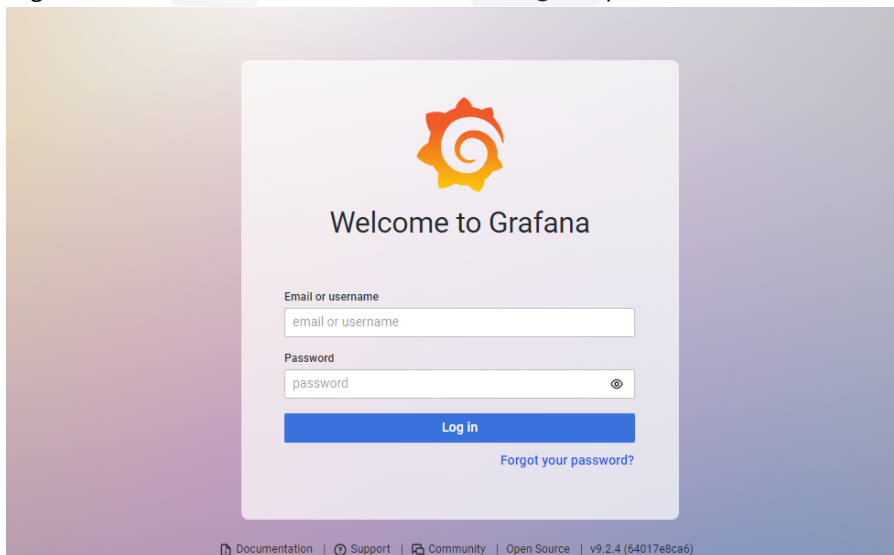
i See [Starting up Entrust log-forwarder](#) for how to forward logs to a Splunk SIEM (Security Information and Event Management) server.

To manage logs with Grafana

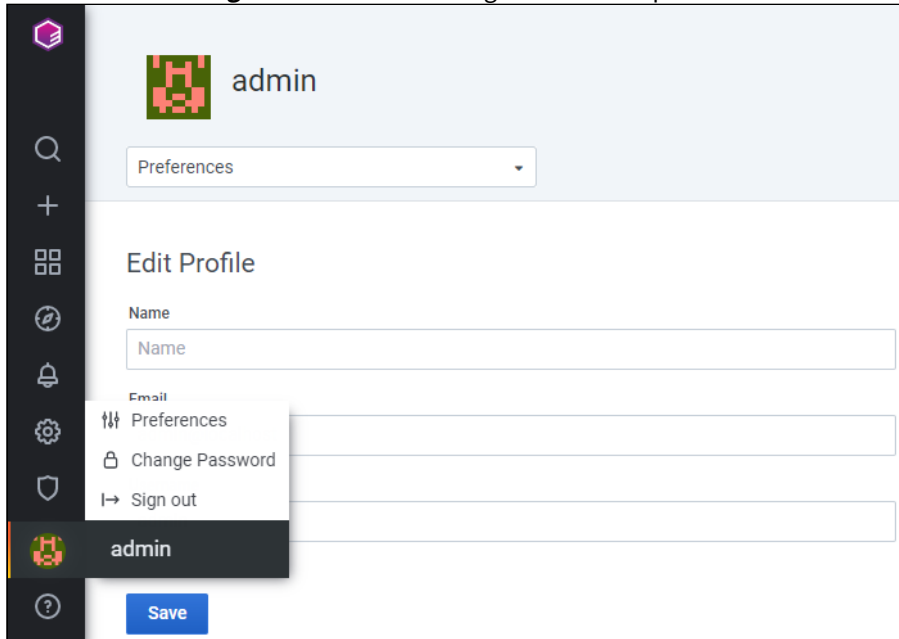
1. Login into the Management Console.
2. Click the **Browse Logs and Metrics** sidebar command.



3. Log in with the `admin` username and the `changeme` password.



- Go to **admin > Change Password** and change the admin's password.



- Perform the following operations.
 - [Browsing and exporting logs with the Grafana Loki Dashboard](#)
 - [Browsing log file contents with Grafana](#)

Browsing and exporting logs with the Grafana Loki Dashboard

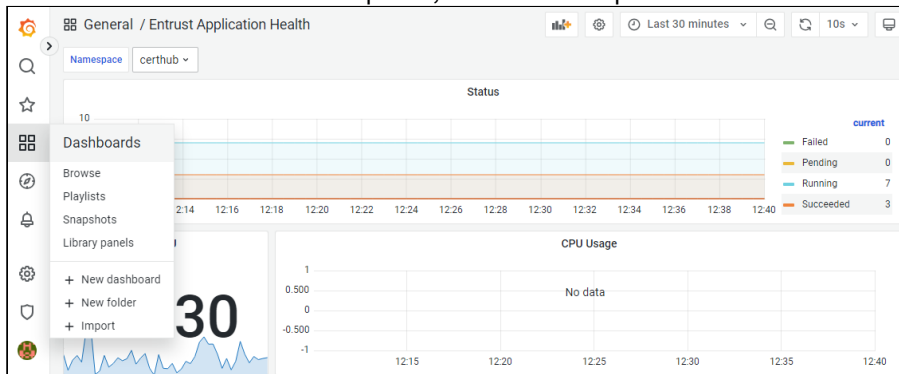
The Grafana Loki Dashboard allows browsing and exporting the logs of the following namespaces.

Namespace	Monitored service
<solution>	The deployed Entrust solution with the <solution> identifier – for example: <code>cagw</code> , <code>certhub</code> , <code>ceg</code> .
calico-system	The Calico networking service.
csf-docker-registry	The Docker registry.
longhorn-system	The Longhorn system.
istio-system	The Istio system.
csf-monitoring	The Prometheus service.
csf-logs	The Loki service.

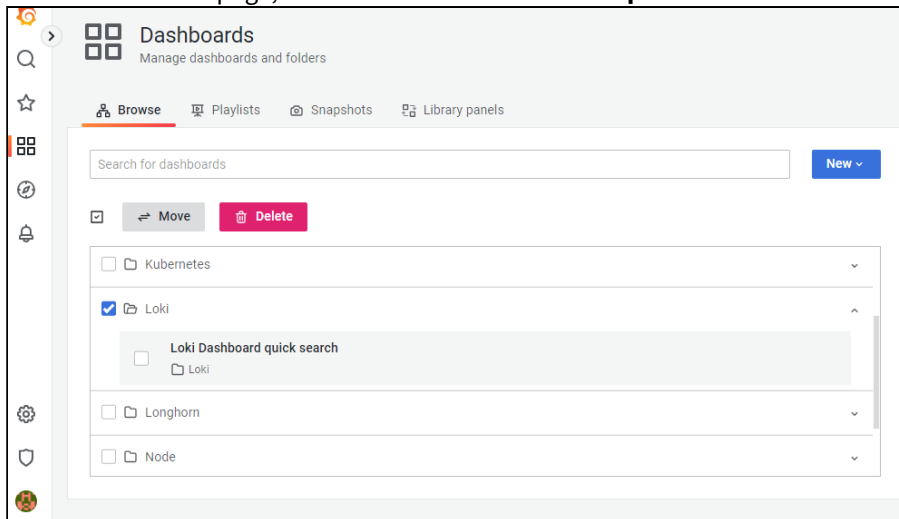
Namespace	Monitored service
kube-system	The K3s system.
management-console	The web console for deploying and managing Entrust solutions.
tigera-operator	The Tigera Operator.
auth-service	The authentication system for Entrust PKI Hub administrators.
solution-manager	The service for managing the deployed Entrust solutions.

To browse and export logs with the Grafana Loki Dashboard

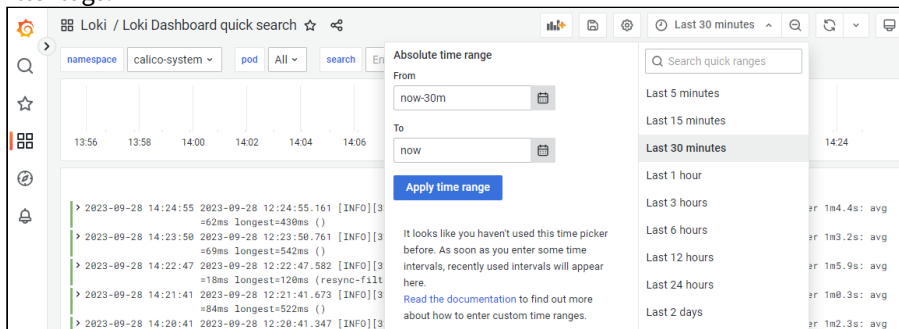
1. In the left sidebar of the Grafana portal, click the four squares icon and select **Dashboards**.



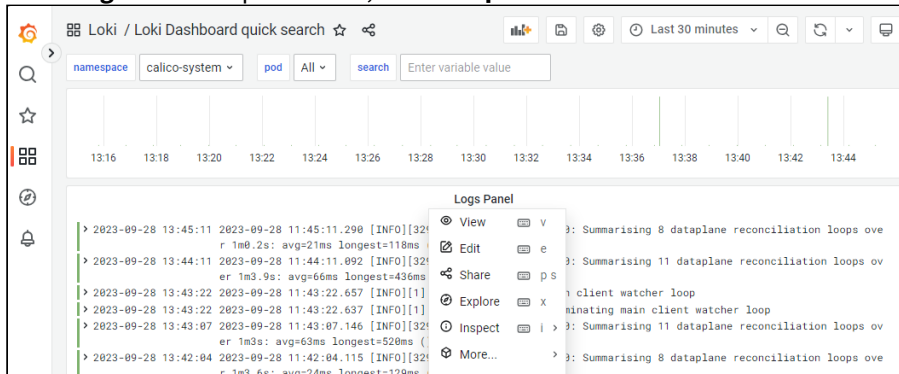
2. In the **Dashboards** page, select **Loki > Loki Dashboard quick search**.



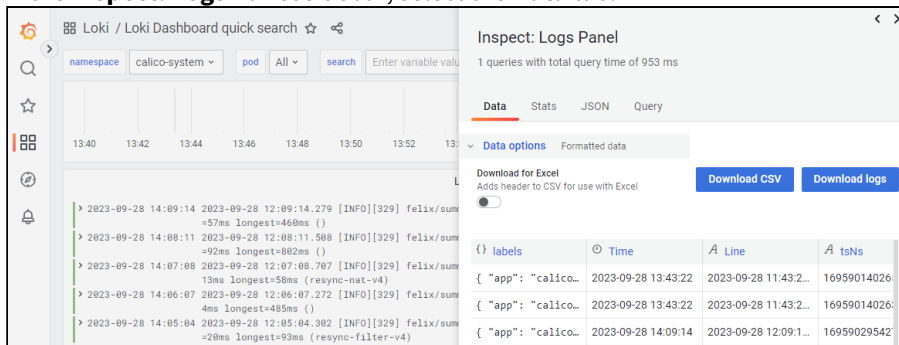
- In the **Locki Dashboard quick search** page, use the **namespace, pod, search, and time range** menus to filter logs.



- In the **Logs Panel** drop-down list, select **Inspect**.



- In the **Inspect: Logs Panel** sidebar, select the **Data** tab.



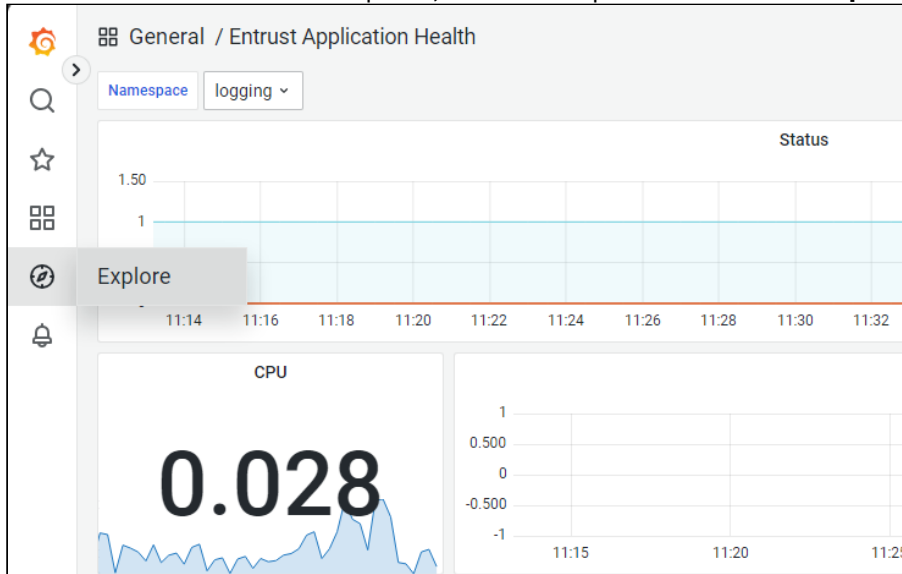
- In the **Data** tab, click:
 - Download CSV** to download the logs into a CSV file.
 - Data options > Download for Excel** and **Download CSV** to download the logs into an Excel-compatible CSV file.
 - Download logs** to download the raw logs into a text file.

Browsing log file contents with Grafana

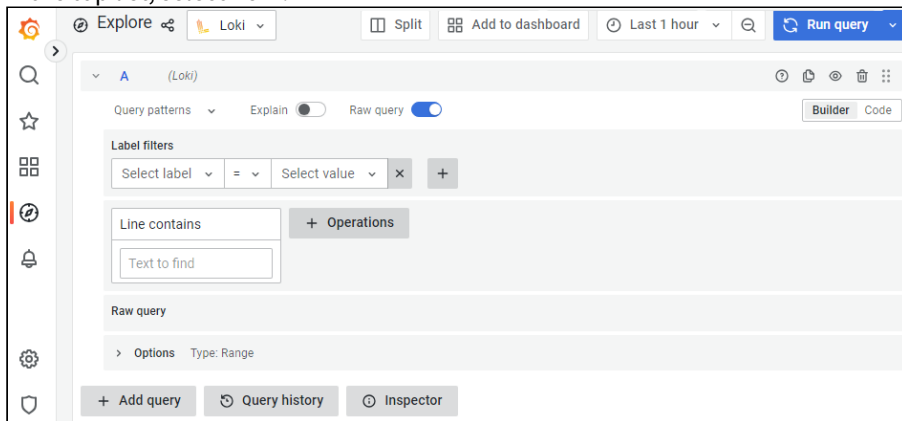
Grafana allows browsing operating system-level logs recorded in files.

To browse log files with Grafana

1. In the left sidebar of the Grafana portal, click the compass icon and select **Explore**.



2. In the top list, select **Loki**.



3. In the **Label filters** list, select **filename**.
4. In the **Select value** field, select a log file.
5. Click **+Operations** to filter the file contents.
6. Click **+Add query** to include the contents of additional files.
7. In the top-right corner, click **Run query** to display the selected contents.

10 Administrating

Entrust PKI Hub supports the administration operations described below.

- [Adding nodes](#)
- [Administrating console users](#)
- [Backing up and restoring the state](#)
- [Checking the etcd database size](#)
- [Checking the persistent volume disk usage](#)
- [Defragmenting the etcd database](#)
- [Managing the retention policies](#)
- [Recovering from disaster](#)
- [Restarting the nodes](#)
- [Updating DNS resolution](#)


See [clusterctl reference](#) for the parameters required by each command.

Adding nodes

When completing the installation of the Entrust PKI Hub cluster in one node, you can add more nodes to the cluster.

To add Entrust PKI Hub nodes


1. Run the [clusterctl node info](#) command to list the nodes already in the cluster.
2. In a node already in Entrust PKI Hub cluster, run the [clusterctl node join-token](#) command to get the joining token.
3. Deploy the [Recommended number of nodes](#).
4. In the new nodes:
 - a. Configure the [Requirements](#).
 - b. Install the product image as explained in [Installing the Entrust PKI Hub image](#).
 - c. Ensure no restriction blocks access to the [Required open ports](#).
 - d. Open a command-line interpreter and run the [clusterctl node add](#) command using the join token obtained in the first step. Do not run this command simultaneously in different nodes of the same Entrust PKI Hub deployment.

 When running this command in AWS or Azure cloud installations, use the private IP of the nodes.

5. Redeploy each deployed Entrust solution (if any) using either the [clusterctl solution deploy](#) command or the Management Console.

Administrating console users

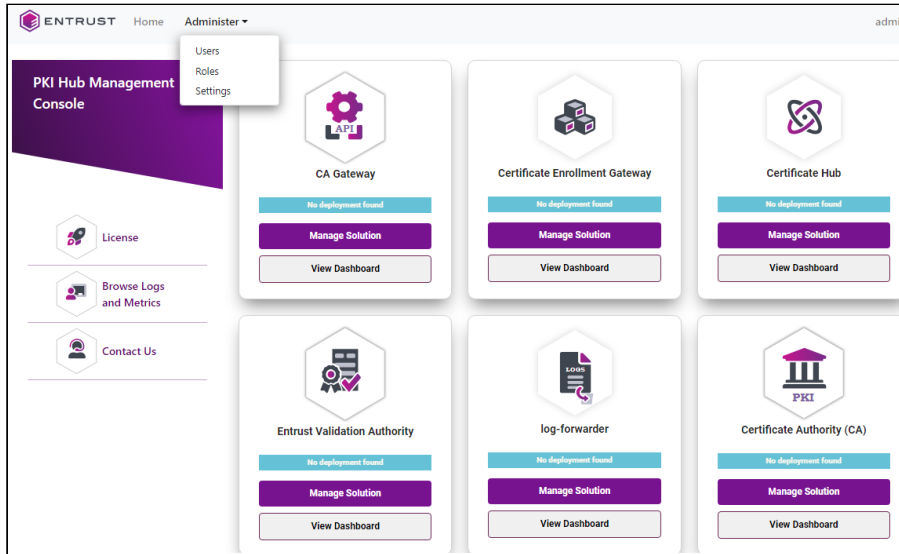
Create and configure the management console users.

 Users with permissions on the operations described in [clusterctl reference](#) are operating system users not related to console users.

To administrate the management console users

1. Login into the Management Console.

2. Click the **Administer** menu.



3. Select the following options.

- [Users](#)
- [Roles](#)
- [Identity provider](#)

Users

Select **Administer > Users** to create, edit, or delete Management Console users. See below for the user settings.

- [Name](#)
- [Email](#)
- [Password](#)
- [Confirm Password](#)
- [Roles](#)

Name

The name of the user. See the table below for the supported format.

Name length	Supported characters
3-13 characters	Lowercase letters, numbers, dashes ("-"), and underscores ("_").

Email

The email address of the user.

Password

The user's password. This password must be at least 8 characters long and complex enough for the **Password Strength** bar to become green.

Confirm Password

Repeat the password.

Roles

The [Roles](#) granted to the user.

Roles

Select **Administer > Roles** to create, edit, or delete roles for the Management Console [Users](#). See the table below for the supported role permissions.

Category	Permission	Operation
Solutions	Manage and Operate Certificate Authorities (CAs)	See Starting up Certificate Authorities
	Manage CA Gateway (CAGW)	See Starting up CA Gateway
	Manage Certificate Enrollment Gateway (CEG)	See Starting up Certificate Enrollment Gateway
	Manage Certificate Hub	See Starting up Certificate Hub
	Manage Entrust Validation Authority (EVA)	See Starting up Entrust Validation Authority
	Manage Timestamping Authority (TSA)	See Starting up Timestamping Authority
Users	Manage Users	See Users
Roles	Manage Roles	The role creation and management operations described in this page
Identity Providers	Manage Identity Providers	See Identity provider

Identity provider

Integrate identity providers already in use in the corporate environment.

To configure an identity provider

1. Go to **Administer > Settings > IDENTITY PROVIDER**.
2. In the **Select Identity Provider** list, choose one of the mechanisms described in the following sections.
 - [Entrust Identity as a Service \(IDaaS\)](#)

- [Internal password](#)
 - [Lightweight Directory Access Protocol](#)
 - [OpenID Connect 1.0](#)
3. Configure the selected identity providers.
 4. Optionally, disable the [Internal Password](#) authentication mechanism.
 5. In **Administer > Administrators**, manage the new IdP-registered administrators.

Entrust Identity as a Service (IDaaS)

In the IDaaS administration interface, configure an OIDC Web application with the following settings.

Setting	Value
Subject Id Attribute	Type a unique user identifier.
ID Token Signing Algorithm	Select RS256 .
Redirect URI(s)	Paste the value of the Redirect URL configuration setting described below.
Supported Scopes > Email address	Mark this checkbox
Require Consent	Unmark this checkbox.
User Info Signing Algorithm	None
Claims	Create a claim with the <code>profile</code> identifier. Set a group name as the value of each claim attribute.
Authentication decision	Select second factors as you wish and ensure users have the required authentications.
Groups	Create one group and add the users with login permissions.

In the Entrust PKI Hub console, configure the following settings for an Entrust Identity as a Service (IDaaS) identity provider.

- [Active](#)
- [Name](#)
- [Redirect URL](#)
- [Client Secret](#)
- [Client ID](#)
- [Base IDaaS URL](#)
- [Required Group Attribute Name](#)
- [Required Group Name](#)
- [JWKS URL](#)
- [Authorization Endpoint](#)

- [Access Token Endpoint](#)
- [UserInfo Endpoint](#)
- [Logout Endpoint](#)

Active

Mark this checkbox to enable the identity provider.

Name

Type a provider name to display when logging into the Entrust PKI Hub console.

Redirect URL

The URL to redirect to when the identity provider successfully authenticates a user. Entrust PKI Hub automatically generates this value when you click **Save**. You must:

1. Copy this value from the Entrust PKI Hub interface.
2. Paste this URL on the **Redirect URI(s)** field of the IDaaS interface.

When the Entrust PKI Hub host URL changes, you must:

1. Re-type the **Client Secret** and **Client ID** values on the Entrust PKI Hub console.
2. Click **Save**.
3. Copy the new **Redirect URL** value from the Entrust PKI Hub console.
4. Paste this URL on the **Redirect URI(s)** field of the IDaaS interface.

Client Secret


Paste the client secret from the IDaaS OIDC application.

Client ID

Paste the client identifier from the IDaaS OIDC application.

Base IDaaS URL

Paste the account URL of the IDaaS OIDC application.

 When you enter this URL, the web browser interface fills in the rest of the URLs.

Required Group Attribute Name

Type the following attribute name.

profile

Required Group Name

Type the name of the group configured in the IDaaS OIDC application.

JWKS URL

Paste the JSON Web Key Set (JWKS) URL of your identity provider. For example:

```
https://asacm.auth0.com/.well-known/jwks.json
```

Authorization Endpoint

Paste the authorization endpoint of your identity provider. For example:

```
https://asacm.auth0.com/authorize
```

Access Token Endpoint

Paste the token endpoint of your identity provider. For example:

```
https://asacm.auth0.com/oauth/token
```

UserInfo Endpoint

Paste the `UserInfo` endpoint of your identity provider. For example:

```
https://asacm.auth0.com/userinfo
```

Logout Endpoint

Paste the logout URL of your identity provider. For example:

```
https://asacm.auth0.com/v2/logout
```

Internal password

The password authentication mechanism for the Web interface-registered administrators. Click the **Force Password Change** button if you want all the administrators to change the password at the next login.

Lightweight Directory Access Protocol

In the Lightweight Directory Access Protocol (LDAP) server, add the following attributes to the authorized users.

Attribute	Value
email	An email address to identify the user uniquely.
memberOf	The name of the group to which the user belongs. You can only omit this attribute if you also omit the Required Group Name setting described below.

In the Entrust PKI Hub console, configure the following settings for an LDAP identity provider.

- [Active](#)
- [LDAP URI](#)
- [User DN Template](#)
- [Required Group Name](#)
- [LDAP SSL CA Bundle \(PEM\)](#)
- [Active Directory](#)
- [Active Directory Email Domain](#)
- [Active Directory Base Lookup DN](#)

Active

Mark this checkbox to enable the identity provider.

LDAP URI

Paste the URI of the LDAP or Active Directory server.

 When the URI does not include the LDAP port, 389 is assumed.

User DN Template

Enter a template for building the user's DN – for example:

```
uid={0},ou=users,dc=abccorp,dc=dev,dc=entrust,dc=com
```

Required Group Name


Enter the value of the `memberOf` LDAP attribute for users with login permissions. Enter the name of an LDAP group, not the full DN – for example, `CorpUser` grants access to members of an LDAP group with the following DN.

```
cn=CorpUser,ou=groups,dc=abccorp,dc=dev,dc=entrust,dc=com
```

Omit this optional field to authorize all LDAP users.

LDAP SSL CA Bundle (PEM)

Paste the certification chain of the LDAP server SSL certificate, as a bundle in PEM format.

 This parameter is mandatory for LDAPS connections when the LDAP server SSL certificate issuer is not a publicly trusted CA.

Active Directory

Mark this checkbox for configuring Active Directory-specific parameters.

Active Directory Email Domain

Enter the domain of the Active Directory email addresses.

Active Directory Base Lookup DN

Enter the root base for searching distinguished names in the Active Directory.

OpenID Connect 1.0

In the Entrust PKI Hub console, configure the following settings for an OpenID Connect 1.0 identity provider.

- [Active](#)
- [Name](#)
- [Redirect URL](#)
- [Client Secret](#)
- [Client ID](#)
- [Required Group Attribute Name](#)
- [Required Group Name](#)
- [JWKS URL](#)
- [Authorization Endpoint](#)
- [Access Token Endpoint](#)
- [UserInfo Endpoint](#)
- [Logout Endpoint](#)

Active

Mark this checkbox to enable the identity provider.

Name

The identity provider name displayed when logging into the Entrust PKI Hub console.

Redirect URL

The URL to redirect to when the identity provider successfully authenticates a user. Entrust PKI Hub automatically generates this value when you click **Save**. You must:

1. Copy this value from the Entrust PKI Hub console.
2. Paste this URL on the redirect URLs field of your IdP interface – for example, on the **Allowed Callback URLs** field of an auth0 identity provider.

When the Entrust PKI Hub host URL changes, you must:

1. Re-type the **Client Secret** and **Client ID** values on the Entrust PKI Hub console.
2. Click **Save**.
3. Copy the new **Redirect URL** value from the Entrust PKI Hub console.
4. Paste this URL on the redirect URLs field of your IdP interface.

Client Secret

The client secret provided by your identity provider.

Client ID

The client identifier provided by your identity provider.

Required Group Attribute Name

The claim name provided by your identity provider for user access restriction. See below an example for Auth0.

```
https://asacm/group
```

This custom claim must start with `https` or `http` and cannot include a dot. For example:

```
function (user, context, callback){
  context.idToken['https://asacm/group'] = user.app_metadata.group;
  callback(null, user, context);
}
```

See <https://auth0.com/docs/scopes/openid-connect-scopes> for how to create a custom claim.

Required Group Name

The claim value provided by your identity provider for user access restriction. See below an example for Auth0 where only users in the "admin" group have access permissions.

```
"app_metadata": {
  "group": "admin"
},
```

JWKS URL

The JSON Web Key Set (JWKS) URL of your identity provider. For example:

```
https://asacm.auth0.com/.well-known/jwks.json
```

Authorization Endpoint

The authorization endpoint of your identity provider. For example:

```
https://asacm.auth0.com/authorize
```

Access Token Endpoint

The token endpoint of your identity provider. For example:

```
https://asacm.auth0.com/oauth/token
```

UserInfo Endpoint

The UserInfo endpoint of your identity provider. For example:

```
https://asacm.auth0.com/userinfo
```

Logout Endpoint

The logout URL of your identity provider. For example:

```
https://asacm.auth0.com/v2/logout
```

Backing up and restoring the state

See below for how to backup and restore the state of an Entrust PKI Hub installation.

- [Backing up the state](#)
- [Restoring the state](#)

Backing up the state

To back up the state of Entrust PKI Hub, run the `clusterctl backup create` command. For example:

```
$ sudo clusterctl backup create --file /home/sysadmin/20230314.bkp --password 7Txsxu
```


In multi-node installations, you must manually copy the data not included in the backup file.


- The key and certificate for TLS
- The registration credentials
- The volume capacity policies configured with the `clusterctl volume capacity` command.
- The retention policies configured with the `clusterctl retention config logs` and `clusterctl retention config metrics` commands.
- The proxy settings configured with the `clusterctl proxy set` command.


 Some Entrust solutions may require additional backup operations. Refer to the solution guide for details.

Restoring the state

See the table below for how to restore the state of an Entrust PKI Hub installation depending on the number of nodes (N).

- When N=1, skip the steps marked .
- When N > 1, perform each step in the nodes indicated.

 As explained in [Running clusterctl install](#), only installations in `multi-node` mode support state restoration, although such installations can indeed use a single node.

#	Step	N=1	N > 1
1	Ensure the machine meets all the Requirements .		Perform in all nodes

#	Step	N=1	N > 1
2	Install the Entrust PKI Hub image as explained in Installing the Entrust PKI Hub image .	✓	Perform in all nodes
3	Set the same hostname and IP address as in the original installation.	✓	Perform in all nodes
4	Install Entrust PKI Hub (on the default multi-node mode) as explained in Running clusterctl install	✗	Perform in a single node
5	Import the license as explained in Setting or updating the license .	✗	Perform in a single node
6	Run the <code>clusterctl node join-token</code> command to get the joining token.	✗	Perform in a single node
7	Add the node to Entrust PKI Hub, as explained in Adding nodes .	✗	Perform in all nodes except the one where the joined token was obtained.
8	Run the <code>clusterctl certificate</code> command to install the TLS certificate and key backup.	✗	Perform in a single node
9	Run the <code>clusterctl volume capacity</code> to restore the previous volume capacity policies.	✗	Perform in a single node
10	Run <code>clusterctl retention config logs</code> to restore the previous log retention period.	✗	Perform in a single node
11	Run <code>clusterctl retention config metrics</code> to restore the previous metric retention period.	✗	Perform in a single node
12	Run the <code>clusterctl proxy set</code> to restore the previous proxy settings.	✗	Perform in a single node
13	Copy the file generated in Backing up the state .	✓	Perform in a single node
14	Run the <code>clusterctl backup restore</code> command to restore the backup.	✓	Perform in a single node

#	Step	N=1	N > 1
15	Redeploy the solutions present in the original installation.	✘	Perform in a single node

Checking the etcd database size

As explained in [Overview](#), Entrust PKI Hub integrates an etcd database. See below for how to monitor the size of this database.

To check the size of the etcd database

1. Browse to the **Etcd > etcd Status** dashboard.
2. Check the information in the **DB Info per Member** section: used space, need for defragmentation, available capacity, etc.
3. If the dashboard alerts that the cluster needs defragmentation, or the etcd used space is close to 100%, perform the operation described in [Defragmenting the etcd database](#).

Checking the persistent volume disk usage

To check the disk usage of the persistent volumes, you can either:

- Login to the Grafana portal, as explained in [Browsing logs with Grafana](#), and browse the **Node Exporter Full** dashboard.
- Run the `clusterctl volume info` command.

Defragmenting the etcd database

See below how to defragment the `etcd` database of one node. In multi-node installations, repeat the procedure sequentially in all the nodes.

⚠ Defragmenting a live etcd member blocks the system from reading and writing data while rebuilding states. This operation can take several seconds, during which the whole cluster node is unavailable.

To defragment the etcd database in one node

1. Get the current etcd revision.

```
rev=$(sudo ETCDCCTL_API=3 /opt/entrust/etcdctl --cert /var/lib/rancher/k3s/server/tls/etcd/client.crt --key /var/lib/rancher/k3s/server/tls/etcd/client.key --cacert /var/lib/rancher/k3s/server/tls/etcd/server-ca.crt endpoint status --write-out="json" | egrep -o '"revision":[0-9]*' | egrep -o '[0-9].*')
```

2. Compact away all old `etcd` revisions.

```
sudo ETCDCCTL_API=3 /opt/entrust/etcdctl --cert /var/lib/rancher/k3s/server/tls/etcd/client.crt --key /var/lib/rancher/k3s/server/tls/etcd/client.key --cacert /var/lib/rancher/k3s/server/tls/etcd/server-ca.crt compact $rev
```

3. Defragment `etcd` on the node.

```
sudo ETCDCTL_API=3 /opt/entrust/etcdctl --cert /var/lib/rancher/k3s/server/tls/etcd/client.crt --key /var/lib/rancher/k3s/server/tls/etcd/client.key --cacert /var/lib/rancher/k3s/server/tls/etcd/server-ca.crt defrag
```

4. Disarm the `etcd` database space alarm.

```
sudo ETCDCTL_API=3 /opt/entrust/etcdctl --cert /var/lib/rancher/k3s/server/tls/etcd/client.crt --key /var/lib/rancher/k3s/server/tls/etcd/client.key --cacert /var/lib/rancher/k3s/server/tls/etcd/server-ca.crt alarm disarm
```

Managing the retention policies

The default retention policy for logs and metrics is the following.

Data	Days	Default storage	Record deletion
Logs	28	10 GB	After 28 days or when reaching 75% of the storage
Metrics	14	10GB	After 14 days or when reaching 80% of the storage


As explained in [clusterctl reference](#), you can manage the retention settings with the following commands.

- [clusterctl retention config logs](#)
- [clusterctl retention config metrics](#)
- [clusterctl retention info](#)

Recovering from disaster

Perform the following steps to recover Entrust PKI Hub and the deployed solutions after a system crash.

- [Recovering single-node installations](#)
- [Recovering multi-node installations with a quorum](#)
- [Recovering multi-node installations without a quorum](#)

 As explained in [Running clusterctl install](#), only installations in `multi-node` mode support disaster recovery, although such installations can indeed use a single node.

Recovering single-node installations

When a single-node installation crashes, recover the Entrust PKI Hub and the deployed solutions from a backup as explained in [Restoring the state](#).

Recovering multi-node installations with a quorum

When your multi-node installation retains the quorum described in [Recommended number of nodes](#), you must simply remove, restore and add the crashed nodes.

To recover a crashed node.

1. Mark the node as unschedulable.

```
sudo /opt/entrust/kubectl cordon <node-to-delete>
```

2. Drain the pods from the node to delete.

```
sudo /opt/entrust/kubectl drain <node-to-delete> --delete-emptydir-data --  
disable-eviction --force --ignore-daemonsets --timeout=600s
```

3. Delete the node from the cluster.

```
sudo /opt/entrust/kubectl delete node <node-to-delete> --timeout=600s
```

4. Restore the node and add it again to the cluster, as explained in [Adding nodes](#).

Recovering multi-node installations without a quorum

When your multi-node installation does not retain the quorum described in [Load balancing requirements](#), follow the steps below for recovery.

To recover a multi-node deployment without a quorum

1. Run `clusterctl uninstall` in all the nodes to uninstall Entrust PKI Hub.
2. Recover the Entrust PKI Hub and the deployed solutions from a backup as explained in [Restoring the state](#).

Restarting the nodes

Some Entrust PKI Hub upgrades may require restarting the system. In multi-node installations, restart the nodes:

- One by one.
- At least 15 minutes apart.

Otherwise, you might encounter system problems and risk losing the quorum described in [Recommended number of nodes](#).

After restarting a node, run the following command to force the start of the `chrony` service.

```
sudo systemctl restart chronyd.service
```

Updating DNS resolution

To update DNS resolution after installing Entrust PKI Hub, repeat the following instructions on each node.

✘ As explained in [DNS requirements](#), Entrust PKI Hub does not support accessing a DNS server through a proxy.

Platform	Instructions
VMware vSphere or physical machine	See the Configuring the connection of a PKI Hub ISO installation section of this guide.
Amazon Web Services	Refer to the docs.aws.amazon.com product documentation.
Azure	Refer to the learn.microsoft.com/azure product documentation.

When completing the DNS update, run the following command to restart the `coredns` service.

```
sudo kubectl rollout restart deployment coredns -n kube-system
```


! If an option value contains spaces or special characters, you must surround the entire value with quotation marks.

-f, --file <file>

Export the backup in `<file>`, where `<file>` is the full path of a file with the `.bkp` extension.

Mandatory: Yes.

-p, --password <pwd>

Protect the backup file with the `<pwd>` password.

Mandatory: No. When omitting this option, the command prompts for the password and password confirmation.

clusterctl backup restore

Executed when: [Restoring the state](#).

Restores the state of Entrust PKI Hub.

```
clusterctl backup restore --file <file> [--password <pwd>]
```

See below for a description of each parameter.

- [-f, --file <file>](#)
- [-p, --password <pwd>](#)

For example:

```
$ sudo clusterctl backup restore --file /home/edm/20211009.bkp
```

Specifically, this command:

1. Restores the state of Entrust PKI Hub along with the deployed Entrust solutions.
2. In physical and VMware machines, enables `firewalld` and creates firewall rules for opening the [Required open ports](#).

The command raises an exception when executed in a node with:

- A different hostname than the backed-up node.
- Entrust PKI Hub already installed.

! If an option value contains spaces or special characters, you must surround the entire value with quotation marks.

-f, --file <file>

Restore the `<file>` backup, where `<file>` is the path of a file generated with the `clusterctl backup create` command.

Mandatory: Yes.

-p, --password <pwd>

Open the backup file with the `<pwd>` password, where `<pwd>` is the password passed to the `clusterctl backup create` command when exporting the backup.

Mandatory: No. When omitting this option, the command prompts for the password.

clusterctl certificate

Installs the TLS certificate of Entrust PKI Hub.

```
clusterctl certificate --cert <bundle> --key <key>
```

See below for a description of each parameter.

- `-c, --cert <bundle>`
- `-k, --key <key>`

For example:

```
$ sudo clusterctl certificate --cert /home/sysadmin/cert.pem --key /home/sysadmin/key.pem
```


If the Certificate Authorities solution is already deployed, redeploy the solution to make the changes effective.

```
$ clusterctl solution deploy --solution-id pkihub
```

-c, --cert <bundle>

Load the TLS certificate and the certification chain from `<bundle>`, where `<bundle>` is the path of a plaintext file in the following PEM format.

```
-----BEGIN CERTIFICATE-----  
<TLS Server cert in B64 encoding>  
-----END CERTIFICATE-----  
-----BEGIN CERTIFICATE-----  
<CA Subordinate Cert in B64 encoding>  
-----END CERTIFICATE-----
```

 The selected certificate must meet the requirements described in [Replacing the default TLS certificate](#).

Mandatory: Yes.

1. Validates the [Requirements](#).
2. Enables `firewalld` and creates firewall rules for opening the [Required open ports](#).
3. Installs Entrust PKI Hub.
4. Generates a self-signed TLS certificate (that you can optionally replace during the product configuration).
5. Records any installation error in the `/var/log/entrust/edm/clusterctl.log` file.

--mode <mode>

Run the installation in `<mode>` mode. Where `<mode>` is either:

- single-node
- multi-node

See the table below for a description of each mode.

Setting	single-node	multi-node
Deployment type	Proof-of-concept deployments.	Production deployments.
Requirements	Does not need the disk performance requirements described in Disk requirements . Specifically, fsync latency is not an issue in this mode.	All the Requirements .
Supported number of nodes	One	One or more. See Recommended number of nodes for details.
Supported operations	You cannot perform the operations described in Adding nodes , Backing up the state , Recovering from disaster , or Restoring the state .	All
Supported updates	You cannot upgrade to a newer version or migrate to a multi-node installation.	All

Mandatory: No. This optional value defaults to `multi-node`.

clusterctl license import

Executed when: [Setting or updating the license](#).

Sets or updates the Entrust PKI Hub license.

```
clusterctl license import -f <license>
```

Where `<license>` is the path of the license file. For example:

clusterctl node info

Lists the nodes of Entrust PKI Hub.

```
clusterctl node info
```

For example:

```
$ sudo clusterctl node info
NODE NAME                                NODE IP      NODE AGE
myhost01.mydomain.com                    10.1.141.50  5h57m
myhost02.mydomain.com                    10.1.141.51  4h42m
myhost03.mydomain.com                    10.1.141.52  4h11m
```

Where:

- `NODE NAME` is the node name.
- `NODE IP` is the node's IP.
- `NODE AGE` is the time during which the node has been in operation.

If Entrust PKI Hub runs on less than 3 nodes, the information includes a warning message because HA requires at least 3 nodes (we recommend an odd number of nodes).

clusterctl node join-token


Executed when: [Adding nodes](#) to Entrust PKI Hub.

Prints the joining token for other nodes to join the cluster.

```
clusterctl node join-token
```

For example:

```
$ sudo clusterctl node join-token
NODE JOIN TOKEN
7dc820922681be2197736246ef32784a530186c3e58e07ac671768de690c3ca0
```

 You must run this command in a node already added to Entrust PKI Hub.

clusterctl proxy clear


Executed when: [Configuring the proxy](#).

Clears the proxy configuration.

```
clusterctl proxy clear
```

For example:

```
$ sudo clusterctl proxy clear
The proxy configuration has been cleared
```

 To make configuration changes effective, run the [clusterctl solution deploy](#) command and redeploy all deployed solutions.

clusterctl proxy info

Executed when: [Configuring the proxy](#).

Prints the current proxy configuration.

```
clusterctl proxy info
```

For example:

```
$ sudo clusterctl proxy info
Protocol:      https
Host:         myproxy
Port:         443
User:         bob
Password:     ****
TLS server cert:  None
```

See [clusterctl proxy set](#) for how to set these parameters.

clusterctl proxy set

Executed when: [Configuring the proxy](#).

Configures the connection with the proxy server.

```
clusterctl proxy set --host <host> [--port <port>] [--protocol <protocol>] [--
certificate <cert>] [--user <user>] [--password <pwd>]
```

See below for a description of each parameter.

- `--host <host>`
- `--port <port>`
- `--user <user>, --password <pwd>`
- `--protocol <protocol>, --certificate <cert>`

For example:

```
$ sudo clusterctl proxy set --host myproxy --user bob --password demo
The proxy configuration has been set
```

! To make configuration changes effective, run the `clusterctl solution deploy` command and redeploy all deployed solutions.

`--host <host>`

Configure the `<host>` proxy, where `<host>` is the IP or hostname of the proxy.

Mandatory: Yes.

`--port <port>`

Configure `<port>` as the proxy port number.

Mandatory: No. When omitted, this parameter defaults to 443.

`--user <user>, --password <pwd>`

Authenticate in the proxy with the `<user>` username and the `<pwd>` password.

Mandatory: No. See the following table for the supported combinations of the `--user` and `--password` options.

<code>--user</code>	<code>--password</code>	Action
<code><user></code>	<code><pwd></code>	Set <code><user></code> and <code><pwd></code> as proxy basic authentication credentials.
<code><user></code>		Prompt for the password of the <code><user></code> user.
	<code><pwd></code>	Throw an error.
		Assume that the proxy requires no authentication.

`--protocol <protocol>, --certificate <cert>`

Authenticate in the proxy with:

- The `<protocol>` protocol, where `<protocol>` is either `http` or `https`.
- The `<cert>` certificate, where `<cert>` is the path of a plaintext file in PEM format. The certificate should include a SAN matching the host.

Mandatory: No. See the following table for the supported combinations of the `--protocol` and the `--certificate` options.

--protocol	--certificate	Action
http	<cert>	Throw an error.
http		Set an HTTP connection.
https	<cert>	Set an HTTPS connection with the <cert> certificate.
https		Set an HTTPS connection with the default certificate of the operating system.
	<cert>	Set an HTTPS connection with the <cert> certificate.
		Set an HTTPS connection with the default certificate of the operating system.

clusterctl retention config logs

Executed when: [Browsing logs with Grafana](#).

Updates the log retention period.

```
clusterctl retention config logs --period <period> [-y]
```

See below for a description of each parameter.


- `--period <period>`
- `-y, --yes`

For example, to set a 15-days retention period.

```
$ sudo clusterctl retention config logs --period 15
Warning: The command execution will temporarily interrupt the selected service. This
might take several minutes
Are you sure you want to modify the retention values? [Y/n]: Y
Configuring logs retention period to 15 days... Done
```

`--period <period>`

Set the retention period to `<period>`, where `<period>` is a number of days.

 A `<period>` value lower than the current period deletes all logs older than `<period>`.

Mandatory: Yes.

-y, --yes

Skip the confirmation prompt.

clusterctl retention config metrics

Executed when: [Browsing logs with Grafana](#).

Updates the metrics retention period, the metrics allocated space, or both.

✘ This command temporarily interrupts the metrics service.

```
clusterctl retention config metrics --period <period> --size <size> [-y]
```

See below for a description of each parameter.

- `--period <period>`
- `--size <size>`
- `-y, --yes`

For example, to update both the retention period and allocated space:

```
$ sudo clusterctl retention config metrics --period 15 --size 2
Warning: The command execution will temporarily interrupt the selected service. This
might take several minutes
Are you sure you want to modify the retention values? [Y/n]: Y
Configuring metrics retention period to 15 days... Done
Configuring metrics retention size to 2Gi... Done
```

To update only the retention size:

```
$ sudo clusterctl retention config metrics --size 2
Warning: The command execution will temporarily interrupt the selected service. This
might take several minutes
Are you sure you want to modify the retention values? [Y/n]: Y
Configuring metrics retention size to 2Gi... Done
```

✘ The `--period` and `--size` options are only mutually exclusive when set to 0. When set to a value greater than 0, you can pass both options or only one.

`--period <period>`

Set the retention period to `<period>`, where `<period>` is a number of days.

- A `<period>` value lower than the current period deletes all metrics older than `<period>`.
- A `<period>` value of 0 disables the retention period so the metrics retention is only limited by `<size>`.

Mandatory: No. When omitting this option, the retention period remains unchanged.

--size <size>

Allocate <size> for the metrics, there <size> is a number of Gi allocated for the metrics (float values supported).

- A <size> value lower than the current size deletes all metrics exceeding <size> .
- A <size> value of 0 disables the allocated space limit, so the metrics retention is only limited by <period> .

⚠ When updating the allocated size, run the [clusterctl volume capacity](#) command to update the Prometheus volume size accordingly.

Mandatory: No. When omitting this option, the metrics space remains unchanged.

-y, --yes

Skip the confirmation prompt.

clusterctl retention info

Executed when: [Browsing logs with Grafana](#).

Prints the current retention settings.

```
clusterctl retention info
```

For example:

```
$ sudo clusterctl retention info
SERVICE      PERIOD  MAX SIZE
metrics       14d    1GB
logs          28d    N/A
```

Where:

- **SERVICE** is the service name: metrics or logs.
- **PERIOD** is the retention period for the logs or metrics.
- **MAX SIZE** is the maximum size allocated for the metrics.

See [clusterctl retention config logs](#) and [clusterctl retention config metrics](#) for updating these retention settings.

clusterctl solution config export

Exports the configuration of an installed Entrust solution.

```
clusterctl solution config export --solution-id <solution_id> --path <dir_path>
```

See below for a description of each parameter.

- `-i, --solution-id <solution_id>`
- `-f, --path <dir_path>`

For example:


```
$ sudo clusterctl solution config export --solution-id eva --path /home/sysadmin/eva/
config
Exporting the configuration files... Done
```

The command will raise an error if:

- The `<dir_path>` folder does not exist in the node where you run the command.
- The `<dir_path>` folder is not empty.
- The Entrust solution is not correctly registered and deployed.

`-i, --solution-id <solution_id>`

Export the configuration of the solution with the `<solution_id>` identifier.

 Run the `clusterctl solution info` command to get the identifiers of the installed Entrust solutions.

Mandatory: Yes.

`-f, --path <dir_path>`

Export the configuration in `<dir_path>`, where `<dir_path>` is the path of an empty folder in the execution node.

Mandatory: Yes.

clusterctl solution config import

Executed when: Starting up and deploying solutions with clusterctl.

Imports the configuration of an installed Entrust solution.

```
clusterctl solution config import --solution-id <solution_id> --path <dir_path>
```

See below for a description of each parameter.


- `-i, --solution-id <solution_id>`
- `-f, --path <dir_path>`

For example:

```
$ sudo clusterctl solution config import --solution-id eva --path /home/sysadmin/eva/
config
Importing the configuration files... Done
```


The command will raise an error if:

- The `<dir_path>` folder does not exist in the node where you run the command.
- The `<dir_path>` folder is empty.
- The Entrust solution is not correctly registered and deployed.

 To make configuration changes effective, redeploy the solution with the [clusterctl solution deploy](#) command.

`-i, --solution-id <solution_id>`

Import the configuration of the solution with the `<solution_id>` identifier.


 Run the [clusterctl solution info](#) command to get the identifiers of the installed Entrust solutions.

Mandatory: Yes.

`-f, --path <dir_path>`

Import all configuration files that are included in the folder with the `<dir_path>` path and have filenames matching the following regular expression.

```
^[a-zA-Z0-9/\\. _-]{3,64}$
```

 The command removes from the solution any other configuration file. Therefore, always run the [clusterctl solution config import](#) command beforehand to ensure you have the complete set of configuration files.

Mandatory: Yes.

`clusterctl solution deploy`


Executed when: Deploying solutions with clusterctl.

Deploys an Entrust solution or redeploys an already deployed Entrust solution to make configuration changes effective.

```
clusterctl solution deploy --solution-id <solution_id>
```

`-i, --solution-id <solution_id>`

Deploy the solution with the `<solution_id>` identifier.

 Run the [clusterctl solution info](#) command to get the identifiers of the installed Entrust solutions.

Mandatory: Yes.

clusterctl solution info

Prints the main settings of each deployed Entrust solution.

```
clusterctl solution info
```

For example:

```
$ sudo clusterctl solution info
SOLUTION ID   DEPLOYED
ceg           1.4 at 2022-04-18T23:45:10.000Z (success)
certhub       1.1 at 2022-07-11T10:49:31.000Z (success)
eva           1.0 at 2022-06-11T13:40:57.000Z
cagw          processing
```

See below for a description of each column.

- [SOLUTION ID](#)
- [DEPLOYED](#)

SOLUTION ID

The solution identifier.

SOLUTION ID	Solution
cagw	CA Gateway
certhub	Certificate Hub
ceg	Certificate Enrollment Gateway
eva	Entrust Validation Authority
tsa	Timestamping Authority

DEPLOYED

Information on the last solution deployment, in the following format.

```
<version> at <date> (<status>)
```

The value of `<version>` is the deployed solution version. The values of `<date>` and `<status>` depend on the deployment status.

Deployment status	<status>	<date>
In progress	deploying	The UTC date when the deployment started.
Successfully completed	success	The UTC date when the deployment ended.
Failed	failure	The UTC date when the deployment started.

clusterctl solution secret set

Sets the value of a secret.


```
clusterctl solution secret set -i <solution_id> --name <secret_id> [--from-literal <secret_id>=<secret_value> | --from-file <secret_id>=<secret_path>]
```

See below for a description of each parameter.

- `-i,--solution-id <solution_id>`
- `--from-literal <secret_id>=<secret_value>`
- `--from-file <secret_id>=<secret_path>`


For example:

```
$ sudo clusterctl solution secret set -i ceg --from-literal password='S!B\*d$zDsb='
```

 If already deployed, redeploy the solution with the [clusterctl solution deploy](#) command to make changes effective.

`-i,--solution-id <solution_id>`


Set the secret in the solution with the `<solution_id>` identifier.

 Run the [clusterctl solution info](#) command to get the identifiers of the installed Entrust solutions.

Mandatory: Yes.

`--from-literal <secret_id>=<secret_value>`

Set the secret value from an inline value.


 This option is mutually exclusive with `--from-path`. When omitting both options, the command assumes the `--from-literal` option and prompts the user for `<secret_id>` and `<secret_value>`.

Parameter	Value	Format restrictions
<secret_id>	The identifier of the secret	3-128 characters in length, start with a letter, and contain only alphanumeric characters, underscores, and hyphens.
<secret_value>	The value of the secret	Cannot exceed 128KB. Must be enclosed in quotes to escape special characters (such as \$, \ , * , = , and !) with a backlash (\) as in the above example.

Mandatory: No.

`--from-file <secret_id>=<secret_path>`


Set the secret value from a file.

 This option is mutually exclusive with `--from-literal`. When omitting both options, the command assumes the `--from-literal` option and prompts the user for `<secret_id>` and `<secret_value>`.

Parameter	Value	Format restrictions
<secret_id>	The identifier of the secret	3-128 characters in length, start with a letter, and contain only alphanumeric characters, underscores, and hyphens.
<secret_path>	The path of a file containing the Base64 encoding of the secret value	Cannot exceed 128KB. Must be enclosed in quotes to escape special characters (such as \$, \ , * , = , and !) with a backlash (\) as in the above example.

Mandatory: No.

clusterctl solution upload

 Execute this command only when instructed by customer support to apply solution hotfixes.

Uploads a solution to the Management Console endpoint.

```
clusterctl solution upload --solution-id <solution_id> --file <sln>
```

See below for a description of each parameter.

- `-i, --solution-id <solution_id>`
- `-f, --file <sln>`


```
clusterctl upgrade --iso-path <iso>
```

For example:

```
$ clusterctl upgrade --iso-path /home/sysadmin/edm-1.0.2.iso
```

-f, --iso-path <iso>

Apply the `<iso>` upgrade, where `<iso>` is the path of the ISO upgrade file.

Mandatory: Yes.

clusterctl version

Prints the Entrust PKI Hub version.

```
clusterctl version
```

For example, after installing or upgrading Entrust PKI Hub to version 1.0.0.


```
$ sudo clusterctl version
Entrust PKI Hub release 1.0.0
Installation mode: multi-node
```

When Entrust PKI Hub is not installed, the command specifies "cluster not installed" in the installation mode.

```
$ sudo clusterctl version
Entrust PKI Hub release 1.0.0
Installation mode: cluster not installed
```

clusterctl volume capacity

Sets the size capacity of a volume.

 This command temporarily interrupts the associated service for up to several minutes.

```
clusterctl volume capacity <pv> --size <size> [-y]
```

See below for a description of each parameter.

- `<pv>`
- `--size <size>`
- `-y, --yes`

When prompted, confirm the size modification – for example:

```
$ sudo clusterctl volume capacity storage-entitlements-service --size 2
Warning: The command execution will temporarily interrupt the service associated to
the Persistent Volume. This might take several minutes
Are you sure you want to modify the Persistent Volume capacity? [Y/n]: Y
Resizing the storage-entitlements-service Persistent Volume to 2Gi... Done
```

<pv>

Set the capacity of the <pv> persistent volume, where <pv> is one of the volume names listed by the [clusterctl volume info](#) command.

Mandatory: Yes.

--size <size>

Set the persistent volume size to <size>, where <size> is a number of Gi (float values supported).

- If this value does not exceed the current volume size, the command will throw an error message. Run the [clusterctl volume info](#) command to check the current volume sizes.
- When changing the size capacity of the Prometheus volume, run the [clusterctl retention config metrics](#) command to update the metrics retention accordingly.

Mandatory: Yes.

-y, --yes

Skip the confirmation prompt.

clusterctl volume info

Prints information on the persistent volumes of Entrust PKI Hub.

```
clusterctl volume info
```

For example:

```
$ sudo clusterctl volume info
NAME                CAPACITY  USAGE    % USED
ceg-config-rwx-pvc  93.0MB    1.0MB    1.08%
ceg-db-rwx-pvc      976.0MB   1.0MB    0.20%
storage-auth-service-0  975.9MB   2.7MB    0.27%
storage-csf-grafana-0  3.9GB     18.7MB   0.47%
storage-csf-loki-0    9.8GB     209.3MB  1.09%
storage-docker-registry  19.6GB    4.1GB    21.16%
storage-prometheus-prometheus-0  9.8GB     5.9GB    60.07%
storage-solution-manager-0  9.8GB     36.2MB   0.36%
```

13 CIS benchmarks

The Entrust PKI Hub third-party software is hardened to meet the following recommendations.


- [Linux CIS benchmarks](#)
- [Password policy CIS benchmarks](#)
- [Kubernetes CIS benchmarks](#)

























Linux CIS benchmarks

The Entrust PKI Hub operating system is hardened to meet the following recommendations.

- **Document:** CIS Red Hat Enterprise Linux 8 Benchmark v1.0.0
- **Profile:** Level 1 - Server

Specifically, this operating system meets all recommendations marked  in the following table.

 The **ISO**, **Raw**, and **VHD** columns refer to the available file formats for [Installing the Entrust PKI Hub image](#).

CIS recommendation	Description	ISO	Raw	VHD
1.1.2.1	Ensure <code>/tmp</code> is a separate partition			
1.1.2.2	Ensure <code>nodev</code> option set on <code>/tmp</code> partition			
1.1.2.3	Ensure <code>noexec</code> option set on <code>/tmp</code> partition			
1.1.2.4	Ensure <code>nosuid</code> option set on <code>/tmp</code> partition			
1.1.3.2	Ensure <code>nodev</code> option set on <code>/var</code> partition			
1.1.3.3	Ensure <code>noexec</code> option set on <code>/var</code> partition			
1.1.3.4	Ensure <code>nosuid</code> option set on <code>/var</code> partition			
1.1.4.2	Ensure <code>nodev</code> option set on <code>/var/tmp</code> partition			

CIS recommendation	Description	ISO	Raw	VHD
1.1.4.3	Ensure <code>noexec</code> option set on <code>/var/tmp</code> partition	✓	✗	✗
1.1.4.4	Ensure <code>nosuid</code> option set on <code>/var/tmp</code> partition	✓	✗	✗
1.1.5.2	Ensure <code>nodev</code> option set on <code>/var/log</code> partition	✓	✗	✗
1.1.5.3	Ensure <code>noexec</code> option set on <code>/var/log</code> partition	✓	✗	✗
1.1.5.4	Ensure <code>nosuid</code> option set on <code>/var/log</code> partition	✓	✗	✗
1.1.6.2	Ensure <code>nodev</code> option set on <code>/var/log/audit</code> partition	✓	✗	✗
1.1.6.3	Ensure <code>noexec</code> option set on <code>/var/log/audit</code> partition	✓	✗	✗
1.1.6.4	Ensure <code>nosuid</code> option set on <code>/var/log/audit</code> partition	✓	✗	✗
1.1.7.2	Ensure <code>nodev</code> option set on <code>/home</code> partition	✓	✗	✗
1.1.7.3	Ensure <code>nosuid</code> option set on <code>/home</code> partition	✓	✗	✗
1.3.1	Ensure AIDE is installed	✗	✗	✗
1.3.2	Ensure filesystem integrity is regularly checked	✗	✗	✗
1.4.1	Ensure bootloader password is set	✓	✗	✗

CIS recommendation	Description	ISO	Raw	VHD
1.6.1.6	Ensure no unconfined services exist	✗	✗	✗
3.2.1	Ensure IP forwarding is disabled	✗	✗	✗
3.3.1	Ensure source routed packets are not accepted	✗	✗	✗
3.3.2	Ensure ICMP redirects are not accepted	✗	✗	✗
3.3.9	Ensure IPv6 router advertisements are not accepted	✗	✗	✗
3.4.1.4	Ensure <code>firewalld</code> service enabled and running	✓	✓	✓
3.4.1.5	Ensure <code>firewalld</code> default zone is set	✓	✗	✗
3.4.3.3.3	Ensure <code>ip6tables</code> firewall rules exist for all open ports	✗	✗	✗
5.5.1	Ensure password creation requirements are configured	✓	✓	✓
6.1.2	Ensure sticky bit is set on all world-writable directories	✗	✗	✗
6.1.11	Ensure no world writable files exist	✗	✗	✗
6.1.12	Ensure no unowned files or directories exist	✗	✗	✗
6.1.13	Ensure no ungrouped files or directories exist	✗	✗	✗

Password policy CIS benchmarks

On the first login, you will be asked to replace the initial password of the Entrust PKI Hub administrator.

Name	Initial password
sysadmin	changeme

As per the 5.5.1 benchmark listed in [Linux CIS benchmarks](#), the new password:

- Must consist of at least 14 characters.
- Cannot be based on a dictionary word.
- Must contain at least one uppercase character.
- Must contain at least one lowercase character.
- Must contain at least one digit.
- Must contain at least one special character (for example, punctuation).

Once set, Entrust PKI Hub user passwords have the following settings.




Setting	Value
Password validity	365 days
Grace period before disabling the user account after password expiry	30 days
Minimum time between password changes	7 days
Shell session inactivity timeout	900 seconds

Kubernetes CIS benchmarks

The Entrust PKI Hub K3s (Lightweight Kubernetes) is hardened to meet the following recommendations.

- **Document:** CIS Kubernetes Benchmark v1.6.0
- **Profiles:** Level 2 - Master Node and Level 2 - Worker Node

Specifically, the Entrust PKI Hub K3s meets all recommendations marked  in the following table.

CIS recommendation	Description	Compliance
1.1.1	Ensure that the API server pod specification file permissions are set to 644 or more restrictive	
1.1.2	Ensure that the API server pod specification file ownership is set to root:root	
1.1.3	Ensure that the controller manager pod specification file permissions are set to 644 or more restrictive	

CIS recommendation	Description	Compliance
1.1.4	Ensure that the controller manager pod specification file ownership is set to root:root	✓
1.1.5	Ensure that the scheduler pod specification file permissions are set to 644 or more restrictive	✓
1.1.6	Ensure that the scheduler pod specification file ownership is set to root:root	✓
1.1.7	Ensure that the etcd pod specification file permissions are set to 644 or more restrictive	✓
1.1.8	Ensure that the etcd pod specification file ownership is set to root:root	✓
1.1.9	Ensure that the Container Network Interface file permissions are set to 644 or more restrictive	✓
1.1.10	Ensure that the Container Network Interface file ownership is set to root:root	✓
1.1.11	Ensure that the etcd data directory permissions are set to 700 or more restrictive if etcd is used	✓
1.1.12	Ensure that the etcd data directory ownership is set to etcd:etcd if etcd is used	✓
1.1.13	Ensure that the admin.kubeconfig file permissions are set to 644 or more restrictive	✓
1.1.14	Ensure that the admin.kubeconfig file ownership is set to root:root	✓
1.1.15	Ensure that the scheduler.kubeconfig file permissions are set to 644 or more restrictive	✓
1.1.16	Ensure that the scheduler.kubeconfig file ownership is set to root:root	✓
1.1.17	Ensure that the cloud-controller.kubeconfig file permissions are set to 644 or more restrictive	✓

CIS recommendation	Description	Compliance
1.1.18	Ensure that the <code>/var/lib/rancher/k3s/server/cred/cloud-controller.kubeconfig</code> file ownership is set to <code>root:root</code>	✓
1.1.19	Ensure that the Kubernetes PKI directory and file ownership is set to <code>root:root</code>	✓
1.1.20	Ensure that the Kubernetes PKI certificate file permissions are set to 644 or more restrictive	✓
1.1.21	Ensure that the Kubernetes PKI key file permissions are set to 600	✓
1.2.1	Ensure that the <code>--anonymous-auth</code> argument is set to <code>false</code>	✓
1.2.2	Ensure that the <code>--basic-auth-file</code> argument is not set	✓
1.2.3	Ensure that the <code>--token-auth-file</code> parameter is not set	✓
1.2.4	Ensure that the <code>--kubelet-https</code> argument is set to <code>true</code>	✓
1.2.5	Ensure that the <code>--kubelet-client-certificate</code> and <code>--kubelet-client-key</code> arguments are set as appropriate	✓
1.2.6	Ensure that the <code>--kubelet-certificate-authority</code> argument is set as appropriate	✓
1.2.7	Ensure that the <code>--authorization-mode</code> argument is not set to <code>AlwaysAllow</code>	✓
1.2.8	Ensure that the <code>--authorization-mode</code> argument includes <code>Node</code>	✓
1.2.9	Ensure that the <code>--authorization-mode</code> argument includes <code>RBAC</code>	✓
1.2.10	Ensure that the admission control plugin <code>EventRateLimit</code> is set	✗

CIS recommendation	Description	Compliance
1.2.11	Ensure that the admission control plugin AlwaysAdmit is not set	✓
1.2.12	Ensure that the admission control plugin AlwaysPullImages is set	✗
1.2.13	Ensure that the admission control plugin SecurityContextDeny is set if PodSecurityPolicy is not used	✓
1.2.14	Ensure that the admission control plugin ServiceAccount is set	✗
1.2.15	Ensure that the admission control plugin NamespaceLifecycle is set	✓
1.2.16	Ensure that the admission control plugin PodSecurityPolicy is set	✓
1.2.17	Ensure that the admission control plugin NodeRestriction is set	✓
1.2.18	Ensure that the --insecure-bind-address argument is not set	✓
1.2.19	Ensure that the --insecure-port argument is set to 0	✗
1.2.20	Ensure that the --secure-port argument is not set to 0	✓
1.2.21	Ensure that the --profiling argument is set to false	✓
1.2.22	Ensure that the --audit-log-path argument is set	✓
1.2.23	Ensure that the --audit-log-maxage argument is set to 30 or as appropriate	✓
1.2.24	Ensure that the --audit-log-maxbackup argument is set to 10 or as appropriate	✓

CIS recommendation	Description	Compliance
1.2.25	Ensure that the --audit-log-maxsize argument is set to 100 or as appropriate	✓
1.2.26	Ensure that the --request-timeout argument is set as appropriate	✓
1.2.27	Ensure that the --service-account-lookup argument is set to true	✗
1.2.28	Ensure that the --service-account-key-file argument is set as appropriate	✓
1.2.29	Ensure that the --etcd-certfile and --etcd-keyfile arguments are set as appropriate	✓
1.2.30	Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as appropriate	✓
1.2.31	Ensure that the --client-ca-file argument is set as appropriate	✓
1.2.32	Ensure that the --etcd-cafile argument is set as appropriate	✓
1.2.33	Ensure that the --encryption-provider-config argument is set as appropriate	✓
1.2.34	Ensure that encryption providers are appropriately configured	✗
1.2.35	Ensure that the API Server only makes use of Strong Cryptographic Ciphers	✓
1.3.1	Ensure that the --terminated-pod-gc-threshold argument is set as appropriate	✗
1.3.2	Ensure that the --profiling argument is set to false	✓
1.3.3	Ensure that the --use-service-account-credentials argument is set to true	✓

CIS recommendation	Description	Compliance
1.3.4	Ensure that the --service-account-private-key-file argument is set as appropriate	✓
1.3.5	Ensure that the --root-ca-file argument is set as appropriate	✓
1.3.6	Ensure that the RotateKubeletServerCertificate argument is set to true	✓
1.3.7	Ensure that the --bind-address argument is set to 127.0.0.1	✓
1.4.1	Ensure that the --profiling argument is set to false	✓
1.4.2	Ensure that the --bind-address argument is set to 127.0.0.1	✓
2.1	Ensure that the --cert-file and --key-file arguments are set as appropriate if use etcd as database	✓
2.2	Ensure that the --client-cert-auth argument is set to true	✓
2.3	Ensure that the --auto-tls argument is not set to true	✓
2.4	Ensure that the --peer-cert-file and --peer-key-file arguments are set as appropriate	✓
2.5	Ensure that the --peer-client-cert-auth argument is set to true	✓
2.6	Ensure that the --peer-auto-tls argument is not set to true	✓
2.7	Ensure that a unique Certificate Authority is used for etcd	✓
3.1.1	Client certificate authentication should not be used for users	✗

CIS recommendation	Description	Compliance
3.2.1	Ensure that a minimal audit policy is created	✘
3.2.2	Ensure that the audit policy covers key security concerns	✘
4.1.1	Ensure that the kubelet service file permissions are set to 644 or more restrictive	✔
4.1.2	Ensure that the kubelet service file ownership is set to root:root	✔
4.1.3	If proxy kubeproxy.kubeconfig file exists ensure permissions are set to 644 or more restrictive	✔
4.1.4	Ensure that the proxy kubeconfig file ownership is set to root:root	✔
4.1.5	Ensure that the --kubeconfig kubelet.conf file permissions are set to 644 or more restrictive	✔
4.1.6	Ensure that the --kubeconfig kubelet.conf file ownership is set to root:root	✔
4.1.7	Ensure that the certificate authorities file permissions are set to 644 or more restrictive	✔
4.1.8	Ensure that the client certificate authorities file ownership is set to root:root	✔
4.1.9	Ensure that the kubelet --config configuration file has permissions set to 644 or more restrictive	✔
4.1.10	Ensure that the kubelet --config configuration file ownership is set to root:root	✔
4.2.1	Ensure that the anonymous-auth argument is set to false	✔
4.2.2	Ensure that the --authorization-mode argument is not set to AlwaysAllow	✔

CIS recommendation	Description	Compliance
4.2.3	Ensure that the --client-ca-file argument is set as appropriate	✓
4.2.4	Ensure that the --read-only-port argument is set to 0	✓
4.2.5	Ensure that the --streaming-connection-idle-timeout argument is not set to 0	✗
4.2.6	Ensure that the --protect-kernel-defaults argument is set to true	✓
4.2.7	Ensure that the --make-iptables-util-chains argument is set to true	✗
4.2.8	Ensure that the --hostname-override argument is not set	✓
4.2.9	Ensure that the --event-qps argument is set to 0 or a level which ensures appropriate event capture	✗
4.2.10	Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as appropriate	✓
4.2.11	Ensure that the --rotate-certificates argument is not set to false	✓
4.2.12	Verify that the RotateKubeletServerCertificate argument is set to true	✓
4.2.13	Ensure that the Kubelet only makes use of Strong Cryptographic Ciphers	✗
5.1.1	Ensure that the cluster-admin role is only used where required	✓
5.1.2	Minimize access to secrets	✓
5.1.3	Minimize wildcard use in Roles and ClusterRoles	✓


CIS recommendation	Description	Compliance
5.1.4	Minimize access to create pods	✓
5.1.5	Ensure that default service accounts are not actively used.	✗
5.1.6	Ensure that Service Account Tokens are only mounted where necessary	✗
5.2.1	Minimize the admission of privileged containers	✓
5.2.2	Minimize the admission of containers wishing to share the host process ID namespace	✓
5.2.3	Minimize the admission of containers wishing to share the host IPC namespace	✓
5.2.4	Minimize the admission of containers wishing to share the host network namespace	✓
5.2.5	Minimize the admission of containers with allowPrivilegeEscalation	✓
5.2.6	Minimize the admission of root containers	✓
5.2.7	Minimize the admission of containers with the NET_RAW capability	✓
5.2.8	Minimize the admission of containers with added capabilities	✓
5.2.9	Minimize the admission of containers with capabilities assigned	✓
5.3.1	Ensure that the CNI in use supports Network Policies	✓
5.3.2	Ensure that all Namespaces have Network Policies defined	✗

CIS recommendation	Description	Compliance
5.4.1	Prefer using secrets as files over secrets as environment variables	✓
5.4.2	Consider external secret storage	✗
5.5.1	Configure Image Provenance using ImagePolicyWebhook admission controller	✗
5.7.1	Create administrative boundaries between resources using namespaces	✓
5.7.2	Ensure that the seccomp profile is set to docker/default in your pod definitions	✗
5.7.3	Apply Security Context to Your Pods and Containers	✓
5.7.4	The default namespace should not be used	✓

14 Troubleshooting and technical assistance

Entrust offers a variety of professional support programs to help you keep Entrust products up and running.

- [Entrust TrustedCare](#)
- [Customer support](#)
- [Professional services](#)
- [Training](#)

 Register for our support programs to use our Web-based support services.

Entrust TrustedCare

Log in to trustedcare.entrust.com for technical resources such as Entrust product documentation, white papers, technical notes, and a comprehensive knowledge base.

Customer support

To contact customer support, generate and send a diagnostics report as explained in the following sections.

- [Generating a diagnostics report](#)
- [Sending the diagnostics report](#)

Generating a diagnostics report

To generate a diagnostic report, run the following command.

```
diagnostic-report.sh [--password <pwd>] [--logs-since <hours>]
```

For example:

```
$ sudo diagnostic-report.sh
Gathering system information. This process could take a few minutes...
Packaging the report...

Randomly generated password:
^zuD,h|o`3!>q

Success. This file is meant for incidence analysis at the Support department's
request.
The file resulting from this execution must be sent to the aforementioned department.
```

See below for the supported options.

- `-p, --password <pwd>`
- `-s, --logs-since <hours>`
- `--fast`
- `--disk-perf`
- `--no-host`
- `-v, --version`

- `-h, --help`

`-p, --password <pwd>`

Encrypt the generated diagnostics file with the `<pwd>` password.

Mandatory: No, When omitting this option, the script ciphers the file with a random password containing 13 OWAS special characters and prints the password in the execution console.

`-s, --logs-since <hours>`

Collect the logs generated in the last `<hour>` hours.

Mandatory: No. When omitted, this value defaults to 24 hours.

`--fast`

Perform a fast diagnostic report.

 Do not use this option unless indicated by customer support.

`--disk-perf`

Generate a disk performance report.

 Do not use this option unless indicated by customer support.

`--no-host`

Skip host reports.

 Do not use this option unless indicated by customer support.

`-v, --version`

Show the version of the report generation tool.

`-h, --help`

Show the help of the report generation tool.

Sending the diagnostics report

Send an email to support@entrust.com with the following information.

- The attached diagnostics report file. Do not change the name of this file.
- Your contact information
- A description of the problem

- The conditions under which the error occurred
- The troubleshooting activities already performed

Professional services

The Entrust team helps organizations worldwide deploy and maintain secure transactions and communications with their partners, customers, suppliers, and employees. Entrust offers a full range of professional services to implement wired and wireless networks, including planning and design, installation, system integration, deployment support, and custom software development.

Whether you choose to operate your Entrust solution in-house or subscribe to hosted services, Entrust Professional Services will design and implement the right solution for your organization's needs. For more information about Entrust Professional Services, please visit our Web site at: <https://www.entrust.com/services>

Training

Entrust delivers practical training to deploy, operate, administer, extend, customize, and support any range of Entrust digital identity and information security solutions through various hands-on courses. Entrust's professional training services provided by training professionals help equip you with the knowledge to speed up the deployment of your security platforms and solutions.

Please visit our training Web site at: <https://www.entrust.com/resource-center/training>

15 Third-party license acknowledgments

See below the license acknowledgments for the third-party software included in Entrust PKI Hub.

- This software is based in part on the work of the FreeType Team.
- This software is based in part on the work of the Independent JPEG Group
- This product includes software developed by The XFree86 Project, Inc (<http://www.xfree86.org>) and its contributors.
- This product includes software developed by Henry Spencer.
- The configuration files in <https://github.com/grafana/helm-charts> have been modified by Entrust for product customization purposes.
- The configuration files in <https://github.com/grafana/loki> have been modified by Entrust for product customization purposes.
- The configuration files in <https://github.com/prometheus-community/helm-charts> have been modified by Entrust for product customization purposes.
- The configuration files in <https://github.com/longhorn/charts> have been modified by Entrust for product customization purposes.

16 Licensing

This section defines licensing terms and permitted uses for the Entrust PKI Hub appliance (“PKI Hub”), and its integrated features/functionality:

- Certificate authority software – currently called PKI Hub Certificate Authorities (CAs).
- Enrollment services – currently called Certificate Enrollment Gateway (CEG).
- Online certificate status protocol – currently called Entrust Validation Authority (EVA).
- Timestamping – currently Timestamping Authority (TSA).
- Certificate lifecycle management – currently called Certificate Hub (Certhub).
- Certificate authority gateway – currently called CAGW).

In this Licensing section, the terms “Customer” and “the Customer” are used to reference an Entrust customer who has:

- purchased one or more PKI Hub licenses; or
- one of that customer’s internal Users who is authorized to access components or features of the PKI Hub in connection with the customer’s business.

In addition, the term “External User” refers to a user who is outside of the Customer organization and the Customer identifies as a required digital certificate user to enable communications between Customer and those External Users concerning Customer’s business.

PKI Hub is licensed for internal Customer use; however, the Customer is permitted to assign identities (uniquely identified end entities) and digital certificates to External Users solely to enable communications between Customer and those External Users concerning Customer’s business.

PKI Hub has three licensing models/types (X-Small, Small, and Medium), each of which includes different product functionalities/features. The functionalities/features included with each model/type are in the table below.

Features / Capabilities	X-SMALL	SMALL	MEDIUM
Certificate Authorities	✓	✓	✓
CA Gateway	✗	✓	✓
OCSP (EVA)	✗	✓	✓
Timestamping	✗	✓	✓
Enrollment Services (CEG)	✗	✓	✓
CLM (Certificate Hub) Find	✗	✓	✓
CLM (Certificate Hub) Control	✗	✗	✓

In addition to the license models/types, PKI Hub is also licensed on the basis of digital certificate volume.

The Customer will receive one or more license key(s) (“licenses”) authorizing or enabling functions/features and certificate volumes based on what the Customer has purchased and subject to the following:

- once issued, digital certificates are deemed to be consumed,

- the Customer may not alter the license key, nor circumvent or attempt to circumvent the license mechanism,
- the Customer may only use a license key provided by Entrust in conjunction with the related Software component of the PKI Hub,
- PKI Hub may be deployed on the Customer's infrastructure and/or commercial cloud accounts. Entrust strongly advises that deployments be kept up to date with our latest product release.

Each PKI Hub license specifies a deployment type, which is categorized either as production or test. If a license is not specifically identified as production or test, it is considered a production license.

- Production licenses allow PKI Hub to be used in a production environment for the active provision of services to issue and manage trusted digital certificates to/for Customer internal and External Users.
- Test licenses require PKI Hub to be deployed and used exclusively in a test (non-production) environment to develop and/or verify integration and configuration changes prior to the promotion of those changes to the Customer production environment.

Each license may be used on multiple deployment clusters of the same type, for example, a test license can be used on multiple test clusters in a test environment.

PKI Hub can extend CAGW functionality, through plugins, to connect to additional CA types. The Customer is permitted to run plugins that are developed by:

- Entrust (sold separately); or
- the Customer or a third-party, pursuant to the CAGW SDK License and recognized (via digital signing) by Entrust.

Plugins are out-of-scope for the product warranty and Entrust support for PKI Hub.

The PKI Hub license explicitly excludes any embedded and/or internal databases and Hardware Security Modules (HSM). These components are external dependencies that must be provided, installed, and configured separately by the Customer prior to the operation of the PKI Hub software.

PKI Hub software contains cryptographic software components. The Customer's country of operation may have import and export requirements that apply.

To ensure Entrust Customer Support is equipped to assist with issues reported, the Customer is expected to maintain reasonable records of the PKI Hub deployment details including:

- instances deployed in production.
- environment(s) in which production instances have been deployed (i.e. Customer infrastructure vs. cloud).
- certificates consumed/available.

17 Certificate profiles reference


Entrust provides the following set of certificate profiles for authorities and end-entities.

- [Basic authority certificate profiles](#)
- [External subordinate CA certificate profiles](#)
- [Subscriber certificate profiles](#)

Basic authority certificate profiles

Entrust provides the following basic profiles for root Certificate Authorities, issuing Certificate Authorities, and Validation Authorities (OCSP).

- basic-ca-root
- basic-ca-subord
- basic-ocsp

 These profiles are not exposed nor configurable. External root CAs are not covered by this profile.


See below a description of these profiles.

- [Key and signature algorithms](#)
- [Certificate fields](#)
- [Certificate critical extensions](#)
- [Certificate non-critical extensions](#)

Key and signature algorithms

All authority basic profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Certificate fields

The authority basic profiles set the following certificate fields.

Field	basic-ca-root	basic-ca-subord	basic-ocsp
Issuer	Self-signed	Customer's online root CA	Customer's online root/ issuing CA
Subject	No constraint	No constraint	No constraint
Validity period	Less than or equal to 20 years	Less than or equal to 10 years. The subordinate expiry cannot exceed the root validity.	30 days

Certificate critical extensions

The authority basic profiles set the following certificate critical extensions.

Extension	basic-ca-root	basic-ca-subord	basic-ocsp
Basic Constraints	cA=True	cA=True, pathLenConstraint=0	cA = False
Extended Key Usage	Never present	Never present	OCSP Signing
Key Usage	digitalSignature, keyCertSign, cRLSign	digitalSignature, keyCertSign, cRLSign	digitalSignature, keyCertSign, cRLSign

Certificate non-critical extensions

The authority basic profiles set the following non-critical certificate extensions.

Extension	basic-ca-root	basic-ca-subord	basic-ocsp
AIA	Never present	Supplied when the customer enables OCSP on CA creation	Always present
Authority Key Identifier	Never present	Matches subjectKeyIdentifier of the signing certificate	Matches subjectKeyIdentifier of the signing certificate

Extension	basic-ca-root	basic-ca-subord	basic-ocsp
CRL Distribution Points	Never present (not applicable)	Always present	Always present
OCSP	Never present	Never present	No check
Subject Key Identifier	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

External subordinate CA certificate profiles

Entrust root Certificate Authorities support the following external subordinate Certificate Authority certificate profiles.

Profile set	Profiles
Azure Firewall Intermediate CA certificate profiles	azure-firewall-ca-subord
TLS Proxy CA certificate profiles	tlsproxy-ca-subord

Azure Firewall Intermediate CA certificate profiles

The Azure Firewall Intermediate CA service provides a `azure-firewall-ca-subord` profile for root Certificate Authorities.

- [Azure Firewall Subordinate CA signing use cases](#)
- [Azure Firewall Subordinate CA request extensions](#)
- [Azure Firewall Subordinate CA certificate fields](#)
- [Azure Firewall Subordinate CA certificate extensions](#)
- [Azure Firewall Subordinate CA algorithm constraints](#)
- [Azure Firewall Subordinate CA distinguished names](#)

i Each external subordinate CA issued by a PKIaaS root CA only consumes one PKIaaS Certificate license. Entrust does not charge for certificates issued by external subordinate CAs because those certificates are considered external and not using the PKIaaS infrastructure.

Azure Firewall Subordinate CA signing use cases


The `azure-firewall-ca-subord` profile supports the following use cases.

- ECS Enterprise UI
- CA Gateway API

Azure Firewall Subordinate CA request extensions

The `azure-firewall-ca-subord` profile supports the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32

 Follow the [Microsoft Azure Intermediate requirements](#) to generate the CSR before requesting the CA certificate from PKIaaS.

Azure Firewall Subordinate CA certificate fields

The `azure-firewall-ca-subord` profile sets the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint
Validity period	Defaults to 1 year if not specified.

Azure Firewall Subordinate CA certificate extensions

The `azure-firewall-ca-subord` profile sets the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA=True, pathLenConstraint=1
CRL Distribution Points	No	Always present
Key Usage	Yes	Certificate Signing, CRL Signing, Digital Signature
Subject Alternative Name	No	No constraints

Extension	Critical	Value
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Azure Firewall Subordinate CA algorithm constraints

The `azure-firewall-ca-subord` profile supports the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Azure Firewall Subordinate CA distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6

Alias	OID
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3

Alias	OID
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

TLS Proxy CA certificate profiles

The TLS Proxy CA service provides provides a `tlsproxy-ca-subord` certificate profile for root Certificate Authorities.

- [TLS Proxy CA use cases](#)
- [TLS Proxy CA request extensions](#)
- [TLS Proxy CA certificate fields](#)
- [TLS Proxy CA certificate extensions](#)
- [TLS Proxy CA algorithm constraints](#)
- [TLS Proxy CA distinguished names](#)

i Each external subordinate CA issued by a PKIaaS root CA only consumes one PKIaaS Certificate license. Entrust does not charge for certificates issued by external subordinate CAs because those certificates are considered external and not using the PKIaaS infrastructure.

TLS Proxy CA use cases

The `tlsproxy-ca-subord` profile supports the following use cases.

- ECS Enterprise UI
- CA Gateway API

TLS Proxy CA request extensions

The `tlsproxy-ca-subord` profile supports the following non-critical extensions in request.

Extension name	Extension OID
CertificatePolicies	2.5.29.32

TLS Proxy CA certificate fields

The `tlsproxy-ca-subord` profile sets the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint
Validity period	Defaults to 1 year if not specified.

TLS Proxy CA certificate extensions

The `tlsproxy-ca-subord` profile sets the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA=True, pathLenConstraint=0
CRL Distribution Points	No	Always present
Extended Key Usage	No	TLS server authentication (1.3.6.1.5.5.7.3.1), TLS client authentication (1.3.6.1.5.5.7.3.2)

Extension	Critical	Value
Key Usage	Yes	Certificate Signing, CRL Signing, Digital Signature
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

TLS Proxy CA algorithm constraints

The `tlsproxy-ca-subord` profile supports the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

TLS Proxy CA distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4

Alias	OID
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1

Alias	OID
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Subscriber certificate profiles

Entrust provides the following certificate policies for end-entity subscribers

Profile set	Profiles
EST certificate profiles	est-digital-signature est-digital-signature-key-encipherment est-key-encipherment est-non-repudiation
Mobile device certificate profile	mobile-device-p12-digital-signature-client-authentication

Profile set	Profiles
CMPv2 certificate profiles	cmp-digital-signature cmp-digital-signature-key-encipherment cmp-key-encipherment cmp-non-repudiation
Active Directory (WSTEP) certificate profiles	wstep-digital-signature wstep-digital-signature-key-encipherment wstep-key-encipherment wstep-non-repudiation wstep-non-repudiation-key-encipherment
Private SSL (ACMEv2) certificate profiles	privatessl-tls-client privatessl-tls-client-server privatessl-tls-client-server-data-encipherment privatessl-tls-client-server-supply-san privatessl-tls-server privatessl-tls-server-supply-san
S_MIME certificate profiles	smime-digital-signature-key-encipherment smime-key-encipherment smime-non-repudiation
Code signing certificate profile	codesigning-digital-signature
MDMWS certificate profiles	mdmws-digital-signature mdmws-digital-signature-key-encipherment mdmws-digital-signature-key-encipherment-clientauth mdmws-key-encipherment mdmws-non-repudiation mdmws-p12-digital-signature mdmws-p12-digital-signature-key-encipherment mdmws-p12-digital-signature-key-encipherment-clientauth mdmws-p12-key-encipherment mdmws-p12-non-repudiation
Smartcard certificate profiles	smartcard-card-authentication smartcard-digital-signature smartcard-domain-controller smartcard-key-management smartcard-piv-authentication smartcard-piv-content-signing
V2G certificate profiles	v2g-supply-equipment v2g-user-identity

Profile set	Profiles
SCEP certificate profiles	scep-digital-signature scep-digital-signature-key-encipherment scep-key-encipherment scep-non-repudiation
Multiuse certificate profiles	multiuse-p12-client multiuse-p12-client-server multiuse-p12-custom multiuse-p12-key-data-encipherment-non-repudiation-client multiuse-p12-key-data-encipherment-non-repudiation-client-server multiuse-p12-key-encipherment-client multiuse-p12-key-encipherment-client-server multiuse-p12-key-encipherment-custom multiuse-p12-key-encipherment-non-repudiation-client multiuse-p12-key-encipherment-non-repudiation-client-server multiuse-p12-key-encipherment-non-repudiation-custom multiuse-p12-key-encipherment-non-repudiation-server multiuse-p12-key-encipherment-server multiuse-p12-non-repudiation-client multiuse-p12-non-repudiation-client-server multiuse-p12-non-repudiation-custom multiuse-p12-non-repudiation-server multiuse-p12-server
Intune certificate profiles	intune-digital-signature intune-digital-signature-key-encipherment intune-digital-signature-key-encipherment-clientauth intune-key-encipherment intune-non-repudiation

Active Directory (WSTEP) certificate profiles

Entrust provides the following Active Directory (WSTEP) certificate profiles.

- wstep-digital-signature
- wstep-digital-signature-key-encipherment

- wstep-key-encipherment
- wstep-non-repudiation
- wstep-non-repudiation-key-encipherment

These profiles support the following features.

- [WSTEP use cases](#)
- [WSTEP key usages](#)
- [WSTEP request extensions](#)
- [WSTEP certificate fields](#)
- [WSTEP certificate extensions](#)
- [WSTEP algorithm constraints](#)
- [WSTEP distinguished names](#)

WSTEP use cases

All WSTEP profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API
- Entrust-hosted Enrollment Gateway
- On-prem Enrollment Gateway

WSTEP key usages

See below the Key Usage extension values supported by each WSTEP profile.

Profile	Key Usage
wstep-digital-signature	Digital Signature
wstep-digital-signature-key-encipherment	Digital Signature, Key Encipherment
wstep-key-encipherment	Key Encipherment
wstep-non-repudiation	Digital Signature, Non-Repudiation
wstep-non-repudiation-key-encipherment	Digital Signature, Non-Repudiation, Key Encipherment

WSTEP request extensions

All WSTEP profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32

Extension name	Extension OID
Extended Key Usage	2.5.29.37
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

WSTEP certificate fields

All WSTEP profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

WSTEP certificate extensions

All WSTEP profiles set the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False

Extension	Critical	Value
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

WSTEP algorithm constraints

All WSTEP profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

WSTEP distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4

Alias	OID
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1

Alias	OID
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

CMPv2 certificate profiles

Entrust provides the following CMPv2 certificate profiles.

- cmp-digital-signature
- cmp-digital-signature-key-encipherment
- cmp-key-encipherment
- cmp-non-repudiation

These profiles support the following features.

- [CMPv2 use cases](#)
- [CMPv2 key usages](#)
- [CMPv2 request extensions](#)
- [CMPv2 certificate fields](#)
- [CMPv2 certificate extensions](#)
- [CMPv2 algorithm constraints](#)

- [CMPv2 distinguished names](#)

CMPv2 use cases

All CMPv2 profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

CMPv2 key usages

See below the Key Usage extension values supported by each CMPv2 profile.

Profile	Key Usage
cmp-digital-signature	Digital Signature
cmp-digital-signature-key-encipherment	Digital Signature, Key Encipherment
cmp-key-encipherment	Key Encipherment
cmp-non-repudiation	Digital Signature, Non-Repudiation

CMPv2 request extensions

All CMPv2 profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32
Extended Key Usage	2.5.29.37
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

CMPv2 certificate fields

All CMPv2 profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

CMPv2 certificate extensions

All CMPv2 profiles set the following certificate extension.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Extended Key Usage	No	No constraints
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

CMPv2 algorithm constraints

All CMPv2 profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

i The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

CMPv2 distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10

Alias	OID
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5

Alias	OID
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Code signing certificate profile

Entrust provides the `codesigning-digital-signature` certificate profile for code signing.

- [Code signing use cases](#)
- [Code signing certificate fields](#)
- [Code signing certificate extensions](#)
- [Code signing algorithm constraints](#)
- [Code signing distinguished names](#)

Code signing use cases

The `codesigning-digital-signature` profile supports the following use cases.

- ECS Enterprise UI
- CA Gateway API

Code signing certificate fields

The `codesigning-digital-signature` profile sets the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA
Validity period	Less than or equal to the expiry of the issuing CA. Default to 1 year if not specified in the request.

Field	Value
Subject	No constraint

Code signing certificate extensions

The `codesigning-digital-signature` profile sets the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Extended Key Usage	No	Code Signing (1.3.6.1.5.5.7.3.3)
Key Usage	Yes	Digital Signature
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Code signing algorithm constraints

The `codesigning-digital-signature` profile supports the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512

Key algorithm	Signature algorithm
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Code signing distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15

Alias	OID
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2

Alias	OID
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

eSIM certificate profiles

Entrust provides the following certificate profiles for eSIM end-entity users.

- esim-delivery-auth
- esim-delivery-binding
- esim-delivery-tls-server
- esim-discovery-auth
- esim-discovery-tls-server

These profiles support the following features.

- [eSIM use cases](#)
- [eSIM key usages and certificate policies](#)
- [eSIM certificate fields](#)
- [eSIM certificate extensions](#)
- [eSIM algorithm constraints](#)
- [eSIM distinguished names](#)

eSIM use cases

All eSIM end-entity profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

eSIM key usages and certificate policies

See below the Key Usage, Extended Key Usage (EKU), and Certificate Policy extension values supported by each eSIM profile.

Profile	Key Usage	Extended Key Usage	Certificate Policy
esim-delivery-auth	Digital Signature	—	id-rspRole-dp-auth (2.23.146.1.2.1.4)
esim-delivery-binding	Digital Signature	—	id-rspRole-dp-pb (2.23.146.1.2.1.5)

Profile	Key Usage	Extended Key Usage	Certificate Policy
esim-delivery-tls-server	Digital Signature	TLS server authentication (1.3.6.1.5.5.7.3.1)	id-rspRole-dp-tls (2.23.146.1.2.1.3)
esim-discovery-auth	Digital Signature	—	id-rspRole-ds-auth (2.23.146.1.2.1.7)
esim-discovery-tls-server	Digital Signature	TLS server authentication (1.3.6.1.5.5.7.3.1)	id-rspRole-ds-tls (2.23.146.1.2.1.6)

eSIM certificate fields

All eSIM profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 3 years if not specified in the request.

eSIM certificate extensions

All eSIM profiles set the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA=False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints

Extension	Critical	Value
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

eSIM algorithm constraints

All eSIM profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

eSIM distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6

Alias	OID
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3

Alias	OID
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

EST certificate profiles

Entrust provides the following EST (Enrollment over Secure Transport) certificate profiles.

- est-digital-signature
- est-digital-signature-key-encipherment
- est-key-encipherment
- est-non-repudiation

These profiles support the following features.

- [EST signing use cases](#)
- [EST key usages](#)
- [EST request extensions](#)
- [EST certificate fields](#)
- [EST certificate extensions](#)
- [EST algorithm constraints](#)
- [EST signing distinguished names](#)

EST signing use cases

All EST profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

EST key usages

See below the Key Usage extension values each EST profile supports.

Profile	Key Usage
est-digital-signature	Digital Signature
est-digital-signature-key-encipherment	Digital Signature, Key Encipherment
est-key-encipherment	Key Encipherment
est-non-repudiation	Digital Signature, Non-Repudiation

EST request extensions

All EST profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32
Extended Key Usage	2.5.29.37
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

EST certificate fields

All EST profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

EST certificate extensions

All EST profiles set the following certificate extension values.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA=False
CRL Distribution Points	No	Always present
Extended Key Usage	No	No constraints
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

EST algorithm constraints

All EST profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384

Key algorithm	Signature algorithm
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

i The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

EST signing distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12

Alias	OID
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13

Alias	OID
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Intune certificate profiles

Entrust provides the following Intune certificate profiles.

- intune-digital-signature
- intune-digital-signature-key-encipherment
- intune-digital-signature-key-encipherment-clientauth
- intune-key-encipherment
- intune-non-repudiation

These profiles support the following features.

- [Intune use cases](#)
- [Intune key usages](#)
- [Intune request extensions](#)
- [Intune certificate fields](#)
- [Intune certificate extensions](#)
- [Intune algorithm constraints](#)
- [Intune distinguished names](#)

Intune use cases

All Intune profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API
- Entrust-hosted Enrollment Gateway
- On-prem Enrollment Gateway

Intune key usages

See below the Key Usage and Extended Key Usage (EKU) extension values supported by each Intune profile.

Profile	Key Usage	Extended Key Usage	Allows Extended Key Usage in request
intune-digital-signature	Digital Signature	—	✓
intune-digital-signature-key-encipherment	Digital Signature, Key Encipherment	—	✓
intune-digital-signature-key-encipherment-clientauth	Digital Signature, Key Encipherment	TLS client authentication (1.3.6.1.5.5.7.3.2)	✗
intune-key-encipherment	Key Encipherment	—	✓
intune-non-repudiation	Digital Signature, Non-Repudiation	—	✓

Intune request extensions

All Intune profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

Intune certificate fields

All Intune profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.

Field	Value
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

Intune certificate extensions

All Intune profiles set the following certificate extensions.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Intune algorithm constraints

All Intune profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption

Key algorithm	Signature algorithm
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

i The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Intune distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17

Alias	OID
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3

Alias	OID
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

MDMWS certificate profiles

Entrust provides the following MDMWS (Mobile Device Management Web Service) certificate profiles.

- mdmws-digital-signature
- mdmws-digital-signature-key-encipherment
- mdmws-digital-signature-key-encipherment-clientauth
- mdmws-key-encipherment
- mdmws-non-repudiation
- mdmws-p12-digital-signature
- mdmws-p12-digital-signature-key-encipherment
- mdmws-p12-digital-signature-key-encipherment-clientauth
- mdmws-p12-key-encipherment
- mdmws-p12-non-repudiation

These profiles support the following features.

- [MDMWS use cases](#)
- [MDMWS issuance modes and key usages](#)
- [MDMWS request extensions](#)
- [MDMWS certificate fields](#)
- [MDMWS certificate extensions](#)
- [MDMWS algorithm constraints](#)
- [MDMWS distinguished names](#)

MDMWS use cases

All MDMWS profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API
- Entrust-hosted Enrollment Gateway
- On-prem Enrollment Gateway

MDMWS issuance modes and key usages

MDMWS profiles support the following issuance modes:

- Issue the certificate from a CSR.
- Issue the certificate and an RSA2048 private key in a P12 file.

See below the issuance mode, Key Usage, and Extended Key Usage (EKU) values each MDMWSprofile supports.

Profile	CSR	P12	Key Usage	Extended Key Usage	Allows Extended Key Usage in request
mdmws-digital-signature	✓	✗	Digital Signature	No constraints	✓
mdmws-digital-signature-key-encipherment	✓	✗	Digital Signature, Key Encipherment	No constraints	✓
mdmws-digital-signature-key-encipherment-clientauth	✓	✗	Digital Signature, Key Encipherment	TLS client authentication (1.3.6.1.5.5.7.3.2)	✗
mdmws-key-encipherment	✓	✗	Key Encipherment	No constraints	✓
mdmws-non-repudiation	✓	✗	Digital Signature, Non-Repudiation	No constraints	✓
mdmws-p12-digital-signature	✓	✓	Digital Signature	No constraints	✓
mdmws-p12-digital-signature-key-encipherment	✓	✓	Digital Signature, Key Encipherment	No constraints	✓
mdmws-p12-digital-signature-key-encipherment-clientauth	✓	✓	Digital Signature, Key Encipherment	TLS client authentication (1.3.6.1.5.5.7.3.2)	✗
mdmws-p12-key-encipherment	✓	✓	Key Encipherment	No constraints	✓
mdmws-p12-non-repudiation	✓	✓	Digital Signature, Non-Repudiation	No constraints	✓

MDMWS request extensions

All MDMWS profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

MDMWS certificate fields

All MDMWS profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

MDMWS certificate extensions

All MDMWS profiles set the following certificate extension values.


Extension	OID
CertificatePolicies	2.5.29.32
ApplicationPolicies	1.3.6.1.4.1.311.21.10
SmimeCapabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7

Extension	OID
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

MDMWS algorithm constraints

All MDMWS profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

MDMWS distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6

Alias	OID
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3

Alias	OID
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Mobile device certificate profile

Entrust provides the `mobile-device-p12-digital-signature-client-authentication` certificate profile for mobile devices.

- [Mobile device use cases](#)
- [Mobile device certificate issuance mode](#)
- [Mobile device certificate fields](#)
- [Mobile device certificate extensions](#)
- [Mobile device algorithm constraints](#)
- [Mobile distinguished names](#)

Mobile device use cases

The `mobile-device-p12-digital-signature-client-authentication` profile supports the following use cases.

- ECS Enterprise UI
- CA Gateway API

Mobile device certificate issuance mode

The `mobile-device-p12-digital-signature-client-authentication` profile supports the following issuance modes:

- Issue the certificate from a CSR.
- Issue the certificate and an RSA2048 private key in a P12 file.

Mobile device certificate fields

The `mobile-device-p12-digital-signature-client-authentication` profile sets the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA
Validity period	Less than or equal to subordinate expiry of the issuing CA. Default to 1 year if not specified in the request.
Subject	No constraint

Mobile device certificate extensions

The `mobile-device-p12-digital-signature-client-authentication` profile sets the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Extended Key Usage	No	TLS client authentication (1.3.6.1.5.5.7.3.2)
Key Usage	Yes	Digital Signature
Subject Alternative Name	No	No constraints

Extension	Critical	Value
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Mobile device algorithm constraints

The `mobile-device-p12-digital-signature-client-authentication` profile supports the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Mobile distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5

Alias	OID
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2

Alias	OID
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Multiuse certificate profiles

Entrust provides the following multiuse profiles.

- multiuse-p12-client
- multiuse-p12-client-server
- multiuse-p12-custom
- multiuse-p12-key-data-encipherment-non-repudiation-client
- multiuse-p12-key-data-encipherment-non-repudiation-client-server
- multiuse-p12-key-encipherment-client
- multiuse-p12-key-encipherment-client-server
- multiuse-p12-key-encipherment-custom
- multiuse-p12-key-encipherment-non-repudiation-client
- multiuse-p12-key-encipherment-non-repudiation-client-server
- multiuse-p12-key-encipherment-non-repudiation-custom
- multiuse-p12-key-encipherment-non-repudiation-server
- multiuse-p12-key-encipherment-server
- multiuse-p12-non-repudiation-client

- multiuse-p12-non-repudiation-client-server
- multiuse-p12-non-repudiation-custom
- multiuse-p12-non-repudiation-server
- multiuse-p12-server

These profiles support the following features.

- [Multiuse use cases](#)
- [Multiuse issuance modes](#)
- [Multiuse key usages](#)
- [Multiuse request extensions](#)
- [Multiuse certificate fields](#)
- [Multiuse certificate extensions](#)
- [Multiuse algorithm constraints](#)
- [Multiuse distinguished names](#)

Multiuse use cases

All multiuse profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

Multiuse issuance modes

All multiuse profiles support the following issuance modes:

- Issue the certificate from a CSR.
- Issue the certificate and an RSA2048 private key in a P12 file.

Multiuse key usages

See below the Key Usage and Extended Key Usage (EKU) extension values each multiuse profile supports.

Profile Name	Key Usage	Extended Key Usage	Allows Extended Key Usage in request
multiuse-p12-client	Digital Signature, Key Agreement	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-client-server	Digital Signature, Key Agreement	TLS client Authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-custom	Digital Signature, Key Agreement	No constraints	✓

Profile Name	Key Usage	Extended Key Usage	Allows Extended Key Usage in request
multiuse-p12-key-data-encipherment-non-repudiation-client	Digital Signature, Non-Repudiation, Key Encipherment, Data Encipherment	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-key-data-encipherment-non-repudiation-client-server	Digital Signature, Non-Repudiation, Key Encipherment, Data Encipherment	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-key-encipherment-client	Digital Signature, Key Agreement, Key Encipherment	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-key-encipherment-client-server	Digital Signature, Key Agreement, Key Encipherment	TLS client Authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-key-encipherment-custom	Digital Signature, Key Agreement, Key Encipherment	No constraints	✓
multiuse-p12-key-encipherment-non-repudiation-client	Digital Signature, Key Agreement, Key Encipherment, Non-Repudiation	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-key-encipherment-non-repudiation-client-server	Digital Signature, Key Agreement, Key Encipherment, Non-Repudiation	TLS client Authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-key-encipherment-non-repudiation-custom	Digital Signature, Key Agreement, Key Encipherment, Non-Repudiation	No constraints	✓

Profile Name	Key Usage	Extended Key Usage	Allows Extended Key Usage in request
multiuse-p12-key-encipherment-non-repudiation-server	Digital Signature, Key Agreement, Key Encipherment, Non-Repudation	TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-key-encipherment-server	Digital Signature, Key Agreement, Key Encipherment	TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-non-repudiation-client	Digital Signature, Key Agreement, Non-Repudation	TLS client Authentication (1.3.6.1.5.5.7.3.2)	✗
multiuse-p12-non-repudiation-client-server	Digital Signature, Key Agreement, Non-Repudation	TLS client Authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-non-repudiation-custom	Digital Signature, Key Agreement, Non-Repudation	No constraints	✓
multiuse-p12-non-repudiation-server	Digital Signature, Key Agreement, Non-Repudation	TLS server authentication (1.3.6.1.5.5.7.3.1)	✗
multiuse-p12-server	Digital Signature, Key Agreement	TLS server authentication (1.3.6.1.5.5.7.3.1)	✗

Multiuse request extensions

All multiuse profiles support the following non-critical extensions in request.

Extension	OID
ApplicationPolicies	1.3.6.1.4.1.311.21.10
CertificatePolicies	2.5.29.32

Multiuse certificate fields

All multiuse profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

Multiuse certificate extensions

All multiuse profiles set the following certificate extensions.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Key Usage	Yes	Digital Signature, Key Encipherment
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Multiuse algorithm constraints

All multiuse profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

i The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Multiuse distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10

Alias	OID
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5

Alias	OID
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Private SSL (ACMEv2) certificate profiles

Entrust provides the following Private SSL (ACMEv2) certificate profiles.

- `privatessl-tls-client`
- `privatessl-tls-client-server`
- `privatessl-tls-client-server-data-encipherment`
- `privatessl-tls-client-server-supply-san`
- `privatessl-tls-server`
- `privatessl-tls-server-supply-san`

These profiles support the following features.

- [Private SSL use cases](#)
- [Private SSL key usages](#)
- [Private SSL fill_san_dns_with_cn](#)
- [Private SSL request extensions](#)
- [Private SSL certificate fields](#)
- [Private SSL certificate extensions](#)
- [Private SSL algorithm constraints](#)
- [Private SSL distinguished names](#)

Private SSL use cases

All private SSL profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

Private SSL key usages

See below the Key Usage and Extended Key Usage (EKU) extension values each private SSL profile supports.

Profile	Key Usage	Extended Key Usage
privatessl-tls-client	Digital Signature	TLS client authentication (1.3.6.1.5.5.7.3.2)
privatessl-tls-client-server	Digital Signature	TLS client authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)
privatessl-tls-client-server-data-encipherment	Digital Signature, Data Encipherment	TLS client authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)
privatessl-tls-client-server-supply-san	Digital Signature	TLS client authentication (1.3.6.1.5.5.7.3.2) TLS server authentication (1.3.6.1.5.5.7.3.1)
privatessl-tls-server	Digital Signature	TLS server authentication (1.3.6.1.5.5.7.3.1)
privatessl-tls-server-supply-san	Digital Signature	TLS server authentication (1.3.6.1.5.5.7.3.1)

Private SSL fill_san_dns_with_cn

When the `fill_san_dns_with_cn` parameter is `True`, the profile copies in the `SubjectAltname` extension all the `CN` fields:

- included in the `Subject` extension, and
- not already in the `SubjectAltname` extension (to avoid duplicated entries).

See below the value of this parameter in each profile.

Profile	fill_san_dns_with_cn
privatessl-tls-client	False
privatessl-tls-client-server	False
privatessl-tls-client-server-data-encipherment	False

Profile	fill_san_dns_with_cn
privatessl-tls-client-server-supply-san	True
privatessl-tls-server	False
privatessl-tls-server-supply-san	True

Private SSL request extensions

All private SSL profiles support the following non-critical extensions in request.

Extension Name	Extension OID
Application Policies	1.3.6.1.4.1.311.21.10
Certificate Policies	2.5.29.32

Private SSL certificate fields

All Private SSL profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

Private SSL certificate extensions

All private SSL profiles set the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate

Extension	Critical	Value
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Private SSL algorithm constraints

All private SSL profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Private SSL distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3

Alias	OID
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8

Alias	OID
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

S/MIME Secure Email certificate profiles

Entrust provides the following S/MIME Secure Email certificate profiles.

- [smime-digital-signature-key-encipherment](#)
- [smime-key-encipherment](#)
- [smime-non-repudiation](#)

These profiles support the following features.

- [S/MIME use cases](#)
- [S/MIME key usages](#)
- [S/MIME certificate fields](#)
- [S/MIME certificate extensions](#)

- [S/MIME algorithm constraints](#)
- [S/MIME distinguished names](#)

S/MIME use cases

All S/MIME Secure Email profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

S/MIME key usages

See below the Key Usage and Extended Key Usage (EKU) extension values each profile supports.

Profile	Key Usage	Extended Key Usage
smime-digital-signature-key-encipherment	Digital Signature, Key Encipherment	TLS client authentication (1.3.6.1.5.5.7.3.2) Email Protection (1.3.6.1.5.5.7.3.4)
smime-key-encipherment	Key Encipherment	Email Protection (1.3.6.1.5.5.7.3.4)
smime-non-repudiation	Digital Signature, Non-Repudiation	Email Protection (1.3.6.1.5.5.7.3.4)

S/MIME certificate fields

All S/MIME Secure Email profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

S/MIME certificate extensions


All S/MIME Secure Email profiles set the following certificate extensions.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

S/MIME algorithm constraints

All S/MIME Secure Email profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

S/MIME distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1

Alias	OID
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

SCEP certificate profiles

Entrust provides the following SCEP (Simple Certificate Enrollment Protocol) certificate profiles.

- scep-digital-signature
- scep-digital-signature-key-encipherment
- scep-key-encipherment
- scep-non-repudiation

These profiles support the following features.

- [SCEP use cases](#)
- [SCEP key usages](#)
- [SCEP request extensions](#)
- [SCEP certificate fields](#)
- [SCEP certificate extensions](#)
- [SCEP algorithm constraints](#)
- [SCEP distinguished names](#)

SCEP use cases

All SCEP profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

SCEP key usages

See below the Key Usage extension values each profile supports.

Profile	Key Usage
scep-digital-signature	Digital Signature
scep-digital-signature-key-encipherment	Digital Signature, Key Encipherment
scep-key-encipherment	Key Encipherment
scep-non-repudiation	Digital Signature, Non-Repudiation

SCEP request extensions

All SCEP profiles support the following non-critical extensions in request.

Extension name	Extension OID
Certificate Policies	2.5.29.32
Extended Key Usage	2.5.29.37
Application Policies	1.3.6.1.4.1.311.21.10
Smime Capabilities	1.2.840.113549.1.9.15
MSTemplateOID	1.3.6.1.4.1.311.21.7

Extension name	Extension OID
MSTemplateName	1.3.6.1.4.1.311.20.2
szOID_NTDS_CA_SECURITY_EXT	1.3.6.1.4.1.311.25.2

SCEP certificate fields

All SCEP profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

SCEP certificate extensions

All SCEP profiles set the following certificate extensions.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA=False
CRL Distribution Points	No	Always present
Extended Key Usage	No	No constraints
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

SCEP algorithm constraints

All SCEP profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption



The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

SCEP distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9

Alias	OID
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4

Alias	OID
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

Smartcard certificate profiles

Entrust provides the following smartcard certificate profiles.

- smartcard-card-authentication
- smartcard-digital-signature
- smartcard-domain-controller
- smartcard-key-management
- smartcard-piv-authentication
- smartcard-piv-content-signing

These profiles support the following features.

- [Smartcard use cases](#)
- [Smartcard key usages and request extensions](#)
- [Smartcard certificate fields](#)
- [Smartcard certificate extensions](#)
- [Smartcard algorithm constraints](#)
- [Smartcard distinguished names](#)

Smartcard use cases

All smartcard profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

Smartcard key usages and request extensions

See below the Key Usage and Extended Key Usage (EKU) extension values each smartcard profile supports.

Profile	Key Usage	Extended Key Usage	Allowed in request
smartcard-card-authentication	Digital Signature	No constraints	PIV Interim Indicator (2.16.840.1.101.3.6.9.1) Security ID (1.3.6.1.4.1.311.25.2)
smartcard-digital-signature	Digital Signature, Non-Repudiation	No constraints	PIV Interim Indicator (2.16.840.1.101.3.6.9.1) Security ID (1.3.6.1.4.1.311.25.2)
smartcard-domain-controller	Digital Signature, Key Encipherment	TLS server authentication (1.3.6.1.5.5.7.3.1) TLS client authentication (1.3.6.1.5.5.7.3.2)	—
smartcard-key-management	Key Encipherment	No constraints	PIV Interim Indicator (2.16.840.1.101.3.6.9.1) Security ID (1.3.6.1.4.1.311.25.2)
smartcard-piv-authentication	Digital Signature	Any Extended Key Usage (2.5.29.37.0) Microsoft Smart Card Login (1.3.6.1.4.1.311.20.2.2) TLS client authentication (1.3.6.1.5.5.7.3.2)	PIV Interim Indicator (2.16.840.1.101.3.6.9.1) Security ID (1.3.6.1.4.1.311.25.2)
smartcard-piv-content-signing	Digital Signature, Non-Repudiation	No constraints	—

Smartcard certificate fields

All smartcard profiles set the following certificate fields.

Field	Value
Issuer	Customer's subordinate issuing CA.
Subject	No constraint.

Field	Value
Validity period	Less than or equal to the expiry of the issuing CA. Defaults to 1 year if not specified in the request.

Smartcard certificate extensions

All smartcard profiles set the following certificate extensions.


Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

Smartcard algorithm constraints

All smartcard profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption
RSA 3072	sha256WithRSAEncryption

Key algorithm	Signature algorithm
RSA 4096	sha512WithRSAEncryption

 The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

Smartcard distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17
'givenName' 'G'	2.5.4.42

Alias	OID
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4

Alias	OID
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6

V2G certificate profiles

Entrust provides the following V2G (Vehicle-to-Grid) certificate profiles

- v2g-supply-equipment
- v2g-user-identity

These profiles support the following features.

- [V2G use cases](#)
- [V2G key usages and validity periods](#)
- [V2G certificate fields](#)
- [V2G certificate extensions](#)
- [V2G algorithm constraints](#)
- [V2G distinguished names](#)

V2G use cases

All V2G profiles support the following use cases.

- ECS Enterprise UI
- CA Gateway API

V2G key usages and validity periods

See below the Key Usage, Extended Key Usage (EKU), and certificate validity period each profile supports.

Profile	Key Usage	Extended Key Usage	Validity period
v2g-supply-equipment	Digital Signature, Key Agreement	TLS server authentication (1.3.6.1.5.5.7.3.1)	1 year
v2g-user-identity	Digital Signature, Non-Repudiation	—	2 year

V2G certificate fields

All V2G profiles set the following certificate fields.

Field	Value
Issuer	The customer's subordinate issuing-CA
Subject	No constraint

V2G certificate extensions

All V2G profiles set the following certificate extensions.

Extension	Critical	Value
AIA	No	Supplied if the customer enables OCSP when creating the CA
Authority Key Identifier	No	Matches subjectKeyIdentifier of the signing certificate
Basic Constraints	Yes	cA =False
CRL Distribution Points	No	Always present
Subject Alternative Name	No	No constraints
Subject Key Identifier	No	«The leftmost 160-bits of the SHA-256 hash of the value of the BIT STRING subjectPublicKey» as described in RFC 7093 section 2

V2G algorithm constraints

All V2G profiles support the following key and signature algorithms.

Key algorithm	Signature algorithm
ECDSA P-256	ecdsa-with-SHA256
ECDSA P-384	ecdsa-with-SHA384
ECDSA P-521	ecdsa-with-SHA512
RSA 2048	sha256WithRSAEncryption

Key algorithm	Signature algorithm
RSA 3072	sha256WithRSAEncryption
RSA 4096	sha512WithRSAEncryption

i The NIST will deprecate some algorithms after Dec 31, 2030. See <https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf>

V2G distinguished names

Entrust has no restriction on Distinguished Names (DNs) per certificate profile. All certificate profiles support the following identifiers.

Alias	OID
'CN' 'CommonName'	2.5.4.3
'SN' 'SurName'	2.5.4.4
'SERIALNUMBER' 'DeviceSerialNumber'	2.5.4.5
'C' 'Country'	2.5.4.6
'L' 'Locality'	2.5.4.7
'ST' 'S' 'State'	2.5.4.8
'STREET' 'StreetAddress'	2.5.4.9
'O' 'Org' 'Organization'	2.5.4.10
'OU' 'OrganizationalUnit' 'OrganizationUnit' 'OrgUnit'	2.5.4.11
'T' 'Title'	2.5.4.12
'BUSINESSCATEGORY'	2.5.4.15
'POSTALCODE'	2.5.4.17

Alias	OID
'givenName' 'G'	2.5.4.42
'I' 'Initials'	2.5.4.43
'ORGANIZATIONIDENTIFIER'	2.5.4.97
'UID'	0.9.2342.19200300.100.1.1
'DC' 'DomainComponent'	0.9.2342.19200300.100.1.25
'Email' 'E'	1.2.840.113549.1.9.1
'unstructuredName'	1.2.840.113549.1.9.2
'unstructuredAddress'	1.2.840.113549.1.9.8
'JurisdictionOfIncorporationLocalityName'	1.3.6.1.4.1.311.60.2.1.1
'JurisdictionOfIncorporationStateOrProvinceName'	1.3.6.1.4.1.311.60.2.1.2
'JurisdictionOfIncorporationCountryName'	1.3.6.1.4.1.311.60.2.1.3
'TrademarkOfficeName'	1.3.6.1.4.1.53087.1.2
'TrademarkCountryOrRegionName'	1.3.6.1.4.1.53087.1.3
'TrademarkRegistration'	1.3.6.1.4.1.53087.1.4
'LegalEntityIdentifier'	1.3.6.1.4.1.53087.1.5
'WordMark'	1.3.6.1.4.1.53087.1.6
'MarkType'	1.3.6.1.4.1.53087.1.13
'StatuteCountryName'	1.3.6.1.4.1.53087.3.2
'StatuteStateOrProvinceName'	1.3.6.1.4.1.53087.3.3

Alias	OID
'StatuteLocalityName'	1.3.6.1.4.1.53087.3.4
'StatuteCitation'	1.3.6.1.4.1.53087.3.5
'StatuteURL'	1.3.6.1.4.1.53087.3.6