

# SwissPKI

libC Technologies SA

User Manual – version 2

# SwissPKI™



**Copyright © 2012-2023, libC Technologies SA. All rights reserved.**

The Programs (which include both the software and documentation) contain proprietary information of libC Technologies SA; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property law. Reverse engineering, disassembly or decompilation of the Programs is prohibited.

Program Documentation is licensed for use solely to support the deployment of the Programs and not for any other purpose. The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. libC Technologies SA does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of libC Technologies SA.

## Revision

Rev	Date	Who	Comment
1.0	23.12.2020	libC Technologies SA	Initial document
1.1	05.01.2021	libC Technologies SA	Change data flow graph
1.2	24.09.2021	libC Technologies SA	Update LDAP schema for user authentication
1.3	09.10.2021	libC Technologies SA	Updated information for user onboarding process
1.4	02.12.2021	libC Technologies SA	Added SCION section
1.5	03.12.2021	libC Technologies SA	Editorial Changes
1.6	08.12.2021	libC Technologies SA	Updated LDAP publication
1.7	13.12.2021	libC Technologies SA	Added Microsoft Certificate Template section
1.8	20.12.2021	libC Technologies SA	Updated Permission templates screen shots
1.9	21.12.2021	libC Technologies SA	Updated Custom extensions and OCSP No Check
1.10	26.12.2021	libC Technologies SA	Updated issuance Workflow
1.11	27.12.2021	libC Technologies SA	Updated certificate publication override options
1.12	29.12.2021	libC Technologies SA	Updated notification templates
1.13	01.01.2022	libC Technologies SA	Updated HSM LunaSA settings
1.14	01.01.2022	libC Technologies SA	Updated Scheduler information
1.15	06.01.2022	libC Technologies SA	Updated renewal rule information, CA settings
1.16	13.01.2022	libC Technologies SA	Editorial changes
1.17	24.01.2022	libC Technologies SA	Updated notification templates
1.18	31.01.2022	libC Technologies SA	Updated publisher settings and scheduler information
1.19	31.01.2022	libC Technologies SA	Added Certificate Publications
1.20	05.02.2022	libC Technologies SA	Editorial changes
1.21	09.02.2022	libC Technologies SA	Editorial changes
1.22	14.02.2022	libC Technologies SA	Updated screenshot for OCSP
1.23	23.02.2022	libC Technologies SA	Editorial changes

1.24	24.02.2022	libC Technologies SA	HSM referenced key alias in policy template
1.25	25.02.2022	libC Technologies SA	Air Gaped CA, Offline CA, editorial changes
1.26	10.03.2022	libC Technologies SA	Editorial Changes
1.27	17.03.2022	libC Technologies SA	PKCS#12 key pair generation with end user PIN protection input
1.28	21.03.2022	libC Technologies SA	Updated Admin UI blacklist
1.29	21.04.2022	libC Technologies SA	Added additional certificate policy template SDN field option
1.30	12.05.2022	libC Technologies SA	Added additional policy instance validators (CN and Serial Numbers)
1.31	16.06.2022	libC Technologies SA	Updated permissions for CA certificate revocation. Updated Job names
1.32	23.06.2022	libC Technologies SA	Updated Scheduler information with automatic CAB Suffix download
1.33	08.07.2022	libC Technologies SA	Updated LDAP Server Publisher with unique publication options
1.34			Skipped
1.35	25.07.2022	libC Technologies SA	Updated migration section with version information
1.36	25.07.2022	libC Technologies SA	Added key generation type PKCS10 or PKCS12 (w PIN)
1.37	22.08.2022	libC Technologies SA	Added TRC signing
1.38	15.09.2022	libC Technologies SA	Updated migration version. Added policy instance activation/deactivation
1.39	29.09.2022	libC Technologies SA	Editorial changes in Event section
1.40	31.10.2022	libC Technologies SA	Added SwissSign CA information. Added ZertES and eIDAS policy extensions. Added updates in version 2.1
1.41	20.12.2022	libC Technologies SA	Added www and wildcard base domain validators
1.42	15.01.2023	libC Technologies SA	Update scheduler section Update CAO dashboard

1.43	24.04.2023	libC Technologies SA	Update scheduler section with EmailValidationLinkScheduler and network connections table
1.44	12.06.2023	libC Technologies SA	2.2.2 features
1.45	03.08.2023	libC Technologies SA	Updated DSS features Updated TSA features New feature 4 eyes control for certificate issuance authorization New extension OCSP must staple New extension Private key usage period
1.46	14.08.2023	libC Technologies SA	Updated rule settings and certificate details for 4-eye authorization

Acronym	Meaning
<b>Administrator</b>	User that has the admin rights on the Admin UI part.
<b>AIA</b>	Authority Information Access
<b>AKI</b>	Authority Key Information
<b>ARL</b>	Authority Revocation List
<b>Authorizer</b>	Can accept or reject certificates issuance, renewal, revocation, and recovery.
<b>BC</b>	Basic Constraint
<b>CA</b>	Certification Authority
<b>CAA</b>	Certification Authority Authorization Rule
<b>CAO</b>	User that has admin rights on the Operator UI part.
<b>CDP</b>	CRL Distribution Point
<b>CEP</b>	Microsoft Certificate Enrolment Policy Service
<b>CES</b>	Microsoft Certificate Enrolment Service
<b>Client</b>	The concept of a client is a logical grouping of the distinct PKIs that can be created for a realm.
<b>CNG</b>	Microsoft's CryptoAPI Next Generation
<b>CMP</b>	Certificate Management Protocol
<b>CPS</b>	Certificate Policies
<b>CRL</b>	Certificate Revocation List
<b>CT</b>	Certificate Transparency
<b>DC</b>	Domain Controller
<b>DIT</b>	Directory Information Tree (LDAP)
<b>DSS</b>	Document Signer Service
<b>EKU</b>	Extended key Usage
<b>KU</b>	Key Usage
<b>LDAP</b>	Lightweight Directory Access Protocol

<b>MAP</b>	Microsoft Application Policies
<b>MCT</b>	Microsoft Certificate Template
<b>MSCA</b>	Microsoft Certification Authority
<b>NC</b>	Name Constraint
<b>OCSP</b>	Online Certificate Status Protocol
<b>OIDC</b>	OpenID Connect v1
<b>Provider</b>	Logical name of the OIDC Identity Provider
<b>QCv2</b>	Qualified Statement v2
<b>RAO</b>	Registration Authority Officer. Can issue, revoke, recover or renew certificates.
<b>Realm</b>	A Realm contains a complete PKI and represents a tenant. Multiple Realms are supported on one SwissPKI deployment. Realms are isolated from one another.
<b>RP</b>	Relying Party (OIDC)
<b>SAN</b>	Subject Alternative Name
<b>SCEP</b>	Simple Certificate Enrolment Protocol
<b>SKI</b>	Subject Key Identifier
<b>TRC</b>	SCION Trusted Root Configuration
<b>TSA</b>	Time Stamp Authority

## Contents

1	Introduction .....	15
1.1	Standards.....	16
2	What is new in SwissPKI v2 .....	17
2.1	What is new in SwissPKI version 2.0 .....	17
2.2	What is new in SwissPKI version 2.1 .....	18
2.3	What is new in SwissPKI version 2.2 .....	18
2.4	Changes between SwissPKI v1 and v2.....	19
3	Deployment Requirements.....	20
3.1	Prerequisites.....	20
3.1.1	Database and Storage .....	20
3.1.2	Infrastructure .....	20
3.1.3	Hardware Security Modules .....	21
3.2	Bare Metal.....	21
3.3	Kubernetes .....	21
4	Configuration .....	22
4.1	Bare Metal.....	22
4.2	Kubernetes .....	22
4.3	Logging .....	22
4.3.1	Bare Metal.....	22
4.3.2	Kubernetes.....	22
4.4	Database Initialization.....	22
4.5	Database Migration.....	23
4.5.1	Migrating the DB.....	23
5	General PKI Considerations .....	24
6	Architecture Overview .....	26
6.1	Components .....	27
6.1.1	Administration Portal (Admin).....	27
6.1.2	Operator Portal (Operator).....	27



6.1.3	Registration Authority (RA).....	27
6.1.4	Certification Authority (CA) .....	27
6.1.5	Microsoft CES/CEP (MSCA) .....	27
6.1.6	Automatic Certificate Management Environment (ACME).....	27
6.1.7	Online Certificate Status Protocol (OCSP) .....	27
6.1.8	Time Stamp Authority (TSA) .....	27
6.1.9	SCEP/NDES (SCEP).....	27
6.1.10	CRL Distribution Point (CDP).....	28
6.1.11	Authority Information Access (AIA) .....	28
6.1.12	SCION PKI Adapter (SCION).....	28
6.1.13	Certificate Management Protocol (CMP) .....	28
6.1.14	Document Signer Server (DSS).....	28
6.1.15	Publisher .....	28
6.1.16	Concierge .....	28
6.1.17	Scheduler .....	28
7	Security Considerations .....	29
7.1	General Security .....	29
7.2	Connection Flows .....	30
7.2.1	Administrator UI .....	30
7.2.2	Operator UI .....	30
7.2.3	Registration UI .....	31
7.2.4	Certification Authority (CA) .....	31
7.2.5	Automatic Certificate Management Environment (ACME).....	31
7.2.6	Microsoft CES/CEP (MSCA) .....	32
7.2.7	Online Certificate Status Protocol (OCSP) .....	32
7.2.8	Time Stamp Authority (TSA) .....	32
7.2.9	SCEP/NDES (SCEP).....	32
7.2.10	CRL Distribution Point (CDP).....	33
7.2.11	Authority Information Access (AIA).....	33

7.2.12	SCION PKI Adapter (SCION).....	33
7.2.13	Certificate Management Protocol (CMP) .....	33
7.2.14	Document Signer Server (DSS).....	34
7.2.15	Publisher .....	34
7.2.16	Concierge .....	34
7.2.17	Scheduler .....	35
7.3	Health Checks .....	36
7.4	Roles and Permissions .....	36
7.4.1	Roles.....	36
7.4.2	Permissions .....	37
8	Working with SwissPKI .....	46
8.1	SwissPKI Architectural Model.....	46
8.1.1	Deployment.....	46
8.1.2	Realm .....	47
8.1.3	Clients.....	47
8.1.4	Certificate Products .....	47
8.1.5	Users .....	50
8.1.6	Initializing SwissPKI .....	51
8.1.7	Configuring Realms .....	51
8.1.8	Setting up the PKI.....	52
8.1.9	Issuing Certificates .....	56
8.2	End User Login Options .....	57
8.2.1	Onboarded vs Validated users .....	58
8.2.2	Username/Password with TOTP Login .....	58
8.2.3	LDAP Server.....	62
8.2.4	OpenID Connect.....	71
8.2.5	Kerberos.....	77
8.3	Certification Authority (CA).....	78
8.3.1	Online and Offline Certification Authorities .....	79

8.3.2	CRL Distribution Points (CDP) .....	80
8.3.3	Authority Information Access (AIA) .....	84
8.4	Automatic Certificate Management Environment (ACME) .....	89
8.5	Microsoft CES/CEP (MSCA).....	92
8.5.1	Microsoft CEP.....	93
8.5.2	Microsoft CES.....	94
8.5.3	Microsoft Service on SwissPKI .....	94
8.6	Online Certificate Status Protocol (OCSP).....	95
8.7	Time Stamp Authority (TSA).....	97
8.8	SCEP/NDES (SCEP) .....	99
8.9	SCION PKI Adapter (SCION) .....	100
8.10	Certificate Management Protocol (CMP).....	101
8.11	Document Signer Server (DSS) .....	104
8.12	Publisher .....	108
8.13	Concierge.....	110
8.14	Scheduler .....	111
8.15	DNS .....	117
8.15.1	Challenge Tokens .....	117
8.15.2	DNS Tree traversal .....	118
8.15.3	CAA Resource record processing.....	121
9	Initializing SwissPKI .....	122
9.1	Step 1 - License Agreement.....	122
9.2	Step 2 - SMTP Server .....	123
9.3	Step 3 - System Administrator .....	124
9.4	Step 4 - QR Code.....	125
9.5	Step 5 - Review .....	126
10	Account .....	127
10.1	Account details .....	127
10.2	Account Permissions.....	128

10.3	Account API Keys .....	129
10.4	Account TOTP .....	130
11	Administrator UI .....	131
11.1	PKI Administrators .....	131
11.1.1	Creating PKI Administrators .....	132
11.1.2	Editing PKI Administrators .....	133
11.2	CloudHSM .....	137
11.3	SMTP Server .....	139
11.4	Permission Templates .....	140
11.4.1	Creating Permission Templates .....	141
11.4.2	Editing Permission Templates .....	142
11.4.3	Deleting Permission Templates .....	142
11.5	Domains .....	142
11.5.1	CAB Public Domain Suffix .....	143
11.5.2	Blacklists .....	143
11.5.3	Whitelists .....	144
11.5.4	Editing black and white lists .....	145
11.6	Realms .....	145
11.6.1	Add Realm .....	147
11.6.2	Edit Realm .....	148
12	Operator UI .....	164
12.1	Dashboard .....	164
12.1.1	Issued certificates .....	165
12.1.2	Expiring certificates .....	165
12.1.3	HSM partition status .....	166
12.1.4	Job Status .....	167
12.2	Manage .....	176
12.2.1	Users .....	176
12.2.2	Auditors .....	182

12.2.3	Clients.....	183
12.2.4	Rules.....	205
12.2.5	Notifications Templates.....	224
12.2.6	Registration Sources .....	241
12.2.7	HSMs .....	246
12.2.8	Permissions .....	254
12.2.9	Events.....	257
12.3	PKI.....	271
12.3.1	Certificate Policy Templates .....	272
12.3.2	Entities .....	299
13	Auditor UI.....	414
14	Registration UI .....	415
15	SCION .....	416
15.1	Protocol Adapter Responsibilities .....	416
15.2	Business Processes .....	416
15.2.1	AS Certificate Renewal.....	416
15.2.2	Frontend API .....	420
15.2.3	API Reference.....	423
16	REST API .....	429
16.1	Roles and Permissions .....	430
16.2	API Key.....	430
16.2.1	API Key Rollover .....	430
16.3	Authentication.....	431
16.3.1	JWT Generation .....	431
16.3.2	HTTP Request.....	431
16.3.3	OpenAPI v3 specification .....	432
17	Migrating SwissPKI v1 to SwissPKI v2 .....	433
17.1	Requirements .....	433
17.2	Procedure .....	433

17.2.1	Changes in TOTP length .....	436
17.2.2	Changes in Notification Tags.....	437
17.3	Microsoft Policy Mappings .....	439

## 1 Introduction

SwissPKI™ is a Public Key Infrastructure which delivers robust hardware based centralized key management backed up by strong cryptography to protect your business processes.

The solution addresses large scale cryptographic key management life cycle, online hardware-to-hardware key distribution, tamper proof audit as well as usage logs for compliance with standards and covers the complete certificate and key management life cycle.

SwissPKI™ integrates with the Primus Cloud or On-Premises HSMs, Thales, Kryptus and ARCA, taking full advantage of the built-in backup and replication mechanisms, reduces your operational overhead, reduces costs, and increases security.

SwissPKI™ is a feature rich, fully integrated Public Key Infrastructure service which helps expand your enterprise security: from large scale deployments to embedded HSM solutions, the solution provides all necessary out-of-the box components to increase your digital security in a safe, simple, and quick way.

Deploy single or complex lattice interconnected Certification Authorities to set up the essential trust between your users and systems. Keep your authority keys safe in the Cloud or on dedicated HSMs. The solution features single or multi-tenant configurations, on premises or cloud deployments as well as single or clustered HSMs.

SwissPKI™ helps you keep your certificates up-to-date and maintain complete visibility over them across issuing authorities. You can assign roles such as registration officer, authorizer, or auditor to trusted persons who can manage issuance, renewal, or recovery to streamline your organization's work flows to control each certificate management phase. In addition to the certificate policy management available out-of-the-box, you can provide your own micro-services on a policy basis to control and validate certificate content.

## 1.1 Standards

SwissPKI™ supports issuance and management of publicly trusted and qualified certificates. Its implementation is governed by the following standards and specifications:

- ✓ “Certificate Issuing and Management Components Protection Profile” defines requirements for components that issue, revoke, and manage public key certificates, such as X.509 public key certificates. The requirements are specified in the Common Criteria (CC).
- ✓ ETSI CAs issuing Qualified Certificates meeting requirements of Regulation.
- ✓ ETSI CAs issuing Web Site certificates meeting requirements of the CA/Browser Forum documents.
- ✓ ETSI Other Trust services including timestamping and CAs issuing certificates other than qualified certificates
- ✓ Mozilla CA Browser Forum Baseline Requirements and Network and Certificate System Security Requirements (CT Log, DNS Owner Checks and CAA Checks)
- ✓ Swiss ZertES and TAV recommendations
- ✓ X.509v3 RFCs



## 2 What is new in SwissPKI v2

### 2.1 What is new in SwissPKI version 2.0

The major features added to this version are:

- ✓ PKI entities <sup>1</sup> are modularized and can run as single applications and scale horizontally. Furthermore, PKI modules can be organized <sup>2</sup> into one or more applications (packaged modules support horizontal scaling).



- ✓ Asynchronous certificate issuance processing for long running validations processes such as DNS Owner checks.
- ✓ Convenience services such as CDP and AIA HTTP/S end points to ease dissemination of CRLs and CA certificates
- ✓ Improved Certificate Policy editor UX
- ✓ Fine grained Create/Read/Update/Delete (CRUD) permissions management for the PKI roles <sup>3</sup>.
- ✓ Support for multi CloudHSM, HSM partition clustering and HSM partition load distribution
- ✓ Support for Securosys, Thales, ARCA and Kryptus KNet HSMs
- ✓ Multi login capability for users with username/TOTP, LDAP, OpenID Connect and Kerberos
- ✓ Support for PostgreSQL 12.x and 14.x with write and read only clustering
- ✓ OpenAPI v3 REST API specification for Administrators, Operators and Registration
- ✓ Nexus repository for distributing release updates
- ✓ Simplified HELM Charts deployments

<sup>2</sup> Please contact [support@swisspki.com](mailto:support@swisspki.com) to obtain packaged modules in an application.

<sup>3</sup> PKI Admin, CA Operator, RA, Auditor, RA Operator and Authorizer roles

## 2.2 What is new in SwissPKI version 2.1

- ✓ Everything from SwissPKI version 2.0
- ✓ Option to print PDF information about Realms, Clients, Users, Permission Templates, Policy Templates and Policy Instances including validation rules.
- ✓ Integration of SwissSign's new Public Trust Managed PKI account. The SwissSign public trust certificates are managed directly using SwissPKI Operator and RA UI (Web UIs), including the pre validated domain names. You need one or more Managed PKI RA user account and corresponding shared secret.
- ✓ Internal architecture improvements for HSM connection management and throughput. Processes requiring access to HSM connections can manage hundreds of parallel HSM partition connections without locking.
- ✓ Improved HSM partition PIN resets synchronization across distributed PKI processes
- ✓ Improved performance for database queries and large data tables using PostgreSQL partitioning.

## 2.3 What is new in SwissPKI version 2.2

- ✓ CAB S/MIME pseudo validation rule to enforce unique pseudos per organization.
- ✓ Display product information in RA UI and SwissSign mapped products.
- ✓ Partitioning of document registration table for archiving. Partitions are created per year until 2050. For users storing many registration documents, an S3 option can be enabled to store the registration documents. This option is enabled at the Realm.
- ✓ Optional revocation code generation for self service revocation. The client policy mapping can enable/disable the generation of revocation codes. When enabled, a link is added to the certificate issuance email. The link is used by the recipient to revoke the certificate via the self service page.
- ✓ CAB S/SMIME certificate policy key generation type added to let the CA generate the end user PKCS#12 PIN. The PIN is sent via email link to the recipient. When opened, the certificate issuance process gets finalized. The end user copies the PIN from the web page. The issuance email contains the PKCS#12 which can be opened with the delivered PIN
- ✓ Option to download issued certificate chain in PEM format (OpenSSL).
- ✓ Allow revocation authorizations to be rejected multiple times
- ✓ ETSI policy assured value short term extension in the policy editor
- ✓ Add friendly name to issued PKCS#12 using the certificate's common name or email when available
- ✓ Option to configure the validity of the end user email validation link in compliance.conf
- ✓ Additional notification recipients can be added to individual requests. Supported additional notification recipients are: DNS CAB validation, DNS CAB email validation link, certificate issuance, certificate revocation and authorizations in addition to the existing certificate renewal.
- ✓ Extended certificate issuance RA API including Subject DN, SAN (email, dns, UPN), extension overrides. Additional notification recipients, optional certificate validity override and

registration documents. An option to issue synchronously or asynchronously is also available as well as a comment field. Support for requesting PKCS#12 via RA API is also available.

- ✓ Option to let the RA (UI and API) include base domain for wildcard or www prefix to the requested DNS
- ✓ New RA API method to publish/unpublish certificate from publication destinations (requires certificate publication enabled for the issuing CA)
- ✓ Notifications can handle additional recipients. Additional recipients are added/removed from individual certificate orders. The option is available both in the RA UI & API

## 2.4 Changes between SwissPKI v1 and v2

Several modifications and naming conventions have changed between SwissPKI v1 and v2:

- ✓ Multi tenancy naming convention change: the 'Client' (or tenant) in SwissPKI v1 is renamed 'Realm' in SwissPKI v2.
- ✓ Group naming convention change: the 'Group' in SwissPKI v1 is renamed 'Client' in SwissPKI v2.
- ✓ The Community concept in SwissPKI v1 is removed in SwissPKI v2. All PKI entities are linked to a 'Realm' instead of a 'Community'
- ✓ The 'Graph' view of the PKI entities in SwissPKI v1 is removed. A 'Tree View' of the PKI entities is used as a replacement in SwissPKI v2.
- ✓ Support for MariaDB and MySQL is dropped in SwissPKI v2. Please refer to section 17 *Migrating SwissPKI v1 to SwissPKI v2* for details

## 3 Deployment Requirements

### 3.1 Prerequisites

#### 3.1.1 Database and Storage

##### 3.1.1.1 Database

PostgreSQL version: 15.X (see SwissPKI v2 requirements document)

- Provided as an external service
- At least 1 Master instance in R/W mode
- At least 2 Replica instances in R mode

##### 3.1.1.2 Cache

Redis version: 6.2 (see SwissPKI v2 requirements document)

- Provided as an external service
- At least 3 Master instances
- At least 3 Replica instances

##### 3.1.1.3 Message Queue

RabbitMQ version: 3.10.x (see SwissPKI v2 requirements document)

- Provided as an external service
- At least 1 Master instance
- At least 3 Replica instances

### 3.1.2 Infrastructure

#### 3.1.2.1 LDAP Server

Optional LDAP Server supporting LDAPv3 protocol for Certificate and CRL/ARL publishing including one user account with R/W permissions enabled on certificate and CRL schema.

#### 3.1.2.2 DNS Server

Optional access to DNS (recursive requests) for SwissPKI applications

#### 3.1.2.3 SMTP Server

One SMTP account with host, port, username, password, and TLS enabled

### 3.1.2.4 OpenID Connect Server

Optional Identity Provider for User login (Registration Officers, Authorizers) with API Host, ClientId and ClientSecret

### 3.1.3 Hardware Security Modules

Optional Securosys Primus E or X Series HSMs on premises or CloudsHSM with firmware 2.7.x or higher, Thales LunaSA 7.x HSMs, ARCA 2.x HSMs or Kryptus KNet.

Ideal requirements:

1. at least 2 HSMs (cluster)
2. no L4 proxy between SwissPKI and HSMs.
3. Remote access control

## 3.2 Bare Metal

Latest Ubuntu 22.04.x LTS or RHEL 8.x

Windows: not supported

Latest OpenJDK or Oracle JRE 11.x for the selected platform

Module image size: 270MB

Minimal RAM per deployed module: 2GB, no upper limit

Minimal CPU per deployed module: 500m, no upper limit

Network throughput: 1000 requests/sec

Module packaging: DEB or RPM

## 3.3 Kubernetes

Kubernetes cluster Kubernetes/Rancher/OpenShift/AWS/GCloud

Supported version: >= 1.24

SwissPKI v2 Docker images are built using the latest openjdk Docker image and is available on Nexus repository <https://nexus.libc.ch> (requires a user account).

HELM Charts are available at <https://helm.libc.ch>.

Please visit <https://support.swisspki.com> or send an email to [support@swisspki.com](mailto:support@swisspki.com) to obtain the detailed requirement PDF document 'SwissPKI Requirements Kubernetes Deployment.'

## 4 Configuration

Initial deployment, configuration, and updates instructions.

### 4.1 Bare Metal

Please visit <https://support.swisspki.com> or send an email to [support@swisspki.com](mailto:support@swisspki.com) to obtain the detailed configuration PDF document 'SwissPKI Deployment for Bare Metal.'

### 4.2 Kubernetes

Please visit <https://support.swisspki.com> or send an email to [support@swisspki.com](mailto:support@swisspki.com) to obtain the detailed configuration PDF document 'SwissPKI Deployment for Kubernetes.'

HELM Charts are available at <https://helm.libc.ch>.

### 4.3 Logging

Logging configuration is available at <http://logback.qos.ch/manual/architecture.html>

#### 4.3.1 Bare Metal

Each module has its log configuration located in `/opt/<module>/conf/logback.xml`

#### 4.3.2 Kubernetes

Each module logs to STDOUT with the following pattern

```
yyyy-MM-dd HH:mm:ss.SSS [thread] - [level] package.class - message
```

#### *Example*

```
2020-10-22 08:33:43.030 [play-dev-mode-akka.actor.default-dispatcher-7] - [info] ch.libc.shared.modules.SharedBootStartImpl - Database migration enabled
```

## 4.4 Database Initialization

Initializing the SwissPKI DB is performed by starting first the Admin UI application. For this, the DB **user** configured in the **db.conf** MUST have CREATE, ALTER, DROP, INSERT, SELECT and UPDATE rights on the SwissPKI DB schema.

The DB schema and user must be created first by a DB admin.

SwissPKI initializes the DB schema automatically on initial startup if DB schema is not present. The SwissPKI DB versioning information is kept in table `t_schema_version`.

No further steps are required.

## 4.5 Database Migration

SwissPKI version information is stored in the DB table `t_schema_version`

Whenever a new release of SwissPKI requiring DB migration is made available, the DB schema gets automatically migrated when either one of Operator UI or Admin UI is deployed and started. Thus, the application's DB user (set in the **db.conf** of the deployment) MUST have CREATE, ALTER, DROP, INSERT, SELECT and UPDATE rights on the SwissPKI DB schema.

When a DB migration is required while installing a new release, then the information is made available in the release notes.

No further steps are required.

### 4.5.1 Migrating the DB

1. Perform a full DB backup
2. Deploy the new SwissPKI in the following order
  - a. First the Admin UI or Operator UI
    - i. The DB migration will occur automatically at start up
  - b. Deploy any other SwissPKI module in any order

## 5 General PKI Considerations

During designing, implementing, and deploying a PKI project, you should consider documenting specific project phases before issuing your first certificates. You will find below some practical advice for the implementation of your PKI.

Documents	Purpose
<b>Role definition guide</b>	<p>Document the separate roles and responsibilities involved in the PKI (project, service owner, operations, support) such as Security Officers, System Administrators, Key Administrators, Key Ceremony Master, Key Ceremony Witnesses, CA Manager, CA Operators, Registration Officers, and other involved roles important to your organization depending on the scope and type of PKI.</p> <p>This document can also define access control rules such as 4-eyes principles or access matrices.</p>
<b>Use Cases</b>	<p>Document the certificate registration, revocation, renewal, and authorization processes whether manual or automatic.</p> <p>Identify where you require notifications and how often as well as which groups of people to notify with which content.</p> <p>Identify if you need additional authorization gates or if specific certificate information content validation is required.</p> <p>Also, consider separating systems/machines from end users.</p>
<b>Certificate policy templates</b>	<p>Document the content of the certificates such as Subject naming, certificate extensions, validity, key sizes, key types whether software or hardware and associated usage.</p>
<b>Object Identifiers</b>	<p>Use Object Identifiers to categorize your certificates defined in the certificate policy templates. For example, use different numbering schemes to identify certificates issued from test, acceptance, or production environments.</p>
<b>Acceptance document</b>	<p>Document an acceptance protocol where you can validate manually or automatically the processes defined in your Use Cases as well as the conformity of the issued certificates and their content.</p>
<b>Certificate practice statement</b>	<p>For larger PKIs or Public Trust PKIs, you describe the practice for issuing and managing the infrastructure. Elements of this document may include detailed descriptions of the issuance,</p>

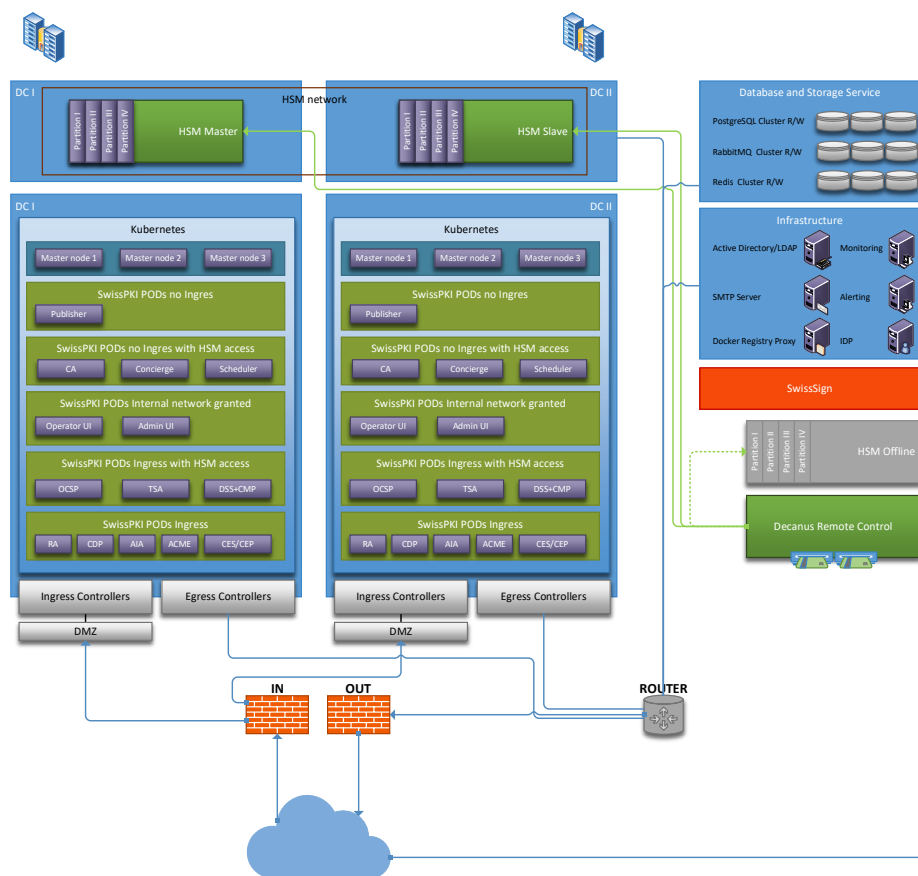


	revocation, publication, and renewal processes to help certified users or entities assess the reliability of the Certification Authority/ies.
<b>Password management</b>	Do not forget to list all passwords, PINs and secrets required by your PKI environment. Whether you use one or multiple list, define password and PIN rules and their validities. Keep the documents in a secure place and accessible to authorized roles. The same applies to smart cards if you plan using hardware security modules (HSMs)
<b>System control</b>	Before setting up your PKI, establish a system control document (check list) to verify that all services and required components are configured and accessible.
<b>Key ceremony</b>	Create a key ceremony document which will lead you through the configuration and issuance of the actual PKI's Certificate Authorities. The Certificate Authorities' validity being issued for a prolonged period, it is always difficult to recall what was done five years in the past. Involve all mandatory roles when initializing your PKI.
<b>Operations manuals</b>	Document the setup, configuration, backup, restore, updates, firmware updates, monitoring, and operations' processes. Train backup and restore procedures once a year, especially if you have productive HSMs.
<b>Environment description</b>	Document the technical and organizational environments such as test, acceptance, and production.
<b>End user instructions</b>	Based on your Use Cases, inform and train the involved groups of people.

## 6 Architecture Overview

The following illustration is an example of a SwissPKI deployment on an active/active Kubernetes cluster in two separate data centers.

1. Redundant active/active HSM cluster. PKI components with HSM partition access will automatically connect to the available HSM if one device is offline. Offline HSM with replicated partitions. This HSM can be brought online to synchronize newly generated keys on the production partitions for backup purposes. Individual partitions can be brought on or offline. Remote HSM management in data centers provided by Decanus device.
2. Data storage and infrastructure services provided as an external service to SwissPKI deployed on the Kubernetes clusters.
3. SSL client/server proxied tunnel to SwissSign service for issuing public trust certificates. Issued certificates are managed through SwissPKI via the integrated CMS interface.
4. Network partitioned SwissPKI components with access to HSMs and external services where granted.



## 6.1 Components

### 6.1.1 Administration Portal (Admin)

Web portal for SwissPKI administrator. This portal is used to setup general PKI settings, Realms, CA Operators and Permissions templates for PKI Administrators and CA Operators roles.

### 6.1.2 Operator Portal (Operator)

Web portal for SwissPKI CA Operators and Auditors for managing the PKI and associated PKI Entities (OCSP, TSA, DSS, CMP, MSCA, ACME, Publisher, AIA, CDP, CRL publication rules, RA Operators, Authorizers, Clients, Certificate policies and Permission templates for RA Officers, Authorizers and Auditor roles).

### 6.1.3 Registration Authority (RA)

Web portal accessible to the Registration Officers and Authorizers, including authenticated and authorized REST API calls, for managing, issuing, and revoking certificates.

### 6.1.4 Certification Authority (CA)

Root and Issuing certification authorities signing service of certificates and CRLs/ARLs.

### 6.1.5 Microsoft CES/CEP (MSCA)

Microsoft CES (registration) and CEP (policy management) HTTPS Services for AD autoenrollment exposed to clients/users. This service requires an additional Microsoft CES/CEP service running on the client's Microsoft Domain.

Supports Windows 8 and higher client devices and Windows Server 2008R2 and higher

### 6.1.6 Automatic Certificate Management Environment (ACME)

RFC8555 ACME HTTPS Service exposed to clients.

### 6.1.7 Online Certificate Status Protocol (OCSP)

RFC6960 OCSP HTTP Service for real time certificate validation.

### 6.1.8 Time Stamp Authority (TSA)

RFC3161 TSA HTTPS Service for producing digital time stamps.

### 6.1.9 SCEP/NDES (SCEP)

RFC8894 SCEP Simple Certificate Enrolment Protocol. HTTPS Service registering certificates.

### **6.1.10 CRL Distribution Point (CDP)**

Convenience HTTP server serving the latest CRLs produced by the Certification Authorities. The service facilitates dissemination of the produced CRLs by offering unique URLs which can be used in the certificate CDP extension as a replacement or as an addition to HTTP servers serving copies of published CRLs.

### **6.1.11 Authority Information Access (AIA)**

Convenience HTTP/ HTTPS server serving the Certification Authorities certificates. The service facilitates dissemination of the CA certificates by offering unique URLs which can be used in the certificate AIA extension as a replacement or in addition to HTTP servers serving copies of the certificates.

### **6.1.12 SCION PKI Adapter (SCION)**

SCION AS Identity certificate renewal. HTTPS Service for renewing SCION AS Identities (<https://www.scion-architecture.net/pdf/SCION-book.pdf>). Scalability, Control, and Isolation on Next-generation networks.

### **6.1.13 Certificate Management Protocol (CMP)**

RFC 6712 CMP HTTPS Service exposed to clients/users. Note that this service only exposes certificate registration and revocation.

### **6.1.14 Document Signer Server (DSS)**

Document Signer Server HTTPS Service exposed to client/users for signing documents in XML, PDF, PKCS#7 and/or ASiC formats according eIDAS signature formats.

### **6.1.15 Publisher**

SwissPKI service for publishing produced CRLs/ARLs and certificates to various destinations such as LDAP, SFTP and file systems. Publication of certificates and CRLs/ARLs is rule based on specific client individual settings. Certificate publication can be overruled in the client's certificate product settings.

### **6.1.16 Concierge**

Service used by the SwissPKI components for processing asynchronous requests (e.g., DNS Owner checks, CAA checks, sending S/MIME emails).

### **6.1.17 Scheduler**

Scheduler service for handling background tasks such as processing automatic renewal tasks, notifications, CRL issuance, HSM availability checks, statistics, and reports.

## 7 Security Considerations

### 7.1 General Security

Application/Service	Security
<p><b>PostgreSQL DB</b></p>	<p>Central data storage for all SwissPKI applications.</p> <p>The DB storage contains no sensitive data except for the following:</p> <ol style="list-style-type: none"> <li>1. SMTP, LDAP, AD, SFTP passwords</li> <li>2. HSM partition permanent PINs. Permanent PINs are exchanged for each partition when initializing the configuration using the Partition Setup PIN (configurable validity). The permanent PIN is therefore undisclosed to administrators, operators, auditors, and system administrators.</li> <li>3. CMC PFX PIN</li> <li>4. User 2FA shared secret and scratch codes</li> <li>5. Private keys for key generation rules using PKCS#12 key settings in certificate policy templates.</li> <li>6. API Keys associated to Roles</li> </ol> <p>Sensitive data is kept ciphered in the DB using AES256/GCM symmetric key and mode. The symmetric key is KDF2 with HMAC SHA256 derived from a salt and secret key using 65536 iteration rounds</p>
<p><b>Redis</b></p>	<p>Cache storage. No sensitive data is stored in the cache. The service does not require backup, the cache data being recreated on demand. Cache information such as search parameters (UI) are kept for max one hour after initial creation and updates. The cache is used by the SwissPKI UIs</p> <p>If Redis is not available at runtime, cache data is kept in process memory</p>

<b>SwissPKI modules</b>	Require PINs at startup of the process. The PINs are obtained from the K8s secrets when deployed on Kubernetes and from the configurations when deployed on bare metal from the <i>secrets.conf</i> and <i>play.conf</i> files
<b>HSM Partitions</b>	Storage of CA keys (RSA, EC)
<b>SonarQube</b>	Vulnerability and quality gates report available upon request
<b>Snyk</b>	CVE and vulnerability report available upon request

## 7.2 Connection Flows

### 7.2.1 Administrator UI

From	To	Protocol
Internal network granted	Administrator UI	HTTPS with 2FA
Administrator UI	LDAP OIDC	LDAPS with 2FA HTTPS with 2FA
Administrator UI	SMTP	TLS
Administrator UI	DB (Read/Write)	TLS
Administrator UI	Redis	TLS

### 7.2.2 Operator UI

From	To	Protocol
Internal network granted	Operator UI	HTTPS with 2FA
Operator UI	LDAP OIDC	LDAPS with 2FA HTTPS with 2FA
Operator UI	SMTP	TLS
Operator UI	DB (Read/Write)	TLS
Operator UI	LDAP	LDAPS with username/pin
Operator UI	HSM	TCP AES 256 GCM
Operator UI	RabbitMQ	TLS

### 7.2.3 Registration UI

From	To	Protocol
External network	Registration UI	HTTPS with 2FA
Registration UI	LDAP OIDC	LDAPS with 2FA HTTPS with 2FA
Registration UI	DB (Read only, except for Event table with write permission)	TLS
Registration UI	Concierge	TLS
Registration UI	DB	TLS
Registration UI	RabbitMQ	TLS

### 7.2.4 Certification Authority (CA)

From	To	Protocol
RabbitMQ	CA	TLS
CA	DB (Read/Write)	TLS
CA	HSM	TCP AES 256 GCM
Publisher	SFTP	SSH username/PIN
CA	RabbitMQ	TLS
Publisher	Filesystem	Group rw to file system

### 7.2.5 Automatic Certificate Management Environment (ACME)

From	To	Protocol
External network	ACME	HTTPS
ACME	DB (Read only, except for Event table with write permission)	TLS
ACME	RabbitMQ	TLS
ACME	Concierge	TLS

### 7.2.6 Microsoft CES/CEP (MSCA)

From	To	Protocol
External network	CES/CEP	HTTPS with signed RSA JWT
CES/CEP	DB (Read only, except for Event table with write permission)	TLS
CES/CEP	RabbitMQ	TLS
CES/CEP	Concierge	TLS

### 7.2.7 Online Certificate Status Protocol (OCSP)

From	To	Protocol
External network	OCSP	HTTP
OCSP	DB (Read only, except for Event table with write permission)	TLS
OCSP	HSM (dedicated partition)	TCP AES 256 GCM
OCSP	RabbitMQ	TLS

### 7.2.8 Time Stamp Authority (TSA)

From	To	Protocol
External network	TSA	HTTPS
TSA	DB (Read only, except for Event table with write permission)	TLS
TSA	HSM (dedicated partition)	TCP AES 256 GCM
TSA	RabbitMQ	TLS

### 7.2.9 SCEP/NDES (SCEP)

From	To	Protocol
External network	SCEP	HTTPS
SCEP	DB (Read only, except for Event table with write permission)	TLS
SCEP	RabbitMQ	TLS



SCEP	Concierge	TLS
------	-----------	-----

### 7.2.10 CRL Distribution Point (CDP)

From	To	Protocol
External network	CDP	HTTP
CDP	DB (Read only)	TLS

### 7.2.11 Authority Information Access (AIA)

From	To	Protocol
External network	AIA	HTTPS or HTTP
AIA	DB (Read only)	TLS

### 7.2.12 SCION PKI Adapter (SCION)

From	To	Protocol
External network	SCION	HTTPS
SCION	DB (Read only, except for Event table with write permission)	TLS
SCION	RabbitMQ	TLS
SCION	Concierge	TLS

### 7.2.13 Certificate Management Protocol (CMP)

From	To	Protocol
External network	CMP	HTTPS with signed (RSA-PSS or ECDSA) and encrypted (AES 256 GCM) payloads
CMP	DB (Read only, except for Event table with write permission)	TLS
CMP	RabbitMQ	TLS

### 7.2.14 Document Signer Server (DSS)

From	To	Protocol
External network	DSS	HTTPS with signed (RSA-PSS or ECDSA) and encrypted (AES 256 GCM) payloads
DSS	DB (Read only, except for Event table with write permission)	TLS
DSS	HSM (dedicated partition)	TCP AES 256 GCM
DSS	RabbitMQ	TLS

### 7.2.15 Publisher

From	To	Protocol
Internal network		
Publisher	DB (Read/Write to Event table)	TLS
Publisher	LDAP	LDAPS username/PIN
Publisher	SFTP	SSH username/PIN
Publisher	Filesystem	Group rw to file system

### 7.2.16 Concierge

From	To	Protocol
RabbitMQ	Concierge	TLS
Concierge	DB (Read/Write)	TLS
Concierge	RabbitMQ	TLS
Concierge	SMTP	TLS

## 7.2.17 Scheduler

From	To	Protocol
Scheduler	DB (Read/Write)	TLS
Scheduler	RabbitMQ	TLS
Scheduler	HSM	TCP AES 256 GCM
Scheduler	RabbitMQ	TLS
Scheduler	Internet (CAB domain list downloads and external certificate CRL check)	HTTP/HTTPS

### 7.3 Health Checks

Each module disposes of liveness, readiness and start up probes

```
GET http(s)://<DNS or IP>/<module>/healthcheck/ready
```

```
GET http(s)://<DNS or IP>/<module>/healthcheck/alive
```

```
GET http(s)://<DNS or IP>/<module>/healthcheck/roundtrip (invokes all associated sub modules for the selected application)
```

### 7.4 Roles and Permissions

Access control to the SwissPKI functionalities is managed through roles and permissions.

#### 7.4.1 Roles

SwissPKI distinguishes three groups of roles:

Groups	Roles	Description
<b>Administrators</b>	PKI_ADMIN	The top-level PKI Administrator with access to the Administration UI for configuring global settings and Realms.
<b>Operators</b>	CAO (CA Operator) AUDITOR	The Realm operator with access to the PKI Entities The Realm auditor with access to the events
<b>Clients</b>	RAO (Registration Officer)  AUTHORIZER	End user/client role for issuing and managing their certificates  End user/client role for authorizing requests

**Note:** A SwissPKI user can have multiple roles, except for the PKI Administrator which can be assigned only a PKI\_ADMIN role. Furthermore, users created within Realms cannot access other Realms created on a same SwissPKI deployment. If you want to have a user access two different Realms, then you must create a distinct user per Realm.

## 7.4.2 Permissions

Roles are assigned permissions through permission templates. Permission templates are lists of permissions which can be configured according to your needs. Permissions are expressed as CRUD Create/Read/Update/Delete expressions. That is each function within SwissPKI has a set of 4 permissions except for a few functionalities which do not necessitate create or delete permissions.

**Note:** When initializing SwissPKI, default templates for each role with all permissions are generated to help you get started. Also, a user logged in with a role cannot modify its own role and/or permissions. To modify your own role/permissions, a user with the same access control level can modify your role/permissions assuming this user has the granted permissions to modify roles/permissions.

### 7.4.2.1 Permissions associated with the PKI Admin Role

Permission Groups	Description
PKI_ADMIN__ADMIN__CREATE PKI_ADMIN__ADMIN__UPDATE PKI_ADMIN__ADMIN__DELETE PKI_ADMIN__ADMIN__VIEW	Create, read, update, delete PKI Administrator
PKI_ADMIN__PERMISSION__CREATE PKI_ADMIN__PERMISSION__UPDATE PKI_ADMIN__PERMISSION__DELETE PKI_ADMIN__PERMISSION__VIEW	Create, read, update, delete permissions in permission templates
PKI_ADMIN__TOTP__CREATE PKI_ADMIN__TOTP__UPDATE PKI_ADMIN__TOTP__DELETE PKI_ADMIN__TOTP__VIEW	Create, read, update, delete PKI Administrator TOTP
PKI_ADMIN__API_KEY__CREATE PKI_ADMIN__API_KEY__UPDATE PKI_ADMIN__API_KEY__DELETE PKI_ADMIN__API_KEY__VIEW	Create, read, update, delete Administrator API Key
PKI_ADMIN__SMTP__UPDATE PKI_ADMIN__SMTP__VIEW	Read, update administration SMTP settings
PKI_ADMIN__CLOUD_HSM__UPDATE PKI_ADMIN__CLOUD_HSM__VIEW	Read, update CloudHSM proxy settings
PKI_ADMIN__REALM__CREATE PKI_ADMIN__REALM__UPDATE PKI_ADMIN__REALM__DELETE PKI_ADMIN__REALM__VIEW	Create, read, update, delete Realm
PKI_ADMIN__REALM_CAO__CREATE PKI_ADMIN__REALM_CAO__UPDATE	Create, read, update, delete Realm CA Operator

PKI_ADMIN__REALM_CAO_DELETE PKI_ADMIN__REALM_CAO_VIEW	
PKI_ADMIN__REALM_CAO_PERMISSION_CREATE PKI_ADMIN__REALM_CAO_PERMISSION_UPDATE PKI_ADMIN__REALM_CAO_PERMISSION_DELETE PKI_ADMIN__REALM_CAO_PERMISSION_VIEW	Create, read, update, delete Realm permission templates
PKI_ADMIN__REALM_CAO_TOTP_CREATE PKI_ADMIN__REALM_CAO_TOTP_UPDATE PKI_ADMIN__REALM_CAO_TOTP_DELETE PKI_ADMIN__REALM_CAO_TOTP_VIEW	Create, read, update, delete Realm CA Operator TOTP
PKI_ADMIN__REALM_CAO_API_KEY_CREATE PKI_ADMIN__REALM_CAO_API_KEY_UPDATE PKI_ADMIN__REALM_CAO_API_KEY_DELETE PKI_ADMIN__REALM_CAO_API_KEY_VIEW	Create, read, update, delete Realm CA Operator API Key
PKI_ADMIN__REALM_SMTP_CREATE PKI_ADMIN__REALM_SMTP_UPDATE PKI_ADMIN__REALM_SMTP_DELETE PKI_ADMIN__REALM_SMTP_VIEW	Create, read, update, delete Realm SMTP settings
PKI_ADMIN__REALM_CNG_CREATE PKI_ADMIN__REALM_CNG_UPDATE PKI_ADMIN__REALM_CNG_DELETE PKI_ADMIN__REALM_CNG_VIEW	Create, read, update, delete Realm CNG settings
PKI_ADMIN__REALM_ANCHOR_CREATE PKI_ADMIN__REALM_ANCHOR_UPDATE PKI_ADMIN__REALM_ANCHOR_DELETE PKI_ADMIN__REALM_ANCHOR_VIEW	Create, read, update, delete Realm trust anchors
PKI_ADMIN__REALM_LINTER_CREATE PKI_ADMIN__REALM_LINTER_UPDATE PKI_ADMIN__REALM_LINTER_DELETE PKI_ADMIN__REALM_LINTER_VIEW	Create, read, update, delete Realm Linters
PKI_ADMIN__REALM_CT_LOG_FAMILY_CREATE PKI_ADMIN__REALM_CT_LOG_FAMILY_UPDATE PKI_ADMIN__REALM_CT_LOG_FAMILY_DELETE PKI_ADMIN__REALM_CT_LOG_FAMILY_VIEW	Create, read, update, delete Realm CT Log families
PKI_ADMIN__REALM_S3_CREATE PKI_ADMIN__REALM_S3_UPDATE PKI_ADMIN__REALM_S3_DELETE PKI_ADMIN__REALM_S3_VIEW	Read, update Realm S3 configuration
PKI_ADMIN__REALM_SCION_UPDATE PKI_ADMIN__REALM_SCION_VIEW	Read, update Realm SCION Identity Repository validation service
PKI_ADMIN__PERMISSION_TEMPLATE_CREATE PKI_ADMIN__PERMISSION_TEMPLATE_UPDATE PKI_ADMIN__PERMISSION_TEMPLATE_DELETE PKI_ADMIN__PERMISSION_TEMPLATE_VIEW	Create, read, update, delete permission templates

PKI_ADMIN_DOMAINS_CREATE PKI_ADMIN_DOMAINS_UPDATE PKI_ADMIN_DOMAINS_DELETE PKI_ADMIN_DOMAINS_VIEW	Create, read, update, delete blacklists.
--	--

### 7.4.2.2 Permissions associated with the CA Operator Role

Permission Groups	Description
CAO_CA_CREATE CAO_CA_UPDATE CAO_CA_DELETE CAO_CA_VIEW CAO_CA_REVOKE	Create, read, update, delete CA entities. Enabled CAO_CA_REVOKE to allow the CA Operator to revoke CA certificates from the Operator UI
CAO_CA_ENTITY_CREATE CAO_CA_ENTITY_UPDATE CAO_CA_ENTITY_DELETE CAO_CA_ENTITY_VIEW	Create, read, update, delete PKI Entities
CAO_CA_AIA_CREATE CAO_CA_AIA_UPDATE CAO_CA_AIA_DELETE CAO_CA_AIA_VIEW	Create, read, update, delete AIA connection points
CAO_CA_CDP_CREATE CAO_CA_CDP_UPDATE CAO_CA_CDP_DELETE CAO_CA_CDP_VIEW	Create, read, update, delete CA CDP connection points
CAO_CA_CERTIFICATES_VIEW	Query/view issued certificates
CAO_CA_MICROSOFT_CREATE CAO_CA_MICROSOFT_UPDATE CAO_CA_MICROSOFT_DELETE CAO_CA_MICROSOFT_VIEW	Create, read, update, delete Microsoft CES/CEP instance for CAs
CAO_CA_CRL_CREATE CAO_CA_CRL_UPDATE CAO_CA_CRL_DELETE CAO_CA_CRL_VIEW	Create, read, update, delete CA CRLs/ARLs Create/Update refers to CRL publication Rules
CAO_CA_POLICIES_CREATE CAO_CA_POLICIES_UPDATE CAO_CA_POLICIES_DELETE CAO_CA_POLICIES_VIEW	Create, read, update, delete CA certificate policy instances
CAO_CA_SETTINGS_UPDATE CAO_CA_SETTINGS_VIEW	
CAO_CA_CROSS_SIGN_CREATE CAO_CA_CROSS_SIGN_UPDATE	Create, read, update, delete CA cross signed requests

<b>CAO__CA_CROSS_SIGN__DELETE</b> <b>CAO__CA_CROSS_SIGN__VIEW</b>	
<b>CAO__TSA__CREATE</b> <b>CAO__TSA__UPDATE</b> <b>CAO__TSA__DELETE</b> <b>CAO__TSA__VIEW</b>	Create, read, update, delete TSA entities
<b>CAO__OCSP__CREATE</b> <b>CAO__OCSP__UPDATE</b> <b>CAO__OCSP__DELETE</b> <b>CAO__OCSP__VIEW</b>	Create, read, update, delete OCSP entities
<b>CAO__DSS__CREATE</b> <b>CAO__DSS__UPDATE</b> <b>CAO__DSS__DELETE</b> <b>CAO__DSS__VIEW</b>	Create, read, update, delete Document Signer entities
<b>CAO__ACME__CREATE</b> <b>CAO__ACME__UPDATE</b> <b>CAO__ACME__DELETE</b> <b>CAO__ACME__VIEW</b>	Create, read, update, delete ACME entities
<b>CAO__MSCA__CREATE</b> <b>CAO__MSCA__UPDATE</b> <b>CAO__MSCA__DELETE</b> <b>CAO__MSCA__VIEW</b>	Create, read, update, delete Microsoft CES/CEP entities
<b>CAO__CMP__CREATE</b> <b>CAO__CMP__UPDATE</b> <b>CAO__CMP__DELETE</b> <b>CAO__CMP__VIEW</b>	Create, read, update, delete CMP entities
<b>CAO__CLIENT_DOMAIN__CREATE</b> <b>CAO__CLIENT_DOMAIN__UPDATE</b> <b>CAO__CLIENT_DOMAIN__DELETE</b> <b>CAO__CLIENT_DOMAIN__VIEW</b>	Create, read, update, delete Client Domain  s (for DNS and Email domain validation)
<b>CAO__TECHNICAL_CONTACT__CREATE</b> <b>CAO__TECHNICAL_CONTACT__UPDATE</b> <b>CAO__TECHNICAL_CONTACT__DELETE</b> <b>CAO__TECHNICAL_CONTACT__VIEW</b>	Create, read, update, delete Client technical contact
<b>CAO__CRL__GENERATE</b> <b>CAO__CRL__GENERATE_LAST</b>	Generate CRL, generate last CRL
<b>CAO__CLIENT__CREATE</b> <b>CAO__CLIENT__UPDATE</b> <b>CAO__CLIENT__DELETE</b> <b>CAO__CLIENT__VIEW</b>	Create, read, update, delete Clients
<b>CAO__AUTHORIZER__CREATE</b> <b>CAO__AUTHORIZER__DELETE</b> <b>CAO__AUTHORIZER__VIEW</b>	Manage the Authorizers in the Realm for Clients



<b>CAO__RAO__CREATE</b> <b>CAO__RAO__DELETE</b> <b>CAO__RAO__VIEW</b>	Manage RA Officers in the Realm for the Clients
<b>CAO__USER__CREATE</b> <b>CAO__USER__UPDATE</b> <b>CAO__USER__DELETE</b> <b>CAO__USER__VIEW</b>	Create, read, update, delete users in the Realm
<b>CAO__USER_PERMISSION__UPDATE</b> <b>CAO__USER_PERMISSION__VIEW</b>	Update, read user permissions in the Realm
<b>CAO__USER_API_KEY__UPDATE</b> <b>CAO__USER_API_KEY__VIEW</b>	Update, read user API Keys in the Realm
<b>CAO__USER_TOTP__UPDATE</b> <b>CAO__USER_TOTP__VIEW</b>	Update, read user TOTP in the Realm
<b>CAO__AUDITOR__CREATE</b> <b>CAO__AUDITOR__DELETE</b> <b>CAO__AUDITOR__VIEW</b>	Manage Auditors for the Realm
<b>CAO__POLICY_VALIDATION__CREATE</b> <b>CAO__POLICY_VALIDATION__UPDATE</b> <b>CAO__POLICY_VALIDATION__DELETE</b> <b>CAO__POLICY_VALIDATION__VIEW</b>	Create, read, update, delete validation services associated to Clients
<b>CAO__CMC_SDN__CREATE</b> <b>CAO__CMC_SDN__UPDATE</b> <b>CAO__CMC_SDN__DELETE</b> <b>CAO__CMC_SDN__VIEW</b>	Create, read, update, delete CMC Serial/Subject DN attributes associated to Clients
<b>CAO__CMP_TRUST_ANCHOR__CREATE</b> <b>CAO__CMP_TRUST_ANCHOR__DELETE</b> <b>CAO__CMP_TRUST_ANCHOR__VIEW</b>	Create, read, update, delete CMP trust anchors for client accounts
<b>CAO__POLICY__CREATE</b> <b>CAO__POLICY__UPDATE</b> <b>CAO__POLICY__DELETE</b> <b>CAO__POLICY__VIEW</b>	Create, read, update, delete policy instances (products) mapped between a CA and Clients
<b>CAO__POLICY_TEMPLATE__CREATE</b> <b>CAO__POLICY_TEMPLATE__UPDATE</b> <b>CAO__POLICY_TEMPLATE__DELETE</b> <b>CAO__POLICY_TEMPLATE__VIEW</b> <b>CAO__POLICY_TEMPLATE__ACTIVATE</b>	<p>Create, read, update, delete policy templates</p> <p>When 4 eyes principle is enabled on the Operator UI, CAOs can ACTIVATE modified certificate policy templates when <code>CAO__POLICY_TEMPLATE__ACTIVATE</code> is enabled</p>
<b>CAO__PKI_ENTITIES__VIEW</b>	View the PKI Entities
<b>CAO__PUBLISHER__CREATE</b> <b>CAO__PUBLISHER__UPDATE</b>	Create, read, update, delete Publisher instances

<b>CAO__PUBLISHER__DELETE</b> <b>CAO__PUBLISHER__VIEW</b>	
<b>CAO__RULE__CREATE</b> <b>CAO__RULE__UPDATE</b> <b>CAO__RULE__DELETE</b> <b>CAO__RULE__VIEW</b>	Create, read, update, delete rules
<b>CAO__NOTIFICATION__CREATE</b> <b>CAO__NOTIFICATION__UPDATE</b> <b>CAO__NOTIFICATION__DELETE</b> <b>CAO__NOTIFICATION__VIEW</b>	Create, read, update, delete notifications
<b>CAO__CERTIFICATE__VIEW</b>	Query/view certificates
<b>CAO__CERTIFICATE__UPDATE</b>	Update certificate settings and information
<b>CAO__CERTIFICATE__REVOKE</b>	Revoke certificates
<b>CAO__CERTIFICATE__PUBLISH</b>	Publish certificates
<b>CAO__HSM__CREATE</b> <b>CAO__HSM__UPDATE</b> <b>CAO__HSM__DELETE</b> <b>CAO__HSM__VIEW</b>	Create, read, update, delete HSM partitions and host configurations
<b>CAO__PERMISSION_TEMPLATE__CREATE</b> <b>CAO__PERMISSION_TEMPLATE__UPDATE</b> <b>CAO__PERMISSION_TEMPLATE__DELETE</b> <b>CAO__PERMISSION_TEMPLATE__VIEW</b>	Create, read, update, delete PKI permission templates
<b>CAO__REGISTRATION_SOURCE__CREATE</b> <b>CAO__REGISTRATION_SOURCE__UPDATE</b> <b>CAO__REGISTRATION_SOURCE__DELETE</b> <b>CAO__REGISTRATION_SOURCE__VIEW</b>	Create, read, update, delete Registration sources (LDAP, DB)
<b>CAO__SNOW__CREATE</b> <b>CAO__SNOW__UPDATE</b> <b>CAO__SNOW__DELETE</b> <b>CAO__SNOW__VIEW</b>	Create, read, update, delete SNOW operations
<b>CAO__CLIENT_SCEP_PIN__VIEW</b> <b>CAO__CLIENT_SCEP_PIN__UPDATE</b>	Read, update Client SCEP PINs
<b>CAO__CLIENT_ACME_TOKEN__VIEW</b>	Read Client AVCME Tokens (automatically generated by ACME protocol)
<b>CAO__JOB__CREATE</b> <b>CAO__JOB__UPDATE</b> <b>CAO__JOB__DELETE</b> <b>CAO__JOB__VIEW</b>	Create, read, update, delete asynchronous Jobs
<b>CAO__SCION_TRC__VIEW</b> <b>CAO__SCION_TRC__UPDATE</b>	Read or Update (Issue) SCION TRC update

<b>CAO__EVENTS__VIEW</b>	CA Operator can query/view Realm's events
<b>CAO__AIR_GAPED__CREATE</b> <b>CAO__AIR_GAPED__UPDATE</b> <b>CAO__AIR_GAPED__DELETE</b> <b>CAO__AIR_GAPED__VIEW</b>	Create, read, update, delete Air Gaped CA elements
<b>CAO__SWS_DOMAIN_PREVALIDATION__CREATE</b> <b>CAO__SWS_DOMAIN_PREVALIDATION__UPDATE</b> <b>CAO__SWS_DOMAIN_PREVALIDATION__DELETE</b> <b>CAO__SWS_DOMAIN_PREVALIDATION__VIEW</b>	Create, read, update, delete SwissSign pre-validated domains

### 7.4.2.3 Permissions associated with the Auditor Role

No permission associated with the Auditor role. When a user has the Auditor role, then he/she can query Realm's events.

### 7.4.2.4 Permissions associated with the Registration Officer Role

Permission Groups	Description
RAO__CERTIFICATE__ISSUE	RA officer can issue certificates
RAO__CERTIFICATE__UPDATE	RA officer can update certificate information
RAO__CERTIFICATE__VIEW	RA officer can query/view certificates
RAO__CERTIFICATE__PUBLISH	RA officer can publish certificates
RAO__CERTIFICATE__REVOKE	RA officer can revoke certificates
RAO__CRL__VIEW	RA officer can query/view CRLs/ARLs
RAO__SCEP_PIN__VIEW RAO__SCEP_PIN__UPDATE	RA officer can view/update SCEP PINs
RAO__ACME_TOKEN__VIEW	RA officer can query/view ACME Tokens

### 7.4.2.5 Permissions associated with the Authorizer Role

Permission Groups	Description
AUTHORIZER__CERT__ISSUANCE__VIEW AUTHORIZER__CERT__ISSUANCE__ALLOW	View, allow certificate issuance
AUTHORIZER__CERT__REVOCATION__VIEW AUTHORIZER__CERT__REVOCATION__ALLOW	View, allow certificate revocation
AUTHORIZER__CERT__RENEWAL__VIEW AUTHORIZER__CERT__RENEWAL__ALLOW	View, allow certificate renewal
AUTHORIZER__CERT__RECOVERY__VIEW AUTHORIZER__CERT__RECOVERY__ALLOW	View, allow recovery

### 7.4.2.6 Permissions associated with User Accounts

Permission Groups	Description
USER__ACCOUNT__VIEW USER__ACCOUNT__UPDATE	User can view his/her account and update his/her user information
USER__ACCOUNT__TOTP__VIEW USER__ACCOUNT__TOTP__UPDATE	User can view his/her TOTP and update his/her TOTP information

<b>USER__ACCOUNT_PERMISSIONS__VIEW</b>	User can view his/her permissions
<b>USER__ACCOUNT_API_KEY__VIEW</b> <b>USER__ACCOUNT_API_KEY__UPDATE</b>	User can view his/her API Key and update his/her API Key

## 8 Working with SwissPKI

### 8.1 SwissPKI Architectural Model

The following section describes the architectural model of SwissPKI. It helps people who are not yet familiar with SwissPKI, to understand the model, the relationship between its elements and their dependencies.

#### 8.1.1 Deployment

A deployment refers to a single logical installation of a SwissPKI environment. This can be on a single server, a group of distributed servers or a cluster of virtual server or container instances.

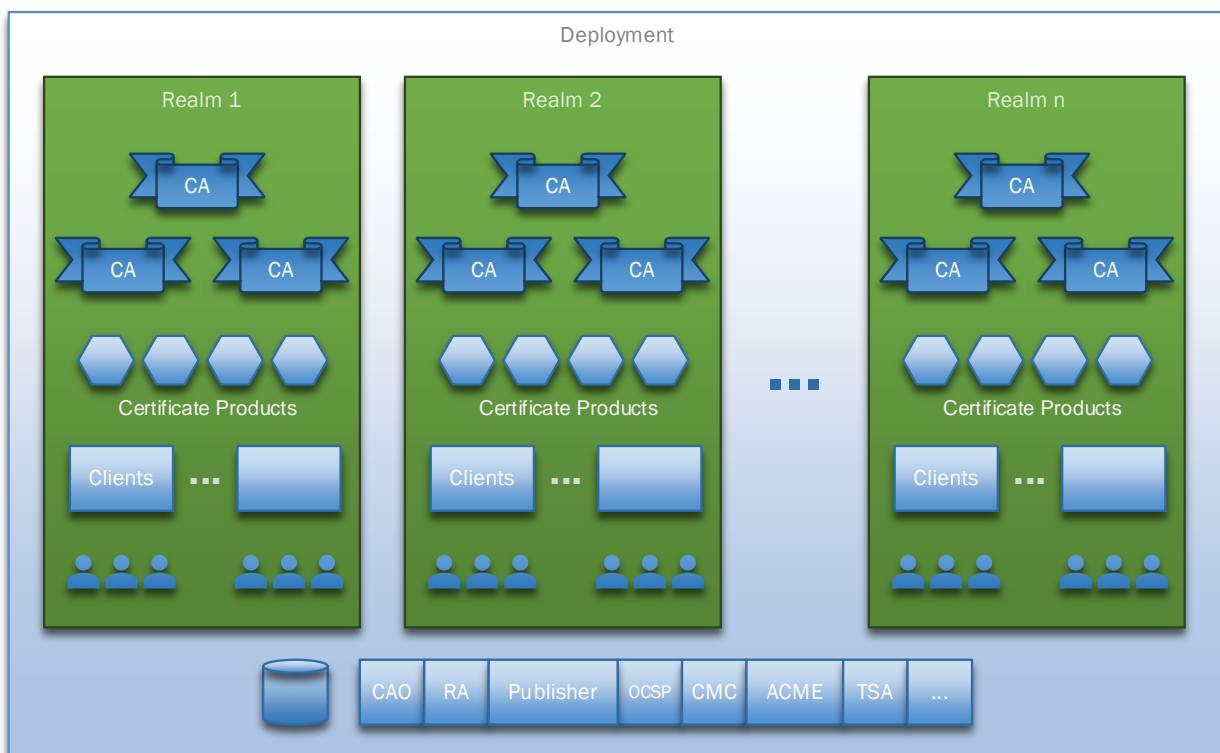
A deployment is a set of physical installations of application modules. It differs from another deployment in that no logical components are shared. Each deployment has its own database instance, its own application instances and thus its own access URLs. Of course, several deployments can be operated on the same server environment. However, as they each have a separate application installation base and thus possibly also different TCP ports.

A deployment contains one or more realms.

### 8.1.2 Realm

A Realm is a tenant and SwissPKI supports multiple Realms (multi-tenant) per deployment. PKIs along with the Certification Authorities, certificates, users, and clients are deployed within Realms. PKIs deployed within a Realm cannot cross their Realm boundary except if you decide to cross-sign Certification Authorities between Realms. Additionally, users created within one Realm cannot access PKI entities deployed in another Realm. You need to create separate users in each Realm if you plan to have one ‘physical’ person accessing different PKIs deployed in different Realms.

To conclude, a Realm is a set of Certification Authorities (linked or unlinked to each other) and defines a set of available certificate policy templates. It also has a set of clients and users that have access to these CA instances according to the permissions granted.



### 8.1.3 Clients

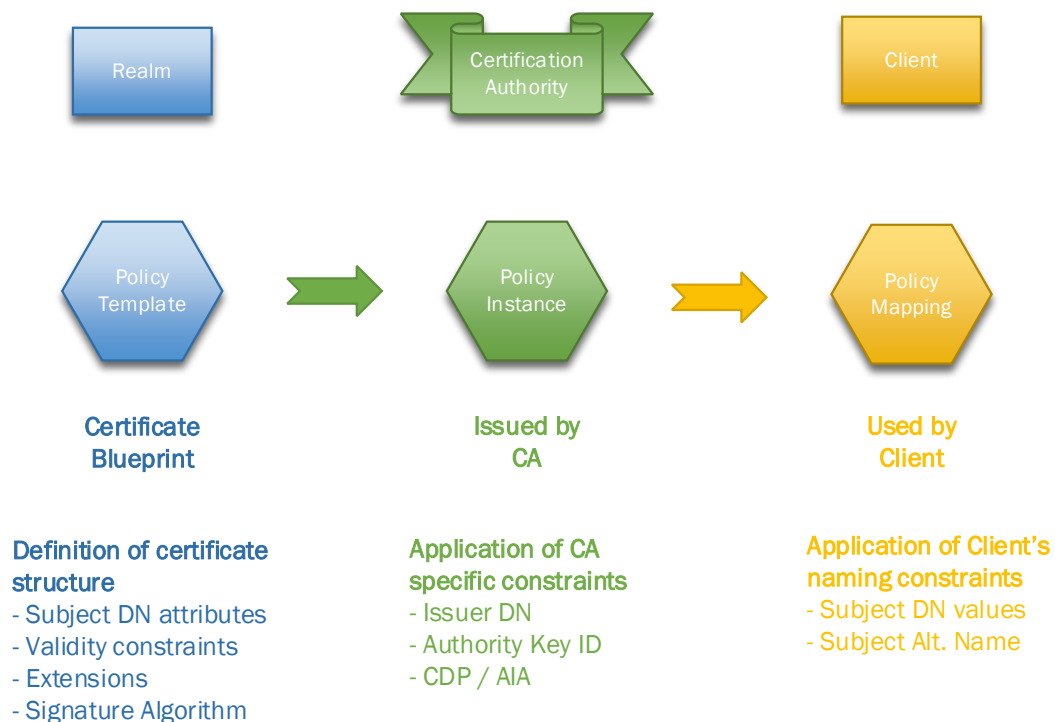
A realm can have one or multiple clients. These Clients can be seen as groups of users that have access to all or only a subset of certain certificate products issued by the different CAs of that realm.

Clients are usually used to group the users of a client’s organization. Clients can be equipped with specific validation rules and naming constraints that are to be applied to the issued certificates.

### 8.1.4 Certificate Products

Certificate product definitions are managed on three distinct levels:

- **Policy Template:** Is the basic certificate blueprint (ASN.1 and encoding) that defines the structure and attributes of a certificate product. It defines whether attributes and extensions are mandatory, editable, or optional, or have predefined values in all issued certificates of that type. A Policy Template exists in the Realm and can be used by any CA of that Realm.
- **Policy Instance:** Is the binding of a Policy Template to a certain Certificate Authority. A Policy Instance extends a Policy Template by defining the Issuer detail values like the Issuer DN, the Authority Key ID and the CA's specific CRL Distribution Point and/or Authority Information Access certificate extensions.
- **Policy Mapping:** Is the assignment of a Policy Instance to a specific client. On this level, the values that will be filled into the certificate attributes can be further restricted using rules. For example, a client organization may only include domain names for which it has provided prior proof of ownership. Or only names of identities that were pre-registered in a database or LDAP server. Each Policy Mapping is specific to one client and restricts the values according to its given rules.



- **Policy Mapping Rules:** define the runtime behavior of the assigned certificate product to a client. Depending on the Policy Type, one can enable/disable or configure the following runtime rules (in addition to the specific DNS Owner Check, CT Log and CAA checks):



- Certificate issuance email notification (multilanguage, multiple recipients, global or client specific)
- Certificate revocation email notification (multilanguage, multiple recipients, global or client specific)
- Certificate authorization rules with multilanguage, multiple recipients, global or client specific notification on a combination of certificate issuance, renewal, or revocation
- Certificate renewal rules with multilanguage, multiple recipients, global or client specific notification for manual or automatic certificate renewal with increment notifications and last reminder if a certificate is going to expire within n days
- Publication of the certificate in a destination repository such as LDAP, SSH or file system
- Registration sources to check against one or multiple LDAP/DB sources if a user/system to issue a certificate for is present
- Registration documents to be uploaded along with the certificate when issued or once it has been issued
- Certificate attribute validation rules (Subject DN, SAN extensions)
- Validity (expiration) date of the certificate product

This layering of the certificate product definition allows for optimal re-use of elements and to control its content based on the assignment structures. Only CAs that have an instantiation of a Policy Template may issue certificates based on that template. And only clients that have a mapping to a CA's Policy Instance may issue this certificate product. And then also only in compliance with the restrictions that apply to it.

### 8.1.5 Users

Users are bound to a single Realm. But they can be authorized to access multiple Clients.

Authentication mechanisms for users include:

- Local username & password with TOTP
- OpenID Connect
- LDAP
- Kerberos for SSO

It is important to understand, that as Users are bound to a Realm, People must use different User IDs for other realms in the same deployment. If the Realms reside in different deployments, the same User IDs can be created in each Realm and then be linked to the same account of the external authentication scheme.

Multiple authentication methods can be activated in parallel:



Detailed authentication configuration options are described in section **8.3 End User Login Options**

### 8.1.6 Initializing SwissPKI

Before issuing any certificate, you need to setup the PKI environment and configure the general settings of the PKI:

1. Initialize SwissPKI. This step is performed by a PKI Administrator role.
2. As a PKI Administrator, you setup Realms and associated CA Operators and configure the environment's general settings.
3. Only after steps 1 and 2 are done, the CA Operator can log into the PKI Realm and start setting up the actual Certification Authorities and associated components.
4. Once the CA Operator has configured the PKI, created, and associated the first Registration Officers to their respective issuing Clients, only then the first certificates can be issued.

**Note:** all UI operation can be executed with the OpenAPI interface

As a PKI Administrator, initialize SwissPKI using the PKI Administration Web UI. The PKI Administrator will be asked to provide configuration information for the system SMTP settings and initialize it is PKI Administrator user account.

Please refer to section 9 *Initializing SwissPKI*

### 8.1.7 Configuring Realms

Log in to SwissPKI using the SwissPKI Administrator Web UI. Only PKI Administrators can log into the Administration application. As a first PKI Administrator user, you will have all privileges assigned to start configuring the Realms and general settings.

What do I configure as PKI Administrator?

1. Manage PKI Administrator and CA Operator permission templates
2. Create and manage other PKI Administrators and assign permissions to those users. Note that you cannot assign yourself other permissions. Only another PKI Administrator with permission management privileges can modify yours.
3. Configure the Primus CloudHSM proxy information (if enabled)
4. Configure the system's SMTP connection information
5. Create and manage Realms
  - a. Create and manage CA Operators and assign them privileges based on the permission templates edited or created in step 1
  - b. Manage Realm SMTP connections
  - c. Manage Realm DNS
  - d. Manage Realm CNG
  - e. Manage Realm Trust Anchors

- f. Manage SCION Identity Repository Validation
- g. Manage Certificate Linters
- h. Manage CT Log families

Please refer to section *11 Administrator UI* for detailed instructions.

### 8.1.8 Setting up the PKI

As a CA Operator, you will initially need to setup the PKI for the assigned Realm. Connect to the Operator UI, from the 'Manage' menu:

1. Create and manage users. This is the list of all users associated to the PKI Realm. Active users with assigned roles can log onto either the Operator UI or the Registration UI.
2. Assign Auditor roles to users. Users with Auditor role can log onto the Operator UI and access event information within the PKI Realm.
3. Create and configure Clients (name, description, parent, external and partner references).
  - a. Assign users as RA Officers. A user with RA Officer role assigned to a Client (one or more) can manage, if privileged, certificates for the associated clients.
  - b. Assign users as Authorizers. A user with Authorizer role assigned to a Client (one or more) can manage, if privileged, authorization requests associated with the clients' and issued by RA Officers.
  - c. Register pre and/or post validation rules. You can register pre/post validation micro-services to a Client. Those are HTTPS URLs which will get pre/post invoked when certificates are issued for this client. Context information is sent with each request.
  - d. Register CMP signer certificates. To enable CMP client protocol for a Client, you register signing certificates for the CMP server to validate incoming requests.
  - e. Browse issued ACME tokens
  - f. Browse and manage issued SCEP PINs
  - g. Browse assigned certificate policies (certificate products).
  - h. Create and manage Client DNS. Override Realm DNS settings for the specific Client (DNS and DNSSEC). DNS information is used in conjunction with DNS Owner Check validation.
  - i. Create and manage technical contacts
4. Create and manage Rules. This is a catalog of rules which can be associated to certificate polices and invoked when specific events during certificate life cycle occur.
  - a. Create and manage global Realm validation rules. Global Realm validation rules are regular expressions or interface implementations which are invoked when validating certificate content. The rules are associated to certificate templates and to perform

- ‘simple’ content validation. For refined content validation, use either client pre/post validation services or policy instance rules.
- b. Create and manage registration rules. Registration rules are applied during certificate registration to include registration documents when issuing certificates. This type of rule is typically used for qualified certificates necessitating strong authentication of the certificate’s recipient. Registration rules are assigned to certificate policies (certificate products) associated to a specific client.
  - c. Create and manage authorization rules. Authorization rules are applied during certificate issuance, renewal, and revocation. The authorization rules are assigned to certificate policies (certificate products) associated to a specific client.
  - d. Create and manage renewal rules. Renewal rules are applied to certificates which require renewal. The renewal rules are assigned to certificate policies (certificate products) associated to a specific client.
  - e. Create and manage CAA rules. These rules are associated to certificate templates and applied when issuing certificates.
  - f. Create and manage DNS Owner Check rules. These rules are associated to certificate templates and applied when issuing certificates. Various DNS Owner checks can be configured.
  - g. Create and manage CT Log rules. These rules are associated to certificate templates and applied when issuing certificates (pre certificate or OCSP stapling).
5. Create and manage Notifications. Notifications are sent during the certificate life cycle when specific events occur (e.g., issuance, revocation, renewal, authorizations ...)
    - a. Create and manage attachments. Attachments are PDF documents which can be attached to notifications.
    - b. Create and manage notification templates. Notification templates are the actual message (configurable) which is sent to the recipients.
  6. Create and manage Registration Sources. Registration sources are data sources (LDAP, DB) which can be assigned to certificate policies (certificate products) associated to a specific client for issuing certificates for recipients which are found in one or multiple sources.
  7. Create and manage HSM partition (if enabled). Register HSM partitions and hosts which can be associated to certificate templates (keys will be generated on the HSM partitions).
  8. Create and manage Permission Templates. Those are the permissions templates which the CA Operator assigns to the Auditor, Authorizer and RA Officer roles.
  9. Browse Events. Search events which occurred within the PKI Realm.

Once the Realm settings are defined, the CA Operators can start configuring the PKI. From the ‘PKI’ menu in the Operator UI:

- a. Create and manage certificate templates. A Certificate templates is the definition of the structural layout of the content of an issued certificate. There are several types of certificate templates: Certificate Authority, OCSP, TSA, CMP, Document Signer Server, Microsoft, SwissSign (Public Trust), External and General
2. Create and manage PKI Entities.
  - a. Create and manage Certification Authorities
    - i. SwissPKI CAs
      1. Query issued certificates
      2. Manage policy instances. Policy instances are certificate templates associated with the Certification Authority and define which type of certificates are issued by which client using a specific rule set.
      3. Manage CRL Distribution Points. CDPs are defined to be included in issued certificates.
      4. Manage Authority Information Access. AIA end points are convenience URLs to avoid copying Certification Authority certificates to different web servers and will return the Certification Authority certificate when invoked.
      5. Manage certificate linters. Register one or multiple certificate linter if you plan to provide pre/post TBS certificate and certificate content inspection. You register the linter service using HTTPS URLs to service interfaces <sup>4</sup>.
      6. Manage CRL publication rules and browse issued CRL/ARL. CRL publication rules will produce CRL/ARL based on a defined schedule with a given CRL/ARL validity. CRL/ARL validity is defined in the Certification Authority settings.
      7. Configure Certification Authority settings. Configure general Certification Authority settings such as CRL/ARL grace period, unique public key checks and extended CRL settings.
      8. Issue cross-signed requests and import cross signed certificates. SwissPKI lets you cross sign a Certification Authority instance and branch the certificate chain to the issued cross signed certificate.
      9. Manage Microsoft CEP/CES connection points. When integrating with Microsoft, define the registration URL for Microsoft AD.
    - ii. SwissSign CAs

---

<sup>4</sup> Contact [support@swisspki.com](mailto:support@swisspki.com) if you need such linter services

1. Associated Public Trusted certificate policies to Clients to allow them to issue SwissSign certificates (this requires a connection to the SwissSign service and requires an additional service agreement)
- iii. External CAs
  1. Import certificates from external Certification Authorities to manage their life cycle.
- b. Create and manage OCSP
  - i. Associated Certification Authorities to OCSP
  - ii. Activate Certification Authorities
- c. Create and manage TSA. Offers time stamping functionality. TSA are issued by a SwissPKI Certification Authority.
- d. Create and manage DSS. Document Signer Server are issued by a SwissPKI Certification Authority.
- e. Create and manage CMP. CMP Server are issued by a SwissPKI Certification Authority.
- f. Create and manage CES/CEP. Create a Microsoft integration service for autoenrollment.
- g. Create and manage Publisher. Associate CRL/ARL and certificate publishing with one or more Certification Authorities.

A Dashboard is available from the 'Dashboard' menu:

1. Query asynchronous Job execution.
2. Query expiring certificates over the Realm.
3. Display HSM partition status.
4. Additional dashboard functionalities available upon request.

Please refer to section *12 Operator UI* for detailed instructions.

### 8.1.9 Issuing Certificates

Once the PKI Realm configured by a CA Operators, you can start issuing certificates with an RA Officer role or Service using:

1. The Registration UI (Web interface)
2. ACME Protocol
3. SCEP Protocol
4. Microsoft Autoenrollment
5. CMP Protocol <sup>5</sup> (limited to issue and revoke)
6. OpenAPI v3 REST API

Using the Registration UI or OpenAPI v3 REST API, you can (if privileged):

1. Query issued certificates assigned to your role
2. Revoke issued certificates for products assigned to your role
3. Edit some certificate attributes and settings
4. Search for issued CRL/ARL
5. Access the Dashboard <sup>6</sup> for an overview of the various certificate order status and expiring certificates.
6. If you have an Authorizer role, you can authorize pending issuance, revocation, and renewal requests.

---

<sup>5</sup> Contact [support@swisspki.com](mailto:support@swisspki.com) to obtain a CMP Client Java library

<sup>6</sup> Contact [support@swisspki.com](mailto:support@swisspki.com) if you wish extended Dashboard functionalities



## **Auditor UI**

Auditors are PKI users who have been assigned the AUDITOR role. They are authorized to access the audit log.

Please refer to section *12.2.9 Events*

## **8.2 End User Login Options**

Several login options are available to authenticate the PKI users on the Administration, Operator and Registration Web UIs <sup>7</sup>:

1. Username/Password with TOTP
2. LDAP Server
3. OpenID Connect
4. Kerberos

A user can login using any of the activated authentication mechanism. If Kerberos is activated, it will be used first and therefore provide the SSO functionality for SwissPKI.

When deploying each Web UI application separately, you have the possibility to configure different authentication mechanisms. For example:

- Deploy the Administration UI with Username/Password and TOTP and LDAP Server
- Deploy the Operator UI with Kerberos and LDAP Server
- Deploy the Registration UI with OpenID connect

**Note:** If you deploy modules in one application, then the authentication mechanisms apply for the deployed application. For example, deploying Administration and Operator UIs in one application would both use the same authentication configurations.

**Note:** SwissPKI usernames are unique across the application. Users' setup in separate Realms cannot have an identical username

---

<sup>7</sup> For authentication configuration settings, please refer to 'SwissPKI Deployment' documentation

### 8.2.1 Onboarded vs Validated users

If you decide to enable external authentication methods (LDAP, OpenID Connect or Kerberos), SwissPKI lets you configure the authentication mechanism for the deployed WebUI application such that authenticated users can be ‘onboarded’ vs ‘validated.’

‘**Onboarded**’ users are created in the SwissPKI database upon first successful login. Using this mechanism, you do not need to create users in SwissPKI upfront.

‘**Validated**’ users are not created in the SwissPKI database upon successful login and must be created first by a PKI Administrator or CA Operator.

When one of LDAP, OpenID Connect or Kerberos authentication mechanism is enabled and ‘onboarding’ is disabled, the PKI Administrator or CA Operator have the option to create ‘validated’ users. A validated user is a user account which is created with a ‘validated’ and ‘ACTIVE’ account.

### 8.2.2 Username/Password with TOTP Login


Username/password with TOTP is the default authentication mechanisms used by SwissPKI. The requirements for username/password with TOTP are:

1. Valid email address (verified via a confirmation link)
2. A QR Code application for scanning the second factor. The second factor is used in conjunction with the password to log into the SwissPKI account.

When an Administrator creates users, a registration link is sent via email for validation. Upon successful validation by the newly registered user, a password reset link is sent out to the user to set it. Additionally, TOTP QR codes and scratch codes<sup>8</sup> are also sent per email to the user. This email message contains the links to QR Code applications for iPhone and Android in case the user has no such application installed. Supported QR Code applications are Google Authenticator and FreeOTP.

---

<sup>8</sup> 10 scratch codes are sent to the user upon registration or TOTP reset. The scratch codes are used to log into SwissPKI if the user has no access to his/her QR Code reader (e.g., mobile phone). Print the scratch codes and keep them in a secure place.

SwissPKI™ 



**TOTP Login**

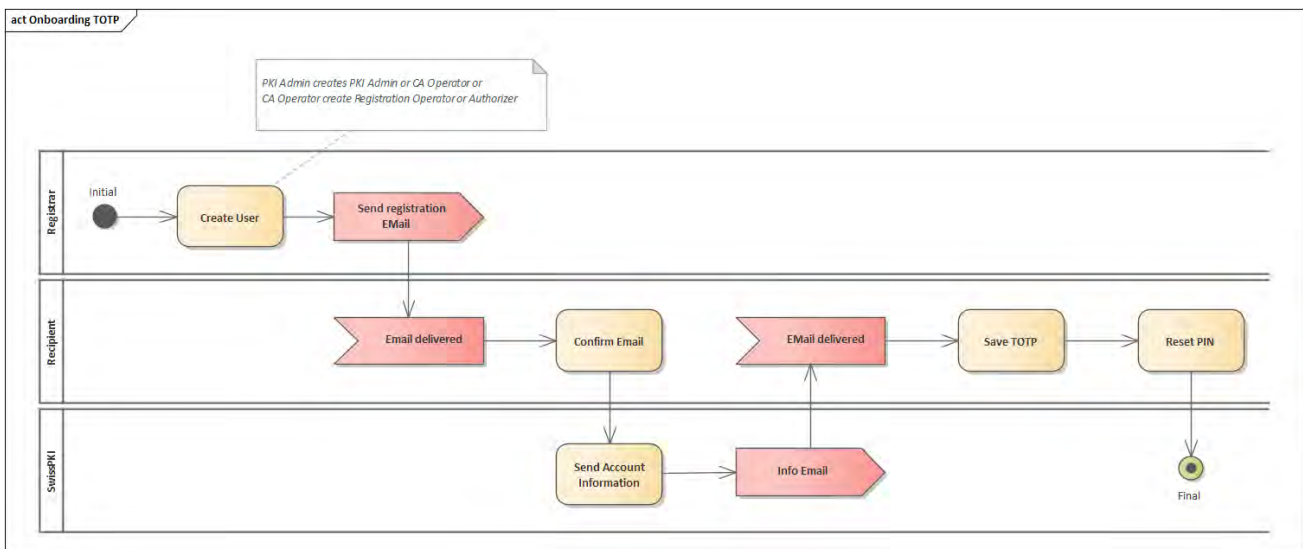
User name\*

Password\*

Second factor\*

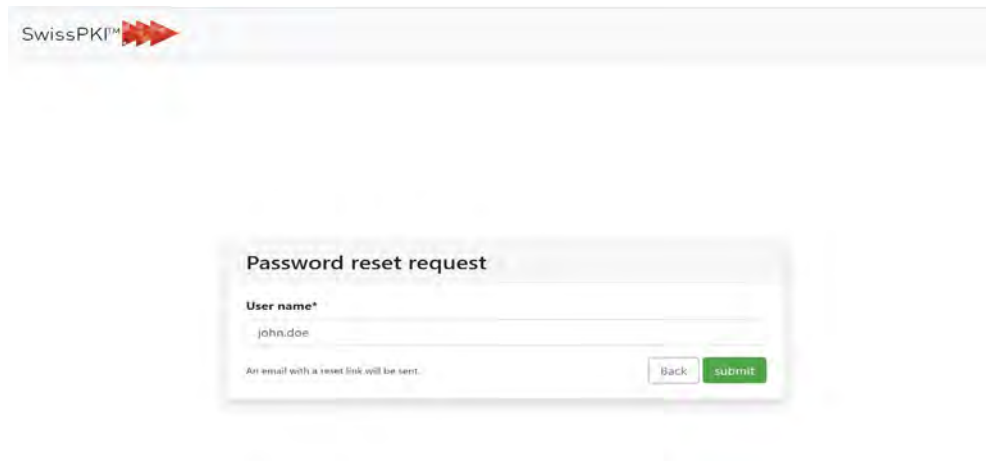
[Forgot your password ?](#)

### 8.2.2.1 Onboarding workflow



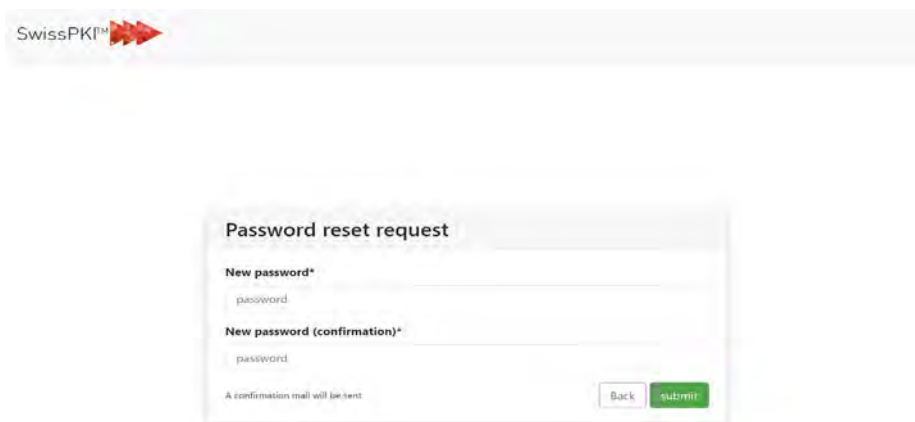
### 8.2.2.2 Password reset

Users can reset their passwords using the 'password reset' link on the TOTP login page:



The screenshot shows the top of a web page with the "SwissPKI™" logo and a navigation bar. Below the navigation bar is a form titled "Password reset request". The form contains a "User name\*" field with the text "john.doe" entered. Below the field is a small text note: "An email with a reset link will be sent." At the bottom right of the form are two buttons: "Back" and "submit".

Submitting the request sends password reset link to the user:

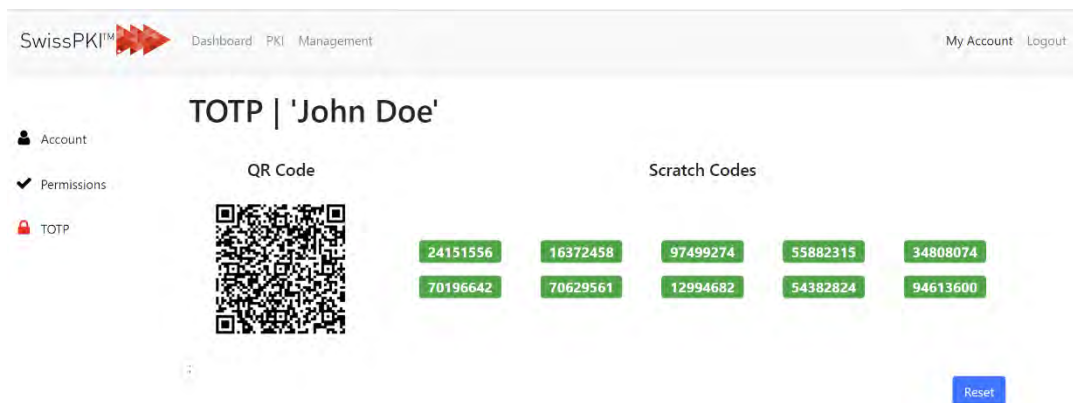


The screenshot shows the top of a web page with the "SwissPKI™" logo and a navigation bar. Below the navigation bar is a form titled "Password reset request". The form contains two fields: "New password\*" with the text "password" entered, and "New password (confirmation)\*" with the text "password" entered. Below the fields is a small text note: "A confirmation mail will be sent." At the bottom right of the form are two buttons: "Back" and "submit".

Upon successful password reset, a confirmation email is sent to the user.

### 8.2.2.3 TOTP reset

Users with the permission enabled to manage their TOTP can reset the token using the 'My Account' menu when logged into SwissPKI.



Clicking 'Reset' will generate a new TOTP and list of scratch codes. Used scratch codes are displayed in red. Resetting the TOTP sends an email to the user with the updated QR Code information.

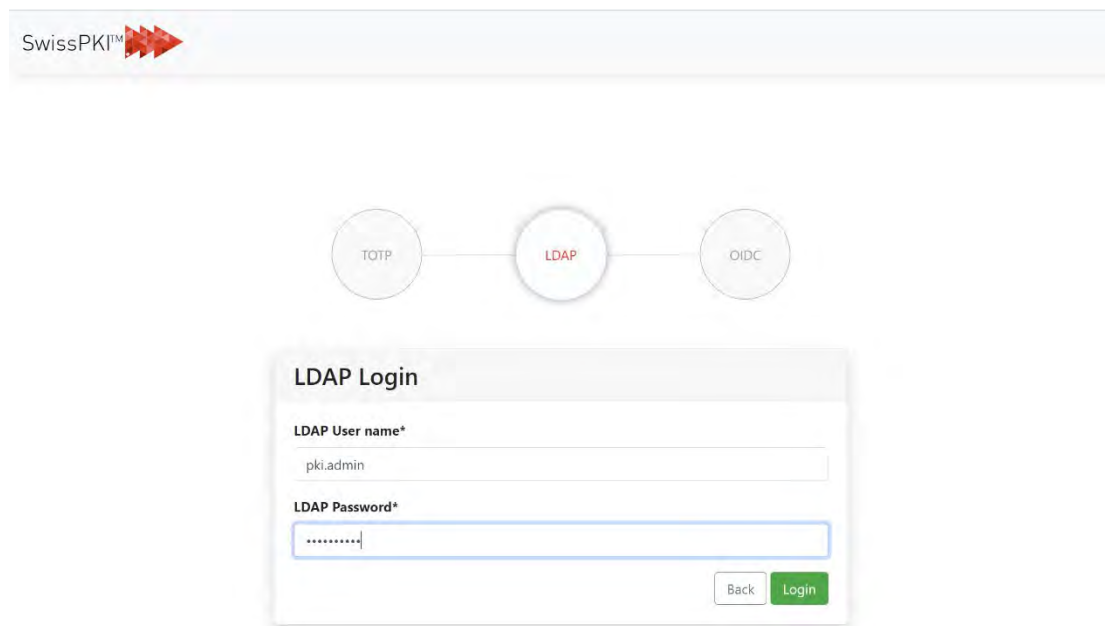
If the user has no permission to update the TOTP token or has no access to his/her account, a CA Operator can perform the TOTP reset on behalf of the user. If a CA Operator has no permission to reset his/her TOTP token or has no access to the his/her account, a PKI Administrator can reset the TOTP token on his/her behalf.

### 8.2.2.4 Password lock

End user password lock occurs after three consecutive password mismatches. The login is locked for 30 minutes. If additional failed login attempts occur during the 30-minute lock out period, then an additional 10 minutes is added for each password mismatch during this period.

### 8.2.3 LDAP Server

Users can log into SwissPKI using LDAP Server authentication. LDAP Server authentication enables you to manage all your SwissPKI Realms, Users, Roles, Permission Templates and Clients in an LDAP.



**Important:**

When LDAP onboarding configuration options is **enabled**: users logging in via LDAP will setup the user account and associated permissions and roles upon initial log on. If a user who logs into SwissPKI has an RA Operator and/or Authorizer Role associated with a Client, then the Client is created within the defined Realm if it does not exist. Realms must be defined in SwissPKI.

When LDAP onboarding configuration options is **disabled**: Users logging in via LDAP must initially be created as 'validated' users in SwissPKI.

### 8.2.3.1 LDAP Requirements

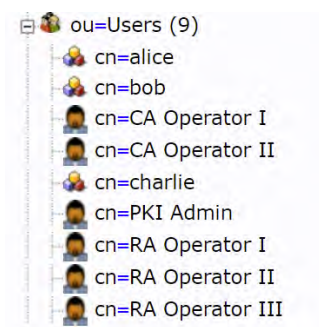
LPDAP v3 server protocol with:

Configuration	Description
<b>Host</b>	LDAP Server hostname or IP
<b>Port</b>	Accessible LDAP port 389 or 636 for SSL
<b>Bind DN</b>	A user with read access to the user BaseDN and SwissPKI BaseDN
<b>Bind Password</b>	A user password
<b>User BaseDN</b>	A distinguished name which identifies the base entry of the users in the DIT
<b>User login attribute</b>	The 'username' to search for in the User BaseDN. Usually set to 'uid'
<b>SwissPKI BaseDN</b>	A distinguished name which identifies the base entry of the SwissPKI groups and roles configuration in the DIT

#### 8.2.3.1.1 User structure

Any DN in the DIT used for searching and authenticating users using the 'user login attribute.' This applies to both 'onboarded' and 'validated' users.

**Example:** `ou=Users,dc=swisspki,dc=com`



### 8.2.3.1.2 Top Level LDAP structure

This section applies **only** when **'onboarding'** is **enabled** in the deployment configuration settings.

Define a top level SwissPKI configuration entry point in the LDAP of *'objectClass'* type *'organizationalUnit'*

**Example:** *ou=SwissPKI,dc=swisspki,dc=com*

The top level SwissPKI configuration DN contains:

1. One instance of *objectClass* of type *groupOfUniqueNames* which holds unique members (*uniqueMember*) for the PKI Administrator roles pointing to the users in the *'User BaseDN.'*
  - a. The *groupOfUniqueNames* **MUST** have the attribute *cn='PKI Administrators'* set as the RDN
  - b. The *groupOfUniqueNames* **SHOULD** have the attribute *'businessCategory=<string>'* set. The value *'<string>'* is the name of the PKI Administrator Permission Template. By default, this value is set to *'Default PKI Administrators'* when initializing SwissPKI. You can create new Permission Templates of type *'PKI Administrator'* in SwissPKI via the Administrator UI and record them as default Permission Template in the *'businessCategory'* attribute of the *'groupOfUniqueNames'* LDAP entry.  
If the Permission Template is not located in SwissPKI, then the initial onboarding of the user will fall back to the default Permission Template *'Default PKI Administrators.'*  
If no Permission Template is found, then the initial onboarding during the LDAP login process for the authenticated user will fail.
2. One or multiple instances of *'objectClass'* of type *'organizationalUnit'* which represent the SwissPKI Realms.
  - a. The Realm **MUST** exist in SwissPKI.
  - b. The LDAP Realm name **MUST** match the attribute *'ou=<string>'* set as RDN of *'organizationalUnit.'*  
**Note:** SwissPKI Realm names are not unique. Make sure that you create unique Realm names in SwissPKI for the LDAP login process to function correctly.



### 8.2.3.1.2.1 Realm Structure

LDAP Realms regroup the users with Auditor and/or CA Operator roles. Additionally, they contain the Clients within the Realm.

For each Realm, you define:

1. One instance of *objectClass* of type *groupOfUniqueNames* which holds unique members (*uniqueMember*) for the CA Operator roles pointing to the users in the 'User BaseDN.'
  - a. The *groupOfUniqueNames* **MUST** have the attribute *cn='CA Operators'* set as the RDN
  - b. The *groupOfUniqueNames* **SHOULD** have the attribute *'businessCategory=<string>'* set. The value '*<string>*' is the name of the CA Operator Permission Template. By default, this value is set to 'Default CA Operators' when initializing SwissPKI. You can create new Permission Templates of type 'CAO' in SwissPKI via the Administrator UI and record them as default Permission Template in the '*businessCategory*' attribute of the '*groupOfUniqueNames*' LDAP entry.

If the Permission Template is not located in Realm, then the initial onboarding of the user will fall back to the default Permission Template 'Default CA Operators.' If no Permission Template is found, then the initial onboarding during the LDAP login process for the authenticated user will fail.

2. One instance of *objectClass* of type *groupOfUniqueNames* which holds unique members (*uniqueMember*) for the Auditor roles pointing to the users in the 'User BaseDN.'
  - a. The *groupOfUniqueNames* **MUST** have the attribute *cn='Auditors'* set as the RDN
  - b. The *groupOfUniqueNames* **SHOULD** have the attribute *'businessCategory=<string>'* set. The value '*<string>*' is the name of the Auditor Permission Template. By default, this value is set to 'Default Auditors' when initializing SwissPKI. You can create new Permission Templates of type 'Auditor' in SwissPKI via the Operator UI and record them as default Permission Template in the '*businessCategory*' attribute of the '*groupOfUniqueNames*' LDAP entry.

If the Permission Template is not located in Realm, then the initial onboarding of the user will fall back to the default Permission Template 'Default Auditors.' If no Permission Template is found, then the initial onboarding during the LDAP login process for the authenticated user will fail.

3. One or multiple instances of *'objectClass'* of type *'organizationalUnit'* which represent the Realm's Client.
  - a. The Client **CAN** exist in SwissPKI. If not present, then the initial onboarding of a RA Operator and/or Authorizer user will automatically create the Client in the Realm.
  - b. The LDAP Client name **MUST** match the attribute *'ou=<string>'* set as RDN of *'organizationalUnit.'*

**Note:** SwissPKI Client names are not unique. Make sure that you create unique Client names in SwissPKI for the LDAP login process to function correctly.

### 8.2.3.1.2.2 Client structure

LDAP Client structures regroup the users with Authorizer and/or RA Operator roles.

For each Client, you define:

1. One instance of *objectClass* of type *groupOfUniqueNames* which holds unique members (*uniqueMember*) for the RA Operator roles pointing to the users in the *'User BaseDN.'*
  - a. The *groupOfUniqueNames* **MUST** have the attribute *cn='RA Operators'* set as the RDN
  - b. The *groupOfUniqueNames* **SHOULD** have the attribute *'businessCategory=<string>'* set. The value *'<string>'* is the name of the RA Operator Permission Template. By default, this value is set to *'Default RA Operators'* when initializing SwissPKI Realm. You can create new Permission Templates of type *'RAO'* in SwissPKI via the Operator UI and record them as default Permission Template in the *'businessCategory'* attribute of the *'groupOfUniqueNames'* LDAP entry.
 

If the Permission Template is not located in Realm, then the initial onboarding of the user will fall back to the default Permission Template *'Default RA Operators.'* If no Permission Template is found, then the initial onboarding during the LDAP login process for the authenticated user will fail.
2. One instance of *objectClass* of type *groupOfUniqueNames* which holds unique members (*uniqueMember*) for the Authorizer roles pointing to the users in the *'User BaseDN.'*
  - a. The *groupOfUniqueNames* **MUST** have the attribute *cn='Authorizers'* set as the RDN
  - b. The *groupOfUniqueNames* **SHOULD** have the attribute *'businessCategory=<string>'* set. The value *'<string>'* is the name of the Authorizer Permission Template. By default, this value is set to *'Default Authorizers'* when initializing SwissPKI Realm. You can create new Permission Templates of type *'Authorizer'* in SwissPKI via the Operator UI and record them as default Permission Template in the *'businessCategory'* attribute of the *'groupOfUniqueNames'* LDAP entry.
 

If the Permission Template is not located in Realm, then the initial onboarding of the user will fall back to the default Permission Template *'Default Authorizers.'* If no

Permission Template is found, then the initial onboarding during the LDAP login process for the authenticated user will fail.

### 8.2.3.1.2.3 Role activation/deactivation

Effects on User objects in SwissPKI database when logging in via LDAP

Role	Action	Effect
<b>PKI Administrator</b>	Add/Update to LDAP group	<p><i>First login:</i></p> <p>User is created in DB with username name set to defined LDAP user attribute (usually <i>uid</i> LDAP attribute)</p> <p>If user with same username is already present (other login mechanisms like TOTP or OIDC), then the creation fails</p> <p>Permission Template assigned as defined in LDAP group</p> <p><i>Subsequent login:</i></p> <p>User is updated with LDAP values.</p>
	Remove from LDAP group	<p>Login denied</p> <p>User not deactivated in DB. Login using other mechanisms (TOTP, OIDC, Kerberos) still possible. Manually disable user in SwissPKI to deny access through other authentication mechanisms.</p>
	Manually deactivated via UI	User 'reactivated' upon successful login
<b>CA Operator</b>	Add/Update to LDAP group	<p><i>First login:</i></p> <p>User is created in DB with username name set to defined LDAP user attribute (usually <i>uid</i> LDAP attribute)</p>

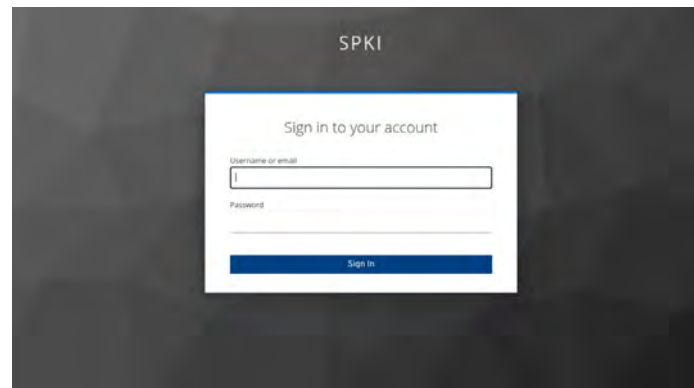
		<p>If user with same username is already present (other login mechanisms like TOTP or OIDC), then the creation fails</p> <p>Permission Template assigned as defined in LDAP group</p> <p>Role is set</p> <p><i>Subsequent login:</i></p> <p>User is updated with LDAP values.</p> <p>Role is updated</p>
	Remove from LDAP group	<p>Login denied</p> <p>User not deactivated in DB. Login using other mechanisms (TOTP, OIDC, Kerberos) still possible. Manually disable user in SwissPKI to deny access through other authentication mechanisms.</p>
	Manually deactivated via UI	User 'reactivated' upon successful login
<b>Auditor</b>	Add/Update to LDAP group	<p><i>First login:</i></p> <p>User is created in DB with username name set to defined LDAP user attribute (usually <i>uid</i> LDAP attribute)</p> <p>If user with same username is already present (other login mechanisms like TOTP or OIDC), then the creation fails</p> <p>Permission Template assigned as defined in LDAP group</p> <p>Role is set</p> <p><i>Subsequent login:</i></p> <p>User is updated with LDAP values.</p>

		Role is updated
	Remove from LDAP group	<p>Login denied</p> <p>User not deactivated in DB. Login using other mechanisms (TOTP, OIDC, Kerberos) still possible. Manually disable user in SwissPKI to deny access through other authentication mechanisms.</p>
	Manually deactivated via UI	User 'reactivated' upon successful login
<b>RA Operator</b>	Add/Update to LDAP group	<p><i>First login:</i></p> <p>User is created in DB with username name set to defined LDAP user attribute (usually <i>uid</i> LDAP attribute)</p> <p>If user with same username is already present (other login mechanisms like TOTP or OIDC), then the creation fails</p> <p>Permission Template assigned as defined in LDAP group</p> <p>Client is created if not present</p> <p>Role is set per Client</p> <p><i>Subsequent login:</i></p> <p>User is updated with LDAP values.</p> <p>Role is updated per Client</p>
	Remove from LDAP group	<p>Login denied</p> <p>User not deactivated in DB. Login using other mechanisms (TOTP, OIDC, Kerberos) still possible. Manually disable user in SwissPKI to deny access through other authentication mechanisms.</p>

	Manually deactivated via UI	User 'reactivated' upon successful login
<b>Authorizer</b>	Add/Update to LDAP group	<p><i>First login:</i></p> <p>User is created in DB with username name set to defined LDAP user attribute (usually <i>uid</i> LDAP attribute)</p> <p>If user with same username is already present (other login mechanisms like TOTP or OIDC), then the creation fails</p> <p>Permission Template assigned as defined in LDAP group</p> <p>Client is created if not present</p> <p>Role is set per Client</p> <p><i>Subsequent login:</i></p> <p>User is updated with LDAP values.</p> <p>Role is updated per Client</p>
	Remove from LDAP group	<p>Login denied</p> <p>User not deactivated in DB. Login using other mechanisms (TOTP, OIDC, Kerberos) still possible. Manually disable user in SwissPKI to deny access through other authentication mechanisms.</p>
	Manually deactivated via UI	User 'reactivated' upon successful login

## 8.2.4 OpenID Connect

Users can log into SwissPKI using OpenID Connect authentication. OpenID Connect authentication enables you to manage all your SwissPKI Users, Roles, Permission Templates and Clients in using extend OpenID user information claims.



### **Important**

When OIDC onboarding configuration options is **enabled**: Users logging in via OpenID Connect will setup the user account and associated permissions and roles upon initial log on. If a user who logs into SwissPKI has an RA Operator and/or Authorizer Role associated with a Client, then the Client is created within the defined Realm if it does not exist. Realms must be defined in SwissPKI.

When OIDC onboarding configuration options is **disabled**: Users logging in via OIDC must initially be created as 'validated' users in SwissPKI. The SwissPKI username is the OIDC UserInfo email address.

### 8.2.4.1 OpenID Connect Requirements

OpenID Connect Provider:

Configuration	Description
<b>Provider</b>	Logical name to display on login screen title
<b>Well Known Config</b>	Fully qualified HTTP url pointing to <code>/.well-known/openid-configuration'</code>
<b>ClientID</b>	The client id for the RP
<b>Client Secret</b>	Client secret for the RP (HMAC256 signed requests)

#### 8.2.4.1.1 User Info attributes

The following User Info claims attributes and properties are managed by SwissPKI during login processing

Attribute/Property	Required
<b>sub</b>	MANDATORY
<b>preferred_username</b>	MANDATORY (set as username in the DB upon first login)
<b>given_name</b>	MANDATORY
<b>family_name</b>	MANDATORY
<b>email</b>	MANDATORY
<b>gender</b>	OPTIONAL (set to MR if claim not provided)
<b>locale</b>	OPTIONAL (set to EN if claim not provided)
<b>name</b>	OPTIONAL
<b>middle_name</b>	OPTIONAL
<b>nickname</b>	OPTIONAL
<b>profile</b>	OPTIONAL
<b>picture</b>	OPTIONAL
<b>website</b>	OPTIONAL
<b>birthdate</b>	OPTIONAL



<b>zoneinfo</b>	OPTIONAL
<b>phone_number</b>	OPTIONAL
<b>updated_at</b>	OPTIONAL
<b>email_verified</b>	OPTIONAL
<b>phone_number_verified</b>	OPTIONAL

#### 8.2.4.1.2 Additional claims for PKI Administrator login

This section applies **only** when **'onboarding'** is **enabled** in the deployment configuration settings.

The following User Info claims attributes and properties are managed by SwissPKI during login processing

Attribute/Property	Required
<b>swisspki_permission_template_name_pki_admin</b>	<p><b>MANDATORY</b></p> <p>'Default PKI Administrators' is the default permissions template when initializing SwissPKI</p> <p>If the claim is not present, then the user cannot login and/or is not create on first login</p>

#### Example User Info claim:

```
{
  "sub": "99835f08-0f7d-4877-b534-1b0ef37fc7c8",
  "email_verified": true,
  "name": "Admin PKI",
  "preferred_username": "admin.pki",
  "given_name": "Admin",
  "swisspki_permission_template_name_pki_admin":
    "Default PKI Administrators",
  "family_name": "PKI",
  "email": "demo@example.com"
}
```

#### 8.2.4.1.3 Additional claims for CA Operator and/or Auditor login

This section applies **only** when **'onboarding'** is **enabled** in the deployment configuration settings.

The following User Info claims attributes and properties are managed by SwissPKI during login processing.

Attribute/Property	Required
<b>swisspki_permission_template_name_cao</b>	<p><b>OPTIONAL</b></p> <p>'Default CA Operators' is the default permissions template when initializing SwissPKI</p> <p>If the claim is not present, then the user is created and associated to the Realm but will not have the CA Operator role assigned.</p>
<b>swisspki_permission_template_name_auditor</b>	<p><b>OPTIONAL</b></p> <p>'Default Auditors' is the default permissions template when initializing SwissPKI</p> <p>If the claim is not present, then the user is created and associated to the Realm but will not have the Auditor role assigned.</p>
<b>swisspki_pki_realm</b>	<p><b>MANDATORY</b></p> <p>The name of the SwissPKI Realm</p> <p>If the claim is not present, then the user cannot login and/or is not create on first login</p> <p><b>Note:</b> SwissPKi Realm names are not unique</p>

#### Example User Info claim:

```
{
  "sub": "1f3656fe-8c14-45e9-b79b-01f6103677ae",
  "email_verified": true,
  "swisspki_permission_template_name_cao": "Default CA Operators",
  "swisspki_permission_template_name_auditor": "Default Auditors",
  "name": "John Doe",
  "preferred_username": "john.doe",
  "given_name": "John",
  "swisspki_pki_realm": "Realm I",
  "family_name": "Doe",
  "email": "alice@example.com"
}
```

#### 8.2.4.1.4 Additional claims for RA Operator and/or Authorizer login

This section applies **only** when ‘onboarding’ is **enabled** in the deployment configuration settings.

The following User Info claims attributes and properties are managed by SwissPKI during login processing.

Attribute/Property	Required
<b>swisspki_permission_template_name_rao</b>	<p>OPTIONAL</p> <p>'Default RA Operators' is the default permissions template when initializing SwissPKI</p> <p>If the claim is not present, then the user is created and associated to the Realm but will not have the RA Operator role assigned with a Client.</p>
<b>swisspki_permission_template_name_authorizer</b>	<p>OPTIONAL</p> <p>'Default Authorizers' is the default permissions template when initializing SwissPKI</p> <p>If the claim is not present, then the user is created and associated to the Realm but will not have the Authorizer role assigned with a Client.</p>
<b>swisspki_pki_realm</b>	<p>MANDATORY</p> <p>The name of the SwissPKI Realm</p> <p>If the claim is not present, then the user cannot login and/or is not create on first login</p> <p><b>Note:</b> SwissPKi Realm names are not unique</p>
<b>swisspki_pki_client</b>	<p>OPTIONAL</p> <p>Array of Client names associated with the user</p> <p>If the claim is not present, then the user is created but will have not Client role associated.</p> <p>If the Client is not present in the Realm, then it is created.</p> <p><b>Note:</b> SwissPKi Client names are not unique</p>

**Example User Info claim:**

```
{  
  "sub": "537ccbe0-09ed-472e-be23-89620900660f",  
  "swisspki_permission_template_name_authorizer": "Default Authorizers",  
  "email_verified": true,  
  "name": "Jane Doe",  
  "preferred_username": "jane.doe",  
  "swisspki_permission_template_name_rao": "Default RA Operators",  
  "given_name": "Jane",  
  "swisspki_pki_client": ["Client B", "Client A"],  
  "swisspki_pki_realm": "Realm I",  
  "family_name": "Doe",  
  "email": "bob@example.com"  
}
```

## 8.2.5 Kerberos

Users can log into SwissPKI using Kerberos authentication. Kerberos SSO enables you to manage all your SwissPKI Realms, Users, Roles, Permission Templates and Clients in an LDAP/AD.

Using Kerberos SSO requires the installation of a reverse proxy processing the Kerberos authentication and copying the authenticated username to the HTTP Header **X-Logon-User** attribute value.

### **Important:**

When Kerberos onboarding configuration options is **enabled**: Users logging in via Kerberos will setup the user account and associated permissions and roles upon initial log in. If a user who logs into SwissPKI has an RA Operator and/or Authorizer Role associated with a Client, then the Client is created within the defined Realm if it does not exist. Realms must be defined in SwissPKI.

When Kerberos onboarding configuration options is **disabled**: Users logging in via Kerberos must initially be created as 'validated' users in SwissPKI.

In both cases, an LDAP/AD lookup is performed for the user logging in to retrieve user attributes.

### 8.2.5.1 LDAP Requirements

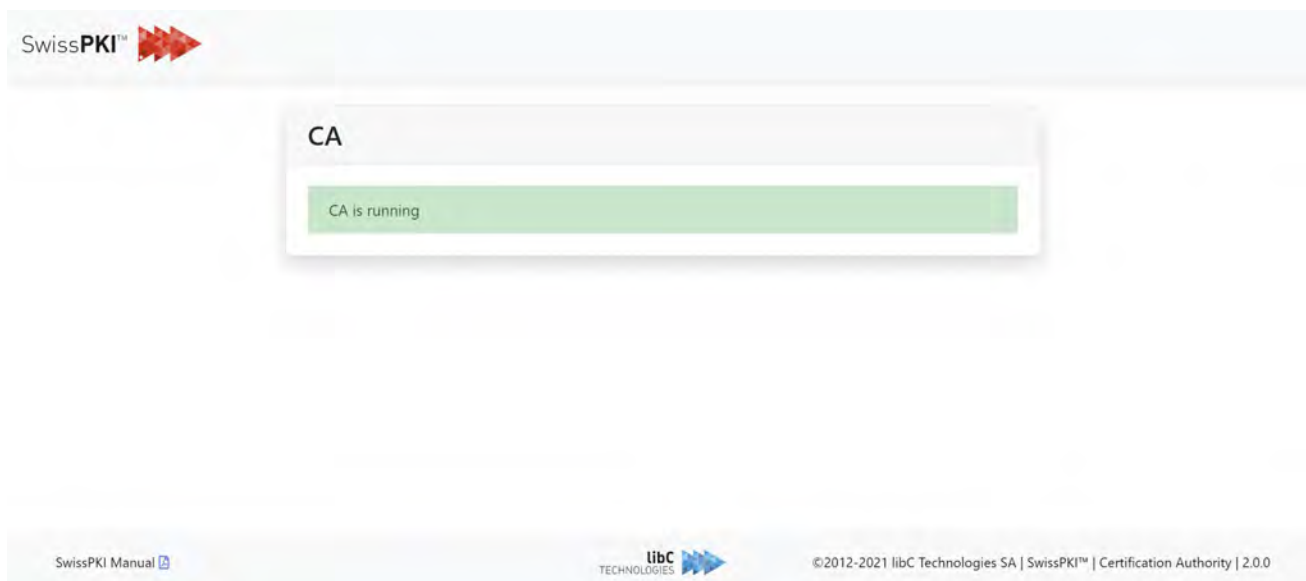
LPDAP v3 server protocol with:

Configuration	Description
<b>Host</b>	LDAP Server hostname or IP
<b>Port</b>	Accessible LDAP port 389 or 636 for SSL
<b>Bind DN</b>	A user with read access to the user BaseDN and SwissPKI BaseDN
<b>Bind Password</b>	A user password
<b>User BaseDN</b>	A distinguished name which identifies the base entry of the users in the DIT
<b>User login attribute</b>	The 'username' to search for in the User BaseDN. Usually set to 'uid' or 'userPrincipalName' on AD
<b>SwissPKI BaseDN</b>	A distinguished name which identifies the base entry of the SwissPKI groups and roles configuration in the DIT

LDAP DIT structure, please refer to [8.3.3.1 User structure](#) and [ss](#)

### 8.3 Certification Authority (CA)

The Certification Authority module is a standalone module running as a background task with no user interaction except for a status page which, when made accessible on the internal network, will display a page accessible under `GET http(s)://<DNS or IP>/ca`. Note that its Health Check URLs are available for monitoring purpose.



The Certification Authority module's functionalities are:

1. Produce certificates by signing the requests sent to it from the Concierge and
2. Produce CRLs/ARLs. CRL/ARL event generation occur in three cases:
  - a. when the Scheduler notifies the CA that a CRL Publication Rule has timed out
  - b. when a CA Operator explicitly generates a CRL by sending a CRL generation event to the CA through the Operator UI or the OpenAPI REST call
  - c. when a CA instance is configured to produce a CRL on every revocation (this option is configured via the Operator UI or OpenAPI REST call)

The Certification Authority module orchestrates all CA instances in a SwissPKI deployment. If you have multiple Realms deployed on a SwissPKI installation with each Realm having multiple CA instances, the Certification Authority module will manage requests for all CA instances across all Realms.

### 8.3.1 Online and Offline Certification Authorities

If you plan to have offline CA instances, then you have the three following options available to achieve this:

1. You deploy a separate SwissPKI environment which contains the offline CA instances and then take the system offline.
2. You deactivate the CA instances on your SwissPKI deployment through the Operator UI or OpenAPI REST call. A deactivated (or disabled) CA instance will not reply to events and therefore taking it 'logically' offline. Additionally, if your CA keys are stored on an HSM, you can:
  - a. Take the CA HSM partition offline. This requires that only offline CA keys be stored on the HSM partition.
  - b. Take the HSM offline if the CA keys are stored on an HSM with no other purpose than storing offline<sup>9</sup> CA keys.
3. Create an Air Gaped CA linked to an Offline CA

Option 2) is the preferred solution for taking CA instances offline as it has the following advantage: updating your SwissPKI environment will also update offline CA instances. You will not have to repeat the procedure on each separate SwissPKI deployment.

Please refer to section *12 Operator UI* for activating/deactivating your CA instances.

---

<sup>9</sup> An offline HSM can also be stored in a secure location

### 8.3.2 CRL Distribution Points (CDP)

Managing HTTP CRL Distribution Points (CDP) can become quite a project as the generated CRLs need to be copied to HTTP server file system locations and renamed to match the URI published in the issued certificates. Additionally, you may generate multiple different CDPs depending on the number of issuing CAs and CDP organization as a CA may produce different CDPs based on your certificate policies.

The CDP Module exposes a status page on the deployed `GET http(s)://<DNS or IP>/cdp`. Note that its Health Check URLs are available for monitoring purpose.



SwissPKI™ 

**CDP**

Number of valid requests: 0

Number of invalid requests: 0

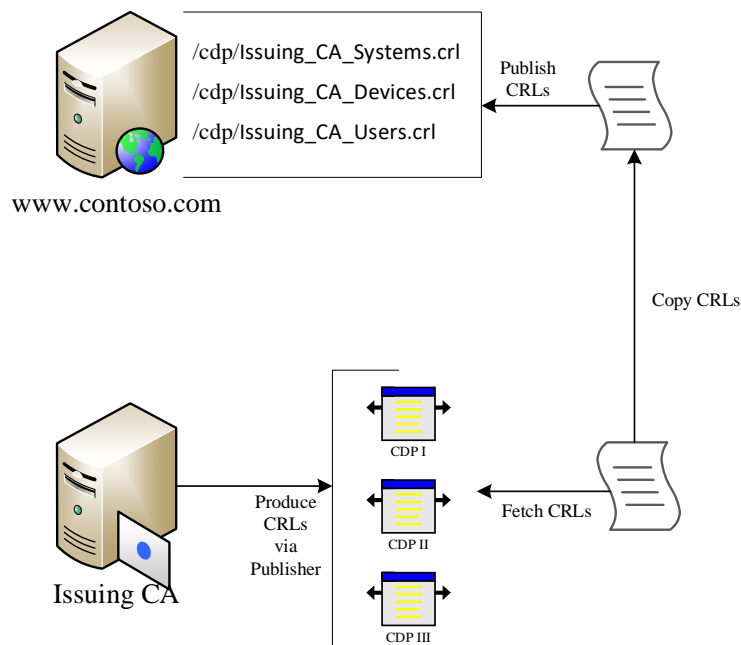
SwissPKI Manual [↗](#)  ©2012-2021 libC Technologies SA | SwissPKI™ | CDP | 2.0.0



As an example, take the following Use Case:

1. An Issuing CA publishes three different CDPs:
  - a. [http://www.contoso.com/cdp/Issuing\\_CA\\_Systems.crl](http://www.contoso.com/cdp/Issuing_CA_Systems.crl)
  - b. [http://www.contoso.com/cdp/Issuing\\_CA\\_Devices.crl](http://www.contoso.com/cdp/Issuing_CA_Devices.crl)
  - c. [http://www.contoso.com/cdp/Issuing\\_CA\\_Users.crl](http://www.contoso.com/cdp/Issuing_CA_Users.crl)

Your Issuing CA will regularly produce three CDPs a), b) and c) which you will need to copy to the [www.contoso.com](http://www.contoso.com) Web server file system.



The SwissPKI CDP module solves the issue of copying and renaming CRL files to the web server CDP URIs by exposing a URI which will always return the latest CRL for the configured CDP.

As a CA Operator, you create HTTP CDP URLs for your Issuing CA as illustrated below. From the Certification Authority CDP editor, create a CDP and update the entry for each URL with the value displayed in the blue box:

## CRL Distribution Point | 'Issuing CA'

**Name\***  **URL\***

You can update the CDP URI to  in order to access the latest generated CRL. This option requires deploying the CDP server.

Include CRL Distribution Point  Include Reason Code

**Reason Code**

The three CDP entries have their URL displayed in the Certification Authority's CDP view:

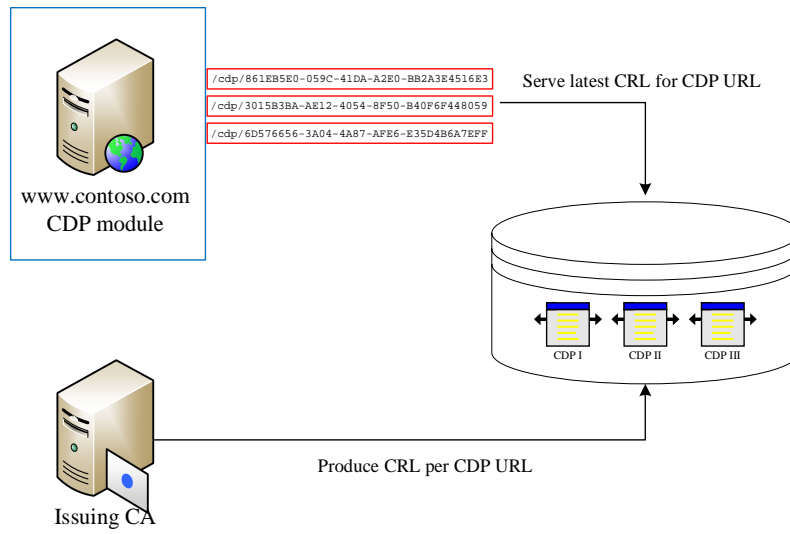
## CRL Distribution Points | 'Issuing CA' +

**Search**

Created	Modified	Name	URL	Editable	Active	Actions
26.01.21	26.01.21	Issuing CA I	http://www.contoso.com/cdp/861EB5E0-059C-41DA-A2E0-BB2A3E4516E3	✓	✓	
26.01.21	26.01.21	Issuing CA II	http://www.contoso.com/cdp/3015B3BA-AE12-4054-8F50-B40F6F448059	✓	✓	
26.01.21	26.01.21	Issuing CA III	http://www.contoso.com/cdp/6D576656-3A04-4A87-AFE6-E35D4B6A7EFF	✓	✓	

Showing 1 to 3 of 3 entries

The SwissPKI CDP module is deployed with an external DNS of www.contoso.com. The CDP module will serve the latest CRL for each CDP through its preconfigured `cdp/<UUID>` URI as illustrated below:



Please refer to section 12 *Operator UI* for creating and updating CDP end points.

### 8.3.3 Authority Information Access (AIA)

As for CDPs, managing HTTP/HTTPS Authority Information Access (CAIssuer field in AIA extension) can become quite a project as the issued Certification Authority certificates need to be copied to HTTP/HTTPS server file system locations and optionally renamed to match the URI published in the AIA extension of the issued certificates.

The AIA Module exposes a status page on the deployed `GET http(s)://<DNS or IP>/aia`. Note that its Health Check URLs are available for monitoring purpose.



SwissPKI™ 

#### AIA

Number of valid requests: 0

Number of invalid requests: 0

SwissPKI Manual 

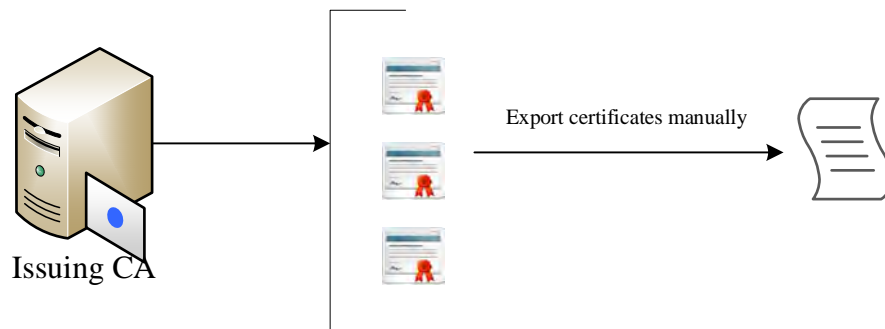
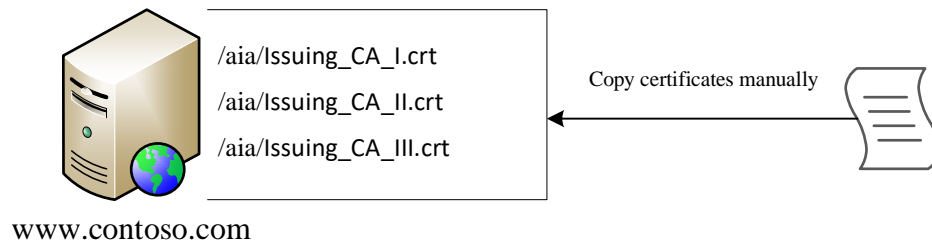
libC  
TECHNOLOGIES 

©2012-2021 libC Technologies SA | SwissPKI™ | AIA | 2.0.0

As an example, take the following Use Case:

1. Issuing CAs certificates published to three different URLs:
  - a. `http://www.contoso.com/aia/Issuing_CA_I.crt`
  - b. `http://www.contoso.com/aia/Issuing_CA_II.crt`
  - c. `http://www.contoso.com/aia/Issuing_CA_III.crt`

Your CA certificates must be copied to the HTTP web server AIA URI file system location.



The SwissPKI AIA module solves the issue of copying issuer certificates files to the web server AIA URIs by exposing a URI which will always return the certificate for the AIA Rule.

As a CA Operator, you create HTTP/HTTPS AIA URLs for your Issuing CA as illustrated below. From the Certification Authority AIA editor, create an AIA and update the entry for each URL with the value displayed in the blue box:

## Create AIA | 'Issuing CA'

**Rule Name\***

Issuig CA AIA Rule

**AIA Rule description**

My AIA Rule for mapping the CA certificate to an URL

**Available Certification Authorities**

Issuing CA | 008FFE00C2117798357D1BDD87AE3134EC | 2021-01-26T06:15:50 | 2046-01-26T06:15:50

Back Create

One (in this example instead of the three mentioned above for sake of simplicity) AIA entry URL displayed in the Certification Authority's AIA view:

## Authority Information Access | 'Issuing CA' +

**Search**

Search

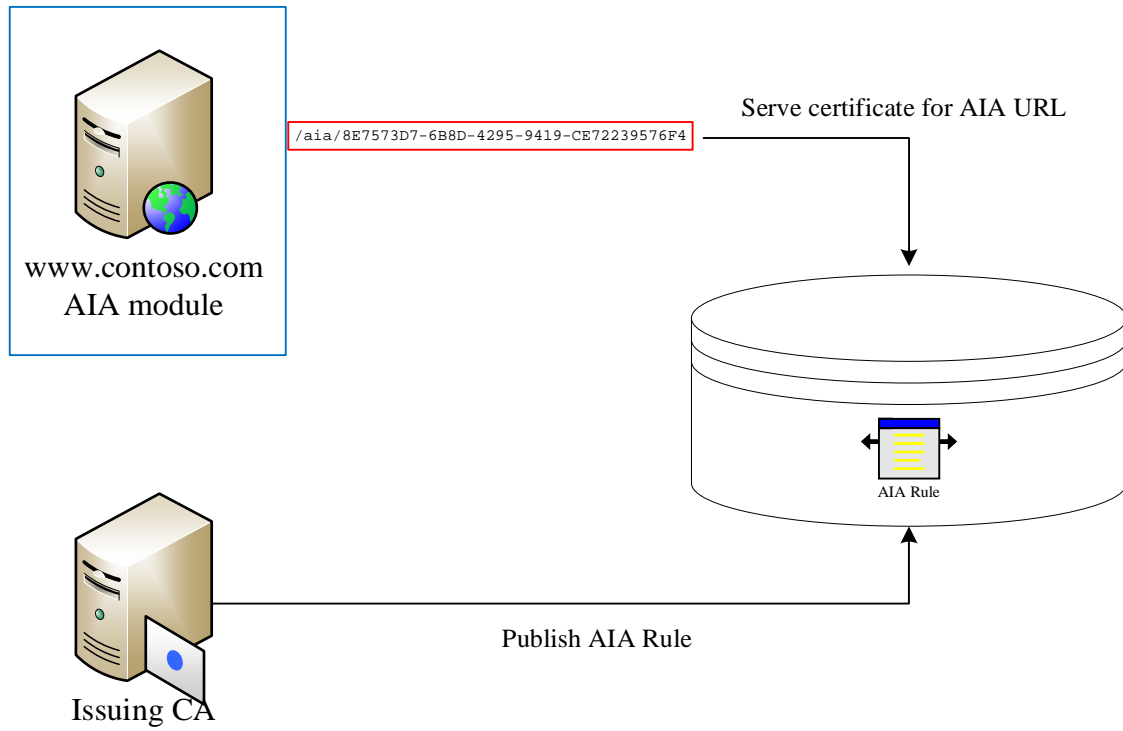
Created	Modified	Rule	URI	Common Name	Start validity	End validity	Actions
26.01.21	26.01.21	Issuig CA AIA Rule	aia/8E7573D7-6B8D-4295-9419-CE72239576F4	Issuing CA	26.01.21	26.01.46	 

Showing 1 to 1 of 1 entries

Previous 1 Next

Back

The SwissPKI AIA module is deployed with an external DNS of www.contoso.com. The AIA module will serve the issuing CA certificate through its preconfigured aia/<UUID> URI as illustrated below:



In the certificate policy template editor, you can then use the AIA `CAIssuer` URL as illustrated below:



Please refer to section *12 Operator UI* for creating and updating AIA end points.



## 8.4 Automatic Certificate Management Environment (ACME)

The ACME module (RFC8555) is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/acme`. Note that its Health Check URLs are available for monitoring purpose.



The ACME Module serves clients by exposing the registration directory to ACME clients. The ACME registration URL is composed of the ACME Module DNS address and a generated URI when mapping a policy instance (certificate product) to a Client as in the following example:

For example, the ACME URL for 'Client A' for issuing 'SSL Gold' certificate types is:

## ACME Policy Instance 'SSL Gold' | 'Issuing CA' +

Search

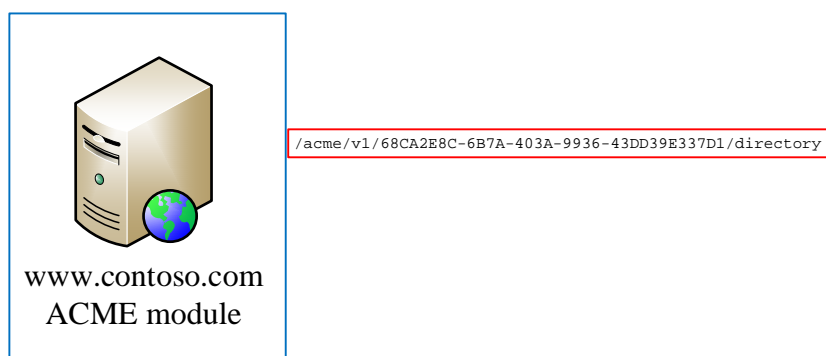
Created	Modified	Client name	ACME URI	Actions
26.01.21	26.01.21	Client A	<a href="/acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/directory">/acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/directory</a>	 

Showing 1 to 1 of 1 entries

Previous **1** Next

Back

Which resolves to the HTTPS URL on the deployed ACME Service to:



Using the ACME directory entry point provided to 'Client A' for product 'SSL Gold':

```
GET https://www.contoso.com/acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/directory
```

Will then transparently resolve the protocol entry points for 'Client A' and product 'SSL Gold' as follow:

```
HEAD /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/newNonce
```

```
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/newAccount
```

```
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/acct/<accountUuid>
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/newOrder
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/order/<orderUuid>
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/order/<orderUuid>/finalize
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/newAuthz
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/authz/<authzUuid>
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/chall/<challengeUuid>
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/revokeCert
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/keyChange
GET /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/acct/<accountuuid>/orders
POST /acme/v1/68CA2E8C-6B7A-403A-9936-43DD39E337D1/cert/<orderuuid>/<certuuid>
```

Please refer to section *12 Operator UI* for creating ACME end points.

Please refer to <https://support.swisspki.com/support/solutions/articles/44001873643--spki2-requesting-acme-tokens-with-certbot> for using the ACME client certbot.

## 8.5 Microsoft CES/CEP (MSCA)

The Microsoft CES/CEP module is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/msca`. Note that its Health Check URLs are available for monitoring purpose.



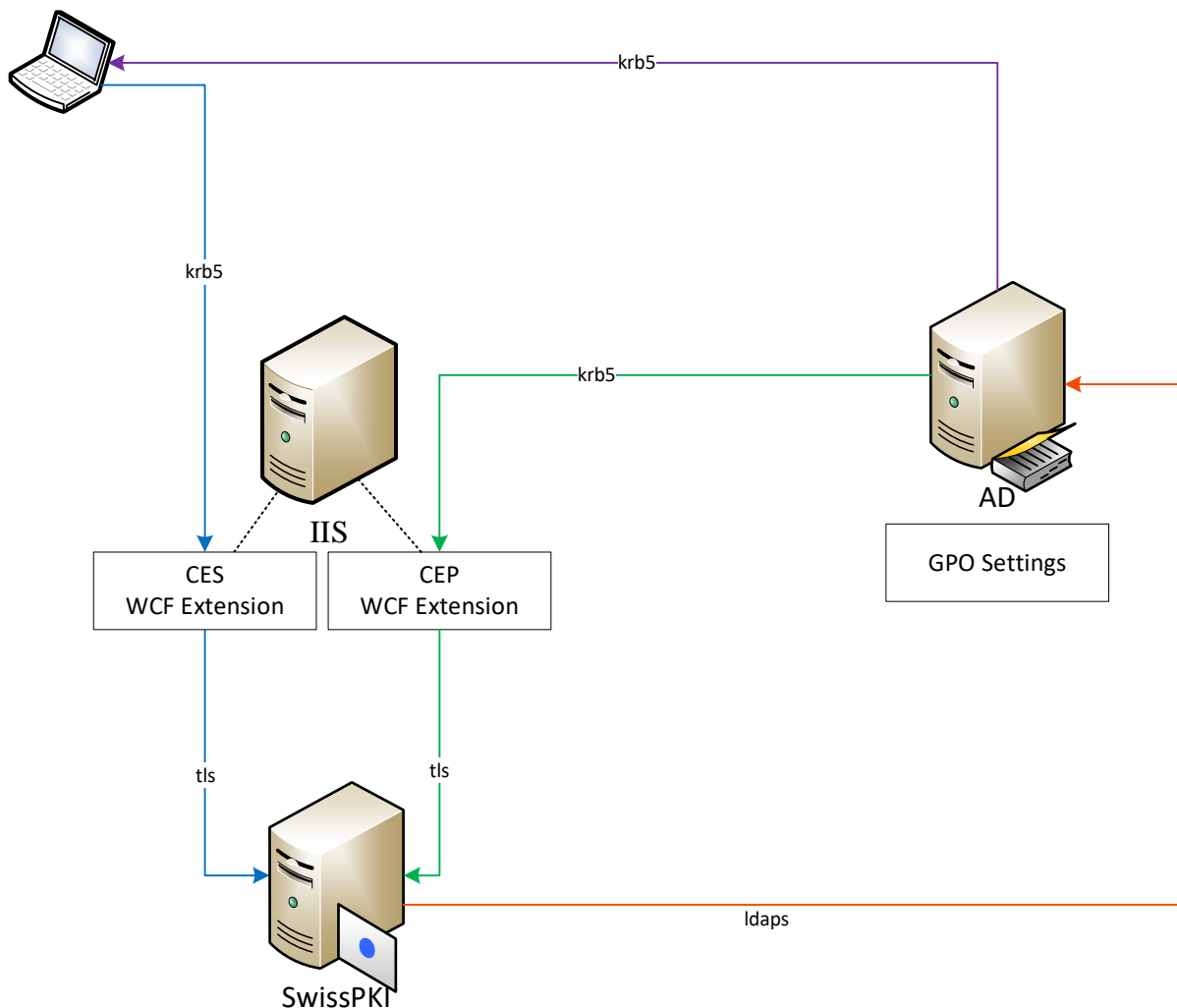
SwissPKI™ 

Microsoft CES & CEP up since 2021-10-24T16:56:24.16701

Number of valid requests: 0
Number of invalid requests: 0

SwissPKI Manual [🔗](#)  ©2012-2021 libC Technologies SA | SwissPKI™ | Microsoft CES/CEP | 2.0.0

The Microsoft CES/CEP module enabled Microsoft end users and devices to enroll and renew certificates with SwissPKI whether the certificates are issued from an organization PKI or SwissSign public trust certificates manually or automatically.



### 8.5.1 Microsoft CEP

The Microsoft CEP is an additional module which installs on the Microsoft Domain as an IIS WCF Extension. The Microsoft CEP WCF Extension serves the policy templates you created in SwissPKI as a CA Operator to the AD. AD will then push the configured certificate policy templates to the devices in the Microsoft Domain, whether those devices are joined or not in the Microsoft Domain.

The URL connection string of the Microsoft CEP WCF Extension is configured in AD through the GPO. This URL is generated with a unique identifier when creating a Microsoft CES/CEP entity in SwissPKI.

The policy certificate templates created in SwissPKI must be of type Microsoft or Microsoft SwissSign Public Trust as they contain all Microsoft AD configuration elements. You cannot push policies of other types to the Microsoft AD.

Detailed Microsoft CEP configuration is available on the support website at <https://support.swisspki.com/support/solutions/articles/44001819320-microsoft-ces-and-cep-setup>.

### 8.5.2 Microsoft CES

The Microsoft CES is an additional module which installs on the Microsoft Domain as an IIS WCF Extension. The Microsoft CES WCF Extension manages the certificate enrollment and revocation from the Microsoft devices, users, and forwards them to SwissPKI. The CES registration URL is pushed to the devices and users via Active Directory which obtained the registration URL from the Microsoft CEP module.

Detailed Microsoft CES configuration is available on the support website at <https://support.swisspki.com/support/solutions/articles/44001819320-microsoft-ces-and-cep-setup>.

### 8.5.3 Microsoft Service on SwissPKI

The SwissPKI Microsoft Service connects to Active Directory to apply the certificate policy settings from incoming requests. Depending on the certificate policy settings, SwissPKI may query entries from Active Directory to populate certificate content or publish issued certificates into Active Directory (e.g., encryption certificates for S/MIME purpose)

Please refer to section *12 Operator UI* for creating Microsoft CES/CEP end points and <https://support.swisspki.com> for detailed CES and CEP module setup and configuration with Microsoft AD Kerberos integration.

## 8.6 Online Certificate Status Protocol (OCSP)

The OCSP module (RFC6960) is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/ocsp`. Note that its Health Check URLs are available for monitoring purpose.



The SwissPKI OCSP Service serves real time certificate status to client requests for one or multiple referenced Certification Authorities. The OCSP Service produces tokens on the fly from certificate information stored in the DB (RO access) and processes only replies for Certification Authorities registered with the Service. The OCSP signer certificate is issued by the referenced Certification Authority. Additionally, the OCSP Service supports CT Log stapling.

Initializing an OCSP Service in SwissPKI will make its Service URL immediately available to the clients and Issuing Certification Authorities including the OCSP URL in the Authority Information Access extension of the issued certificates (if indicated in the extension)

Deployed OCSP Service on www.contoso.com



Handling certificate request status for the Certification Authority 'Issuing CA'

### Mapped Certification Authorities for OCSP | 'OCSP' +

OCSP Server initialized and ready at URI: '[protocol://your.ocsp.server]/ocsp/sign/94C6562E-049E-4C8F-9210-D42249B71CC0'

Policy	OCSP	Subject CN	Serial#	Start validity	End validity	Actions
OCSP	Issuing CA	OCSP	00D275D94CC9939E6E58C0B686880651D7	28.01.21	28.01.24	

Showing 1 to 1 of 1 entries

Previous 1 Next  
Back

With issued certificates referencing the OCSP URL in the certificate policy template  
<http://www.contoso.com/ocsp/sign/94C6562E-049E-4C8F-9210-D42249B71CC0>

Authority Information Access 

Authority Information Access is critical  

**OCSP**

URI   

<http://www.contoso.com/ocsp/sign/94C6562E-049E-4C8F-9210-D42249B71CC0>

+ Add item

Please refer to section 12 Operator UI for initializing OCSP end points.

Please refer to <https://support.swisspki.com/support/solutions/articles/44001819455-testing-online-certificate-server-protocol> for client OCSP requests.

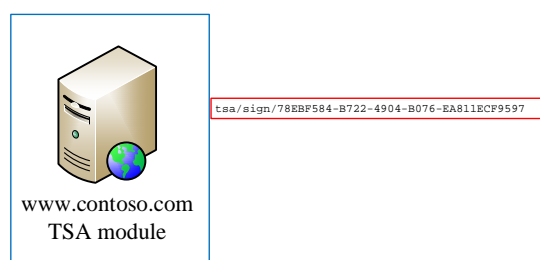


## 8.7 Time Stamp Authority (TSA)

The TSA module (RFC3161) is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/tsa`. Note that its Health Check URLs are available for monitoring purpose.



Initializing a SwissPKI TSA Service makes it immediately available to clients by publishing the URI to the deployed server. Additionally, Document Signer Servers can reference TSA URLs if you wish to include TSA time stamps in digital signatures.



The TSA Service initialized and signed by the 'Issuing CA' Certification Authority and its generated URI.

## Time Stamp Authority (TSA) | 'TSA'

Time Stamp Authority initialized and ready at URI: '[protocol://your.tsa.server]/tsa/sign/78EBF584-B722-4904-B076-EA811ECF9597'

**Name**  
TSA

**Description**  
TSA

**Signature algorithm**  
sha256

Include CMS Algorithm Protect Attribute

Use NTP time source instead of local server time

**Comment**  
TSA

**TimeStamp Authority Policy Id**  
2.16.756.3.2.1

Automatic certificate renewal (20 days before expiration)

Please refer to section 12 *Operator UI* for initializing TSA end points.

Please refer to <https://support.swisspki.com/support/solutions/articles/44001818710-testing-time-stamp-authority> for TSA client requests.

## 8.8 SCEP/NDES (SCEP)

The SCEP/NDES is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/scep`. Note that its Health Check URLs are available for monitoring purpose.



The screenshot shows the SwissPKI SCEP status page. At the top left is the SwissPKI logo. The main content area is titled 'SCEP' and contains two horizontal bars: a green bar for 'Number of valid requests: 0' and a red bar for 'Number of invalid requests: 0'. The footer contains a link to the 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | SCEP | 2.0.0'.

## 8.9 SCION PKI Adapter (SCION)

The Publisher is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/publisher`. Note that its Health Check URLs are available for monitoring purpose.



## 8.10 Certificate Management Protocol (CMP)

The CMP module (RFC6712) is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/cmp`. Note that its Health Check URLs are available for monitoring purpose. Note that this service only exposes certificate registration and revocation.



The screenshot shows the SwissPKI interface for the Certificate Management Protocol (CMP). At the top left, there is a header with the SwissPKI logo. The main content area is titled "CMP" and contains two status bars: a green bar indicating "Number of valid requests: 0" and a red bar indicating "Number of invalid requests: 0". At the bottom of the page, there is a footer containing a link to the "SwissPKI Manual", the libC Technologies logo, and the copyright notice "©2012-2021 libC Technologies SA | SwissPKI™ | CMP Protocol Handler | 2.0.0".

Initializing a CMP end point through the Operator UI will expose the service to the end users on the deployed CMP module.

## Certificate Management Protocol (CMP) | 'CMP Sample'

CMP enabled at URI: '[protocol://your.cmp.server/cmp/6F679437-5356-45EC-8AC6-B9753C3848B3]'

### Name\*

CMP Sample

### Description\*

CMP Sample

### Comment

CMP Sample

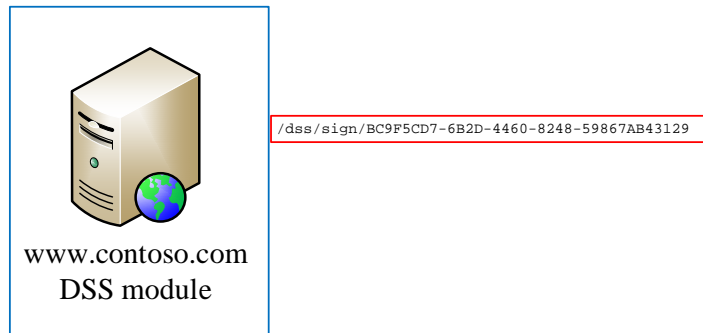
### Message authentication signing algorithms

RSA PSS with sha256  
EC DSA with sha256

- Confirm certificate request message
- Require message timestamp
- Verify message timestamp bias
- Automatic certificate renewal (20 days before expiration)
- Include CA certificate in response
- Include responder certificate
- Confirm wait time

The CMP URI is then immediately available via

<https://www.contoso.com/cmp/6F679437-5356-45EC-8AC6-B9753C3848B3>:



To issue or revoke certificates through with the CMP Client, you will need to register a signature certificate with the Client within your PKI Realm. You can drag and drop CMP client signing certificates to the client's CMP settings via the Operator UI. Additionally, you will also need to register an Issuing CA with the CMP. To do so, the Issuing CA will need to have two certificates (a CMP signing and Cipher certificate) registered with the CMP Module. The communication between client and server requires digitally signed and encrypted client CMP requests over CMS for the CMP backend to validate the incoming requests. Besides, if you use CMP client signing certificates which are not issued by a CA within your Realm, you will need to register its certificate trust chain (Issuing and Root CAs) with the Realm's Trust Anchor settings. Registering Realm's Trust Anchors is done by a PKI Administrator role. Please refer to section *12 Operator UI* for initializing CMP end points.

## 8.11 Document Signer Server (DSS)

The DSS module (eIDAS/ETSI) is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/dss`. Note that its Health Check URLs are available for monitoring purpose.



You initialize a DSS instance through the Operator UI by issuing a signing certificate using an Issuing CA from your Realm. Once the DSS key and certificate is issued, the DSS Service is online and accessible to clients for sending (HTTP POST) signing requests to the DSS Service. A URI is generated for the initialized instance:



## Document Signer | 'DSS Sample'

Document Signer enabled at URI: '[protocol://your.dss.server]/dss/sign/BC9F5CD7-6B2D-4460-8248-59867AB43129'

**Name\***

DSS Sample

**Description\***

DSS Sample

**Comment**

DSS Sample

**Container\***

None

**Format\***

PAdES

**Base line\***

Baseline-B

**Signature Algorithm\***

sha256

**Time Stamp Authorities**

Nothing selected

**Envelope\***

Enveloped

Automatic certificate renewal (20 days before expiration)

Allow signing with expired certificate

The DSS URI is available on the deployed DSS Module

<https://www.contoso.com/dss/sign/BC9F5CD7-6B2D-4460-8248-59867AB43129> :



/dss/sign/BC9F5CD7-6B2D-4460-8248-59867AB43129

Three formats of advanced signature and one format of signature container are specified in the European Telecommunications Standards Institute (ETSI) standards, namely:

1. XML advanced electronic signature (XAdES), based on XML signatures.
2. PDF advanced electronic signature (PAdES), based on PDF signatures.
3. CMS advanced electronic signature (CAAdES), based on Cryptographic Message Syntax (CMS).
4. Associated Signature Container (ASiC) based on ZIP format and supporting XAdES and CAAdES signature formats.

When signing a single document, the format of signature to choose typically depends on the format of the document to sign:

1. XML documents are suggested to be signed using XAdES signature format (either with enveloped or enveloping packaging).
2. PDF documents are suggested to be signed using PAdES signature format.
3. Binary files are suggested to be signed with XAdES or CAdES signature formats (with enveloping packaging).

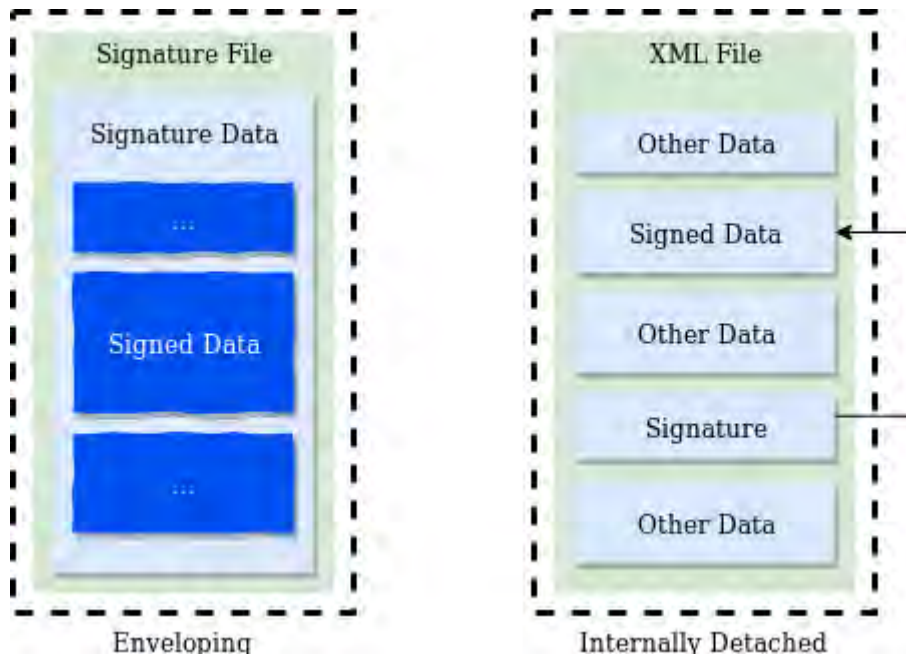
When signing/sealing multiple documents, it is suggested to use ASiC containers.

The current DSS version manages single file signing but supports all the formats defined by the ETSI standards.

A signature can be enveloped or detached, whether it is included as an element of the file containing the signed data or a separate signature file is created, that refers to the data upon which it bears:



It can also be enveloping when the signed data are included as a sub-element of the signature, and in exceptional cases where the signature is detached but both the signed data and the signature data are included in another file, it is called internally detached. (Internally detached signatures are very rarely used).



Not all signature formats support these various locations and positioning of a signature, and a simplified overview can be given by the following:

1. Enveloped signatures can be created using XAdES or PAdES formats
2. Detached signatures can be created using XAdES or CAdES formats
3. Enveloping signatures can be created using XAdES or CAdES formats
4. Internally detached signatures can only be created using XAdES format.

Please refer to section 12 *Operator UI* for initializing DSS end points.

## 8.12 Publisher

The Publisher is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/publisher`. Note that its Health Check URLs are available for monitoring purpose.



Its function is to publish issued certificates and CRLs/ARLs to remote servers and/or file system. Supported publication end points are file system (accessible to the process), SFTP and LDAP servers. Each Publisher supports multiple end point publishing configuration settings. For example, you can configure three different file system locations, two separate SFTP servers and four LDAP servers per single Publisher instance deployed within your Realm.

To activate publication of issued certificates and CRLs/ARLs, you link any Certification Authority with a Publisher instance. Note that you can link one Certification Authority with multiple publishers depending on the rules set you plan to configure. When a Certification Authority produces CRLs/ARLs or issues certificates, the linked Publishers will invoke each configured publication end point and write the files to the target destination.

By default, Publishers will dispatch all certificates and CRLs/ARLs to the target destinations but in some cases, you may not wish to publish certificates for specific clients. You can suppress certificate

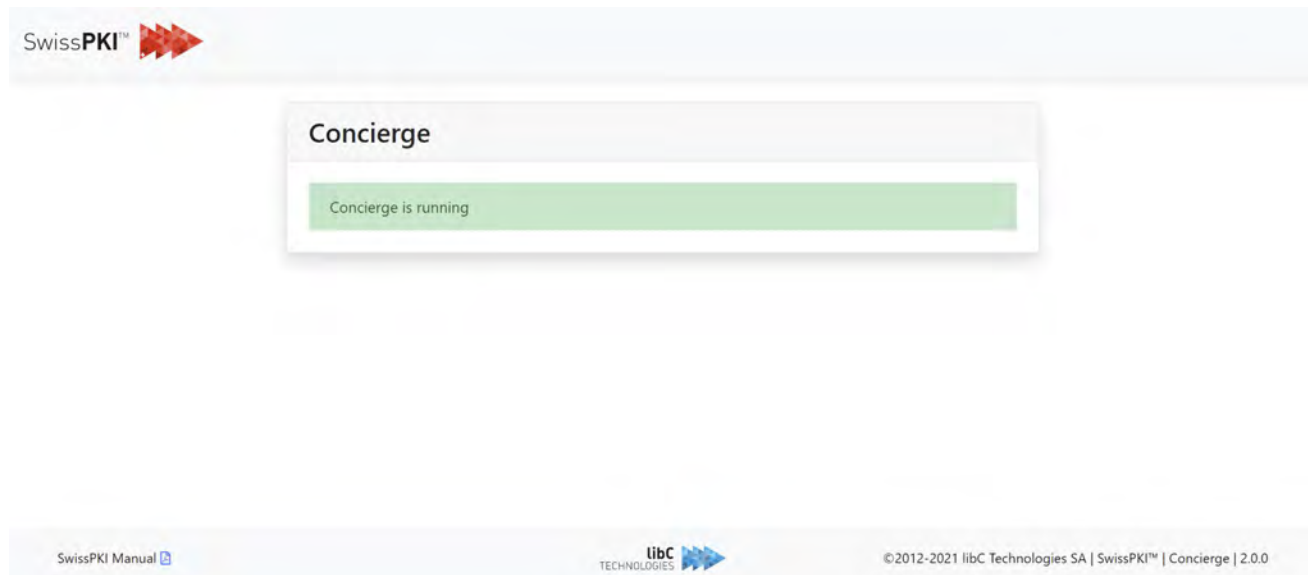
publication in the certificate policy instance (certificate product) associated to a Client by explicitly disabling the certificate publication rule for the selected product.

Certificates, CRLs and ARLs are published in DER format. When published to file systems and SFTP servers, the file naming will include the serial number and the extension in either one of .cer, .crl or .arl. Additionally, CRLs and ARLs will be prefixed with the CDP name. For the global CRL, the prefix file name is 'global.'

Please refer to section *12 Operator UI* for initializing Publishers.

## 8.13 Concierge

The Concierge is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/concierge`. Note that its Health Check URLs are available for monitoring purpose.



The Concierge is the SwissPKI certificate management workflow engine with fork/join and scheduling capabilities. It manages both synchronous and asynchronous tasks. Its principal duties are:

- Orchestrating certificate issuance and renewal requests between SwissPKI modules (ACME, Microsoft CES/CEP, CMP, CA, Registration UI and Operator UI)
- Orchestrating certificate revocation requests between SwissPKI modules (ACME, Microsoft CES/CEP, CMP, CA, Registration UI and Operator UI)
- Orchestrating authorization, notifications, and pre/post validation tasks
- Sending Emails with optional S/MIME capability

The Concierge makes extensive usage of message queues. Therefore, you must ensure that the AMQP server is configured with persistent message queues in case you shutdown/restart the server.

## 8.14 Scheduler

The Scheduler is a standalone module running as a Service with no user interaction except for a status page which, when made accessible, will display a page accessible under `GET http(s)://<DNS or IP>/scheduler`. Note that its Health Check URLs are available for monitoring purpose.

### Scheduler is up

Scheduler is running

- Job 'CABSuffixDownload' will trigger next at 'Sun Jan 29 02:00:00 CET 2023'
- Job 'CertificateDownloadLinkCleanUp' will trigger next at 'Fri Jan 27 22:15:00 CET 2023'
- Job 'CertificateRenewal' will trigger next at 'Wed Jan 25 11:00:00 CET 2023'
- Job 'CertificateRevocation' will trigger next at 'Wed Jan 25 11:00:00 CET 2023'
- Job 'DailyTaskScheduler' will trigger next at 'Wed Jan 25 23:00:00 CET 2023'
- Job 'DnsCheckScheduler' will trigger next at 'Wed Jan 25 23:00:00 CET 2023'
- Job 'EntityRenewal' will trigger next at 'Wed Jan 25 23:30:00 CET 2023'
- Job 'ExternalCertificateStatus' will trigger next at 'Wed Jan 25 22:00:00 CET 2023'
- Job 'HSMStatusCheck' will trigger next at 'Wed Jan 25 11:00:00 CET 2023'
- Job 'JobScheduler' will trigger next at 'Wed Jan 25 11:00:00 CET 2023'
- Job 'LDAP Importer' will trigger next at 'Wed Jan 25 22:30:00 CET 2023'
- Job 'MicrosoftAutoRevocation' will trigger next at 'Wed Jan 25 22:30:00 CET 2023'
- Job 'OrderScheduler' will trigger next at 'Wed Jan 25 11:00:00 CET 2023'
- Job 'PublisherCleanup' will trigger next at 'Thu Jan 26 03:00:00 CET 2023'
- Job 'WaitingCertificateOrdersCleanUp' will trigger next at 'Wed Jan 25 10:54:00 CET 2023'

Note that the Scheduler is the only process in SwissPKI which cannot scale horizontally as it executes scheduled crontab like tasks. Starting multiple instances of the Scheduler will start the crontabs anew in parallel, causing in some cases renewals and notifications being processed multiple times.

The tasks processed on a regular basis by the Scheduler are:

Schedule	Expression	Description
<b>CRLPublication_&lt;id&gt;</b>	User defined	CRL/ARL generation based on the CRL Publication Rules defined by the CA Operator
<b>CertificateRenewal</b>	0 0 * ? * *	Certificate renewals for certificates with associated renewal rules which trigger either renewal notifications or actual certificate renewals.
<b>CertificateRevocation</b>	0 0 0/1 ? * *	Certificate revocations for renewed certificates which have a revocation configured in their associated renewal rule
<b>EntityRenewal</b>	0 30 23 ? * *	PKI entities renewals for DSS, CMP, TSA and OCSP of enabled for the instances
<b>HSMStatusCheck</b>	0 0/15 * ? * *	HSM status checks. Controls access to the private keys on all active HSM partitions
<b>ExternalCertificateStatus</b>	0 0 22 ? * *	External Certificate Status check. Update the certificate status of imported certificates using their CDP extensions to retrieve the revocation state. Connects to Internet via HTTP for CDP/OCSP checks
<b>JobScheduler</b>	0 0/15 * ? * *	Job scheduling
<b>OrderScheduler</b>	0 0/30 * ? * *	Reschedule orders which have no corresponding jobs
<b>DailyTaskScheduler</b>	0 0 23 * * ?	Daily job for cleanup tasks



<b>EmailValidationLinkScheduler</b>	0 0 0/1 * * ?	Schedule email validation link clean up. Aborts certificate orders where the email link lifespan expired
<b>DnsCheckScheduler</b>	0 0 23 * * ?	Notify owners of expiring DNS pre-validations, expiring DNS random values or expired DNS random values.
<b>CertificateDownloadLinkCleanUp</b>	0 15 22 ? * 6L	Certificate download link clean up task. Deletes certificate download link which are older than 3 months.
<b>PublisherCleanup</b>	0 0 3 ? * *	Clean up expired certificates from LDAP Servers and other publication destinations.  Send notifications to CA Operators if the published CAs and CRL/ARL have expired (LDAP only)
<b>MicrosoftAutoRevocation</b>	0 30 22 ? * *	Auto revocation for Microsoft CES issued certificates.  For a certificate to be auto revoked, the following conditions must be met: <ol style="list-style-type: none"> <li>1. certificate is of type Microsoft</li> <li>2. certificate is valid and not revoked</li> <li>3. certificate MUST contain a SAN UPN (user) and/or SAN DNS (machine) for it to be retained in the list of auto revocation.</li> <li>4. For all Microsoft certificates on the deployed instance, for each CEP/CES service deployed on the instance <ol style="list-style-type: none"> <li>a. Get valid non revoked certificates</li> <li>b. For each AD linked to a CEP/CES</li> <li>c. Search certificate using the UPN/DNS attribute in AD</li> <li>d. If certificate is not located in ADs (for all</li> </ol> </li> </ol>

		UPN/DNS values as some certificates may have multiple SAN entries), revoke the certificate e. All other Microsoft certificates without UPN/DNS attributes are listed to the log with their serial, DN and start/end validity dates
<b>CABSuffixDownload</b>	0 0 2 ? * SUN	Automatically download the CAB Suffix list from the configured URL and notify administrators of diffs upon import.
<b>WaitingCertificateOrdersCleanUp</b>	0 0/2 * ? * *	Cancel waiting certificate orders after n days.
<b>ShelveRegistrationDocuments</b>	0 30 22 ? * *	When S3 is enabled on the Realm, archives registration documents to the defined S3 bucket

**Note:** schedules and calendars are configured in the `scheduler.conf` file

Cron expressions <sup>10</sup> are comprised of 6 required fields and one optional field separated by white space. The fields respectively are described as follows:

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

The '\*' character is used to specify all values. For example, "\*" in the minute field means "every minute".

<sup>10</sup> <http://www.quartz-scheduler.org>

The '?' character is allowed for the day-of-month and day-of-week fields. It is used to specify 'no specific value'. This is useful when you need to specify something in one of the two fields, but not the other.

The '-' character is used to specify ranges. For example "10-12" in the hour field means "the hours 10, 11 and 12".

The ',' character is used to specify additional values. For example, "MON,WED,FRI" in the day-of-week field means "the days Monday, Wednesday, and Friday".

The '/' character is used to specify increments. For example, "0/15" in the seconds field means "the seconds 0, 15, 30, and 45". And "5/15" in the seconds field means "the seconds 5, 20, 35, and 50". Specifying '\*' before the '/' is equivalent to specifying 0 is the value to start with. For each field in the expression, there is a set of numbers that can be turned on or off. For seconds and minutes, the numbers range from 0 to 59. For hours 0 to 23, for days of the month 0 to 31, and for months 1 to 12. The '/' character simply helps you turn on every "nth" value in the given set. Thus "7/6" in the month field only turns on month "7", it does NOT mean every 6th month, please note that subtlety.

The 'L' character is allowed for the day-of-month and day-of-week fields. This character is shorthand for "last", but it has different meaning in each of the two fields. For example, the value "L" in the day-of-month field means "the last day of the month" - day 31 for January, day 28 for February on non-leap years. If used in the day-of-week field by itself, it simply means "7" or "SAT". But if used in the day-of-week field after another value, it means "the last xxx day of the month" - for example "6L" means "the last Friday of the month". You can also specify an offset from the last day of the month, such as "L-3" which would mean the third-to-last day of the calendar month. *When using the 'L' option, it is important not to specify lists, or ranges of values, as you will get confusing/unexpected results.*

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

The 'L' and 'W' characters can also be combined for the day-of-month expression to yield 'LW', which translates to "last weekday of the month".

The '#' character is allowed for the day-of-week field. This character is used to specify "the nth" XXX day of the month. For example, the value of "6#3" in the day-of-week field means the third Friday of

the month (day 6 = Friday and "#3" = the 3rd one in the month). Other examples: "2#1" = the first Monday of the month and "4#5" = the fifth Wednesday of the month. Note that if you specify "#5" and there is not 5 of the given day-of-week in the month, then no firing will occur that month. If the '#' character is used, there can only be one expression in the day-of-week field ("3#1,6#3" is not valid since there are two expressions).

The legal characters and the names of months and days of the week are not case sensitive.

## 8.15 DNS

DNS validation methods supported by SwissPKI and its characteristics

### 8.15.1 Challenge Tokens

The validation methods make use of different challenge token types.

#### 8.15.1.1 Random Value conforming to BR

BR requires at least 112 bits of entropy in a Random Value.

Random Values generated by SwissPKI consist of a byte sequence of 160 bits (20 bytes) which will then be base64url encoded before they will be provided to the applicant (download as a file-download, copy & paste, or sent per email).

#### 8.15.1.2 Request Token conforming to BR

Request Tokens as defined in BR Section 1.6.1 are currently not used.

##### 8.15.1.2.1 ACME Challenge Token

RFC 8555 requires at least 128 bits of entropy in its tokens.

ACME Tokens generated by SwissPKI consist of a byte sequence of 160 bits (20 bytes) and be encoded as defined in http-01 (Section 8.3 of RFC 8555) or dns-01 (Section 8.4 of RFC 8555).

##### 8.15.1.2.2 ACME Key Authorization

Key authorization strings are generated as defined in RFC 8555 Section 8.1

#### 8.15.1.3 Constructed Email to Domain Contact

By sending a Random Value (not Request Token) in an Email to the constructed recipient address(es)

#### 8.15.1.4 Email to Applicant

Verifying control over an email address by sending a Random Value in an email to the email address(es) contained in the certificate request CSR.

#### 8.15.1.5 DNS Change

By using only Random Values (no Request Token)

- DNS entry types of TXT

The CA warns the applicant if the domain name validation is about to expire

- For EV TLS certificates: 13 months after successful domain validation
- For DV and OV SSL/TLS certificates: 24 months after successful domain validation
- For all other certificates (Email, Code Signing): according to CPS

### 8.15.1.6 Agreed-Upon Change to Website v2

- Random Value will be used.
- HTTP and HTTPS may be used.
- Validation path: `./well-known/pki-validation/<filename>[.suffix]`

### 8.15.1.7 Implementation

Support for both dns-01 and http-01 challenge verification

- http-01
  - Generate a unique file name containing the challenge. The generated file is sent via email to the RAO, the list of technical contacts and constructed postmaster emails of the Client in the Realm
  - The email contains the instructions and the attachment with the exact file name and content (challenge)
- dns-01
  - TXT is used for the Challenge to check
    - sent via email to the RAO AND to the list of technical contacts and constructed postmaster emails of the Client in the Realm.
- We support email box validation for email certificates: send a confirmation email to the end user's email box for him/her to validate the email -> verification that the user has control over the mail box (outside BR)

### 8.15.1.8 Agreed-Upon Change to Website (ACME)

- Token to be used according to RFC 8555 Section 8.2
- Only HTTP (not HTTPS) to be used.
- Validation path: `./well-known/acme-challenge/<token>` and/or dns-01

## 8.15.2 DNS Tree traversal

Section 4 of the RFC 6844 defines the DNS tree traversal mechanisms to be applied to detect the correct CAA Resource Record.

The RFC 6844 has an error in the definition on how to process CNAME and DNAME alias entries, resulting in resolving the CAA Resource Record of the CNAME target's host domain. (If the CNAME refers to a Google cloud service like Google App Engine, then google.com's CAA Resource Record would have to be resolved and be checked. And Google will not authorize the same CAs as a requester would like to do in his own CAA Records.)

To avoid this, the Errata 5056 corrects this situation. (The content of Errata 5056 is also available in BRG's Appendix A.)

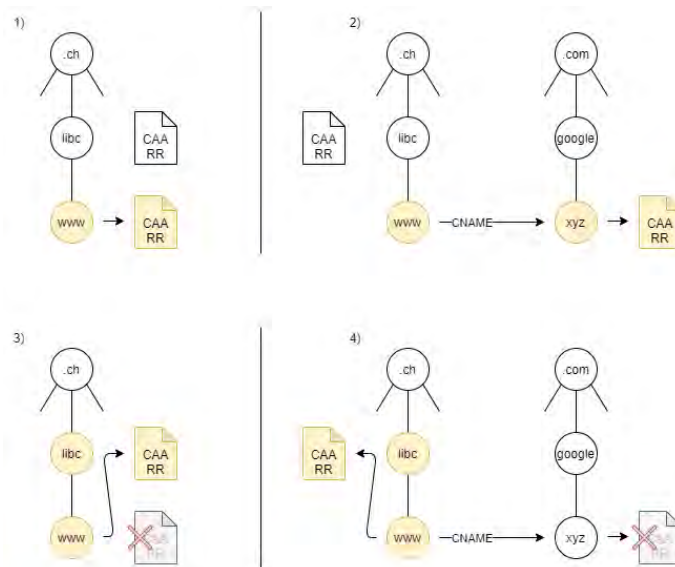
RFC 6844 and Errata 5056 use a set of variables to define the tree traversal (tree climbing) algorithm

Variable	Description	Example
<b>X</b>	Label	<b>www.libc.ch</b>
<b>P(X)</b>	Parent Label of Label X	<b>www.libc.ch</b> (for X = www) <b>www.libc.ch</b> (for X = libc)
<b>CAA(X)</b>	CAA Resource Record set of X	
<b>R(X)</b>	Relevant Record set to be returned.	
<b>A(X)</b>	Target of a CNAME or DNAME alias record for X	<b>www.libc.ch</b> CNAME <b>host.cloudservice.com</b>

The following rules apply:

- 1) If the search for a CAA Record of X directly returns a record, then R(X) is directly CAA(X), and the resolution is done.
- 2) Otherwise, if X is a CNAME or DNAME alias entry A(X), then R(X) is the CAA Resource Record of the Alias entry CAA(A(X)).
- 3) 4) Otherwise if X is not a top-level domain, then try to resolve a CAA Resource Record on the parent domain level P(X).
- If none of the above matches, consider there is no CAA Resource Record available.

### 8.15.2.1 Diagram



### 8.15.2.2 Tree Traversal in SwissPKI

**Step 1:** CA checks the CAA RRs for the domain name on the certificate request—my.blog.example.com.

If the CA finds a CAA record for the domain on the certificate request, the search stops. The CA checks to see if there is a CAA record that authorizes them to issue your certificate. If they find the record, the CA issues the certificate. If they do not find the record, the CA cannot issue the certificate.

If the CA does not find a CAA record for the domain on the certificate request, the CAA record search continues.

**Step 2:** CA checks the CAA RRs for the CNAME target domain—my.blog.example.net.

If the CA finds a CAA record for the CNAME target domain, the search stops. The CA checks to see if there is a CAA record that authorizes them to issue your certificate. If they find the record, the CA issues the certificate. If they do not find the record, the CA cannot issue the certificate.

If the CA does not find a CAA record for the CNAME target domain, the CAA record search continues.

**Step 3:** CA checks the CAA RRs for the original domain's parent domain—blog.example.com.

If the CA finds a CAA record for the original domain's parent domain, the search stops. The CA checks to see if there is a CAA record that authorizes them to issue your certificate. If they find the record, the CA issues the certificate. If they do not find the record, the CA cannot issue the certificate.



If the CA does not find a CAA record for the original domain's parent domain, the CAA record search continues.

**Step 4:** CA checks the CAA RRs for the original domain's base domain—example.com.

If the CA finds a CAA record for the original domain's base domain, the search stops. The CA checks to see if there is a CAA record that authorizes them to issue your certificate. If they find the record, the CA issues the certificate. If they do not find the record, the CA cannot issue the certificate.

If the CA does not find a CAA record for the original domain's base domain, the CAA record search continues.

**Step 5:** CA checks the CAA RRs for the original domain's top-level domain—com.

If the CA finds a CAA record for the original domain's top-level domain, the search stops. The CA checks to see if there is a CAA record that authorizes them to issue your certificate. If they find the record, the CA issues the certificate. If they do not find the record, the CA cannot issue the certificate.

If the CA does not find a CAA record for the original domain's top-level domain, the CA issues the certificate.

### 8.15.3 CAA Resource record processing

The CA processes every domain name present in the issue request's SAN extension or Subject DN. Only if the result of all these checks is the permission to issue, the CA is issuing the certificate.

In case of failures of a CAA check that is based on a failure outside of the CA's infrastructure, the CA is permitted to issue.

If no CAA Resource Record can be found, the CA is allowed to issue. If the CAA Resource Record for the given domain has an invalid structure, the CA may consider this as a failure outside of the CA's infrastructure and is permitted to issue.

If the CAA Resource Record contains unknown properties marked as critical, the CA does not issue the certificate.

When these base checks pass, the CA must try to find an "issue" or "issuwild" property value that explicitly names the issuer domain name of the CA.

If a violation to the policy set defined in the CAA Resource Record is detected, the CA reports the incident (IODEF settings in the CCA Check)

## 9 Initializing SwissPKI

Initializing SwissPKI is the very first step you must execute to configure the first administrator and create the database schema. You will need to provide following information during the initialization process:

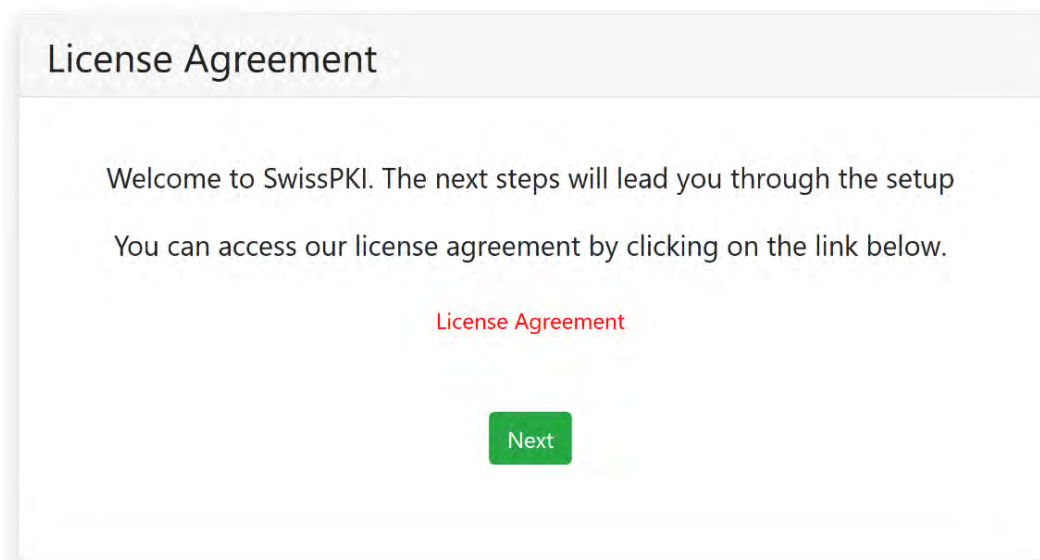
1. SMTP server, port (TLS), user and user password
2. An administrator username (minimum six characters), email and user PIN (At least 8 characters, at least 1 capital letter, at least 1 number and at 1 special character)
3. A QR Code reader (FreeOTP or Google Authenticator)

Access the Administrator UI on the deployed URL **Error! Hyperlink reference not valid.** IP or DNS>/admin

Initialization must occur with username/password and TOTP enabled. Once performed, you can switch to another authentication mechanism.

### 9.1 Step 1 - License Agreement

Click on the “license agreement” link to read them. To accept and continue through the initialization, click on the next button.



## 9.2 Step 2 - SMTP Server

The SMTP Server is used to send Email messages to PKI Administrators.

Fields	Description
<b>Host</b>	The SMTP Server Host
<b>Port</b>	The SMTP Server Port
<b>TLS</b>	TLS (recommended)
<b>From</b>	The email sender
<b>Login User</b>	The SMTP login user
<b>Password</b>	The SMTP login user PIN

### SMTP Server

**Host\***

**Port\***

Use TLS

**From\***

**Login user\***

**Login password\***

### 9.3 Step 3 - System Administrator

Enter the System Administrator details:

Fields	Description
<b>Username</b>	The PKI administrator's username (at least six characters, no space)
<b>Email</b>	The PKI administrator's email address
<b>First Name</b>	The PKI administrator's first name
<b>Last Name</b>	The PKI administrator's last name
<b>Title</b>	The PKI administrator's title
<b>Language</b>	The PKI administrator's language preference
<b>Password</b>	The PKI administrator login PIN (at least eight characters at least 1 capital letter, at least 1 number and at least 1 special character)
<b>Password (repeat)</b>	Repeat the password entered in the password field

#### Administrator

**User name\***

**Email\***

**First name**

**Last name**

**Title**


**Language**

**Password\***


**Password (repeat)\***

You will need an authenticator for the second factor

Authenticator for Android



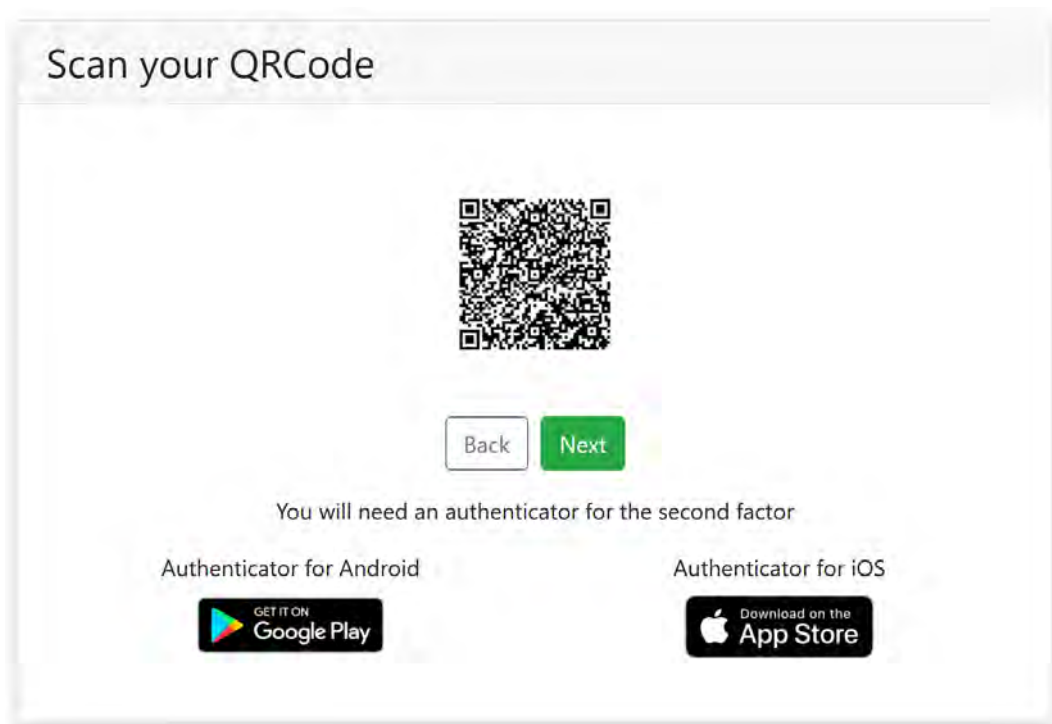
Authenticator for iOS



## 9.4 Step 4 - QR Code

The first initialization uses a two-factor authentication for a secure login method (for other authentication methods, please see section 8.3 End User Login Options).

Scan the QR code. If you do not have an app allowing you to do it, you can download one with the provided links.



## 9.5 Step 5 - Review

Review the information you entered in the previous steps and copy the scratch codes and keep them in a safe place. You may need them to login if you do not have access to your authenticator application.

Clicking 'next' will initialize SwissPKI and redirect you to the Administrator UI login page.

### Review

---

**Mail server** Edit

Host and port *mail.infomaniak.com :587, tls: true*

Sender *demo@swisspki.com*

SMTP login *demo@swisspki.com*

---

**Administrator** Edit

Title *MR*

User name *pki.admin*

Email *pki.admin@gmail.com*

Scratch codes

60997560

92325810

29480017

70521389

32602178

56072514

75815681

75599954

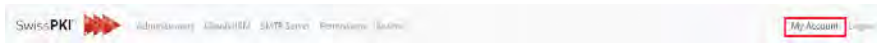
56601019

48878391

Back
Next

## 10 Account

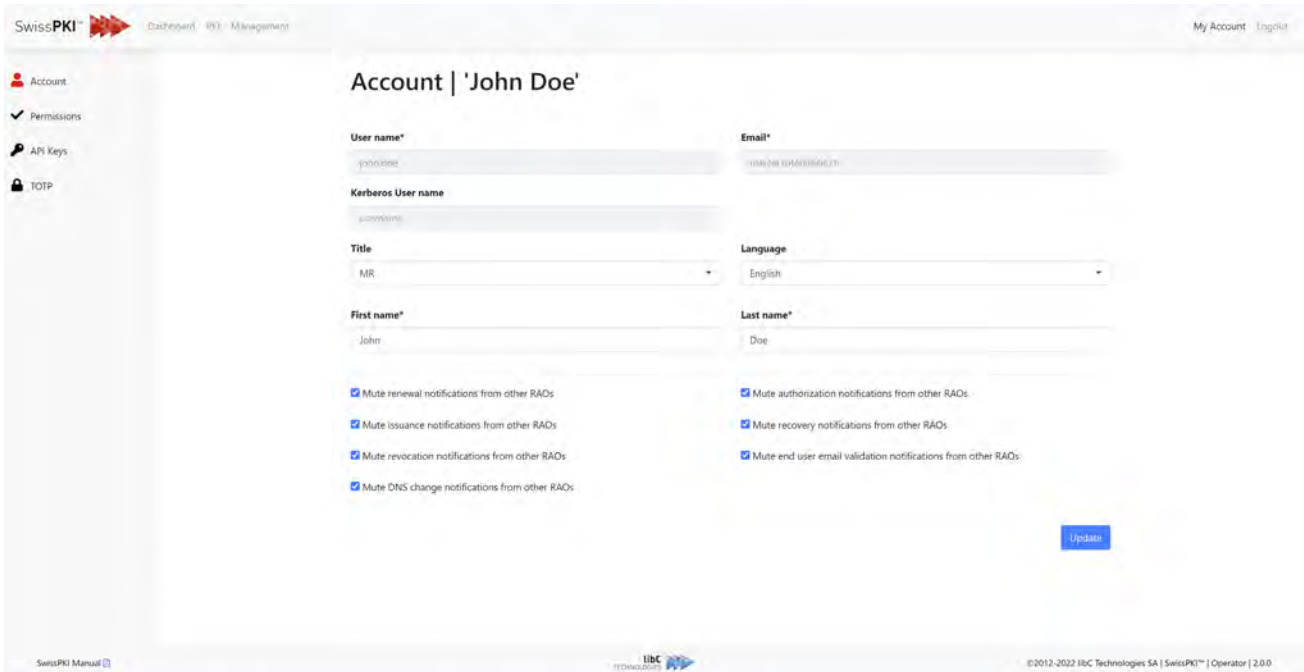
The account page, available to all users via the ‘My Account’ menu option in the Administration, Operator and RA WebUI, allows you to manage your user’s details.



### 10.1 Account details

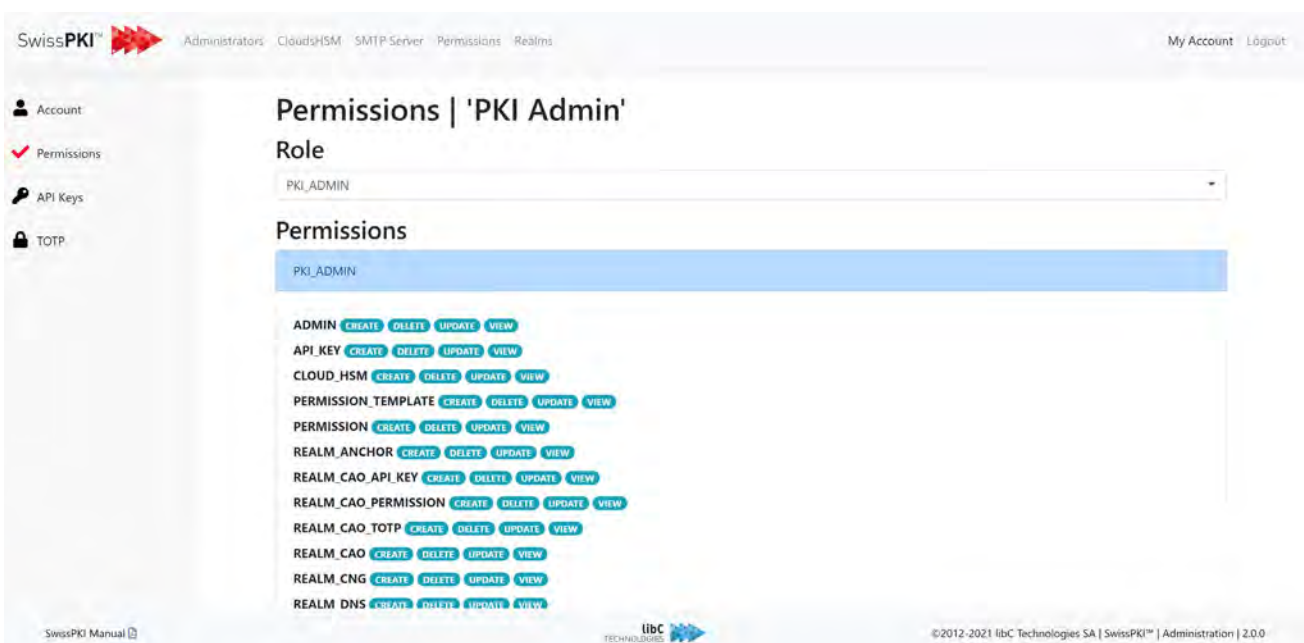
This page allows you to review and modify your account details. The following fields are available:

Fields	Description
<b>Username (not modifiable)</b>	Your username
<b>Email (not modifiable)</b>	The email associated with your account
<b>LDAP Username (not modifiable)</b>	The LDAP user linked with your account (if enabled)
<b>OIDC Username (not modifiable)</b>	The OpenID user linked with your account (if enabled)
<b>Kerberos Username (not modifiable)</b>	The Kerberos user linked with your account (if enabled)
<b>First name</b>	Your first name
<b>Last name</b>	Your last name
<b>Title</b>	Your title
<b>Language</b>	Your preferred language
<b>Mute notification</b>	<p>You may optionally set notification muting.</p> <p>This option is only available in the Operator UI and RA UI.</p> <p>Refer to <i>12.2.5.1 Notifications and Recipients</i></p>



## 10.2 Account Permissions

The account permissions display the permissions per assigned roles to your user account. Select the roles from the 'Role' drop down to display the assigned permissions. You cannot edit your own permissions/roles.





### 10.3 Account API Keys

From the account API Keys tab, you can create and manage your user's API Keys. API Key management is enabled if permission is granted.

Actions	Description
<b>Add new API Key</b>	Issues a new API Key
<b>Refresh</b>	Issues a new API key and set the active API key in an expiration status. The previous API key is still valid for a 7-day period
<b>Retire</b>	Retires the active API Key and sets the key into an expiration status. The API Key is still valid for a 7-day period.
<b>Delete</b>	Deletes an API Key. Deletion is only possible for 'expiring' key.  To delete an active API key: retire the active API key followed by 'dele' API Key



## 10.4 Account TOTP

The account TOTP page allows you to access your TOTP QR code as well as the scratch codes. You are also able to reset them by clicking on the reset button.

**Note:** This page is enabled when username/password with TOTP is enabled.



The screenshot shows the SwissPKI administration interface. The top navigation bar includes 'SwissPKI', 'Administrators', 'CloudHSM', 'SMTP Server', 'Permissions', 'Issuance', 'My Account', and 'Logout'. The left sidebar has a menu with 'Account', 'Permissions', 'API Keys', and 'TOTP'. The main content area is titled 'TOTP | 'PKI Admin'' and contains a 'QR Code' section with a QR code and a 'Scratch Codes' section with ten green buttons displaying the following numbers:

60997560	92325810	29480017	70521389	32602178
56072514	75815681	75599954	56601019	48876391

Below the scratch codes is a blue 'Reset' button. The footer contains 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0'.

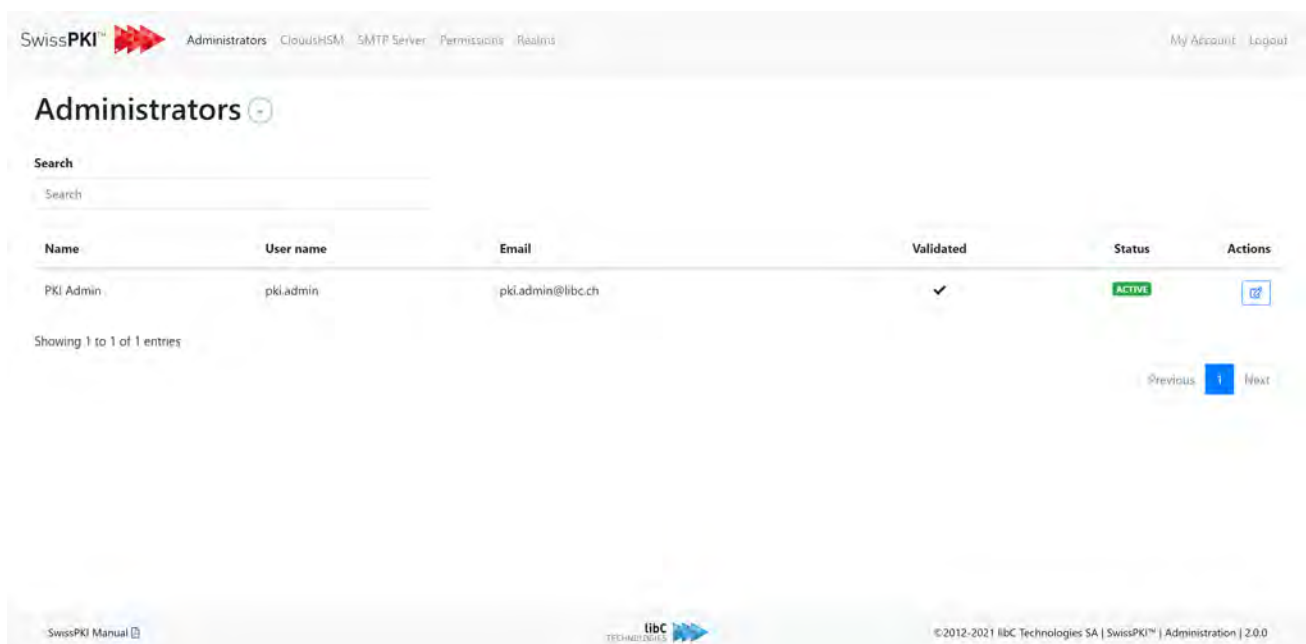
## 11 Administrator UI

The Administration UI is accessible at the deployed URL **Error! Hyperlink reference not valid.** or `DNS>/admin` to registered PKI Administrator roles. As a PKI Administrator, you can:

- Manage other PKI Administrators
- Configure CloudHSM proxy configuration
- Update the administration SMTP server connection
- Manage Permission Templates for PKI Administrator and CA Operator roles
- Manage blacklists
- Manage Realms

### 11.1 PKI Administrators

The *'Administrators'* menu tab displays the list of all PKI Administrators. You can add new PKI Administrators by clicking on the *'add'* button located on right of the page title. To access detailed information about PKI Administrator, click on *'edit'* button on the far right of the table in the action's column.




SwissPKI™ Administrators CloudHSM SMTP Server Permissions Realms My Account Logout

### Administrators


Search


Search

Name	User name	Email	Validated	Status	Actions
PKI Admin	pki.admin	pki.admin@libc.ch	✓	ACTIVE	

Showing 1 to 1 of 1 entries

Previous 1 Next

SwissPKI Manual 

libC TECHNOLOGIES 

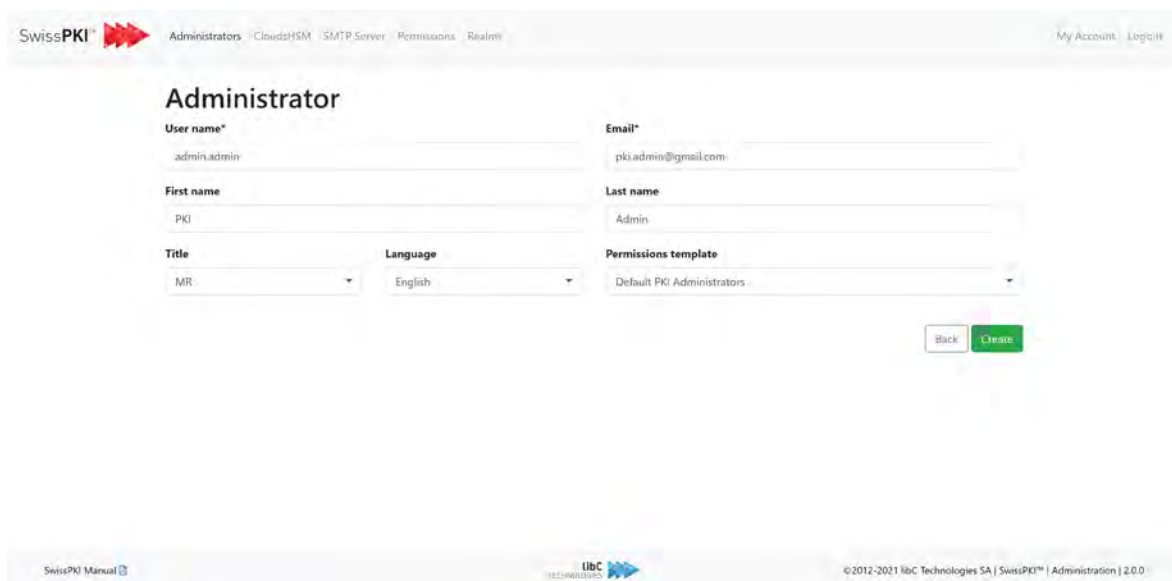
©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0

### 11.1.1 Creating PKI Administrators

When you create a new PKI Administrator, a confirmation Email is sent to the new user with its TOTP information. Additionally, the new user will have to confirm its Email address and set its password prior to login.

Provide the following information to create a new PKI Administrator:

Fields	Description
<b>Username</b>	The system administrator's username (must be unique). <ol style="list-style-type: none"> <li>At least six characters</li> <li>Cannot contain spaces</li> </ol>
<b>Email</b>	The PKI Administrator's email address
<b>First Name</b>	The PKI Administrator's first name
<b>Last Name</b>	The PKI Administrator's last name
<b>Title</b>	The PKI Administrator's title
<b>Language</b>	The PKI Administrator's preferred language
<b>Permission Template</b>	Assign a PKI Administrator 'Permission Template.' For detailed information about 'Permission Templates,' please refer to 11.4 Permission



The screenshot shows the 'Administrator' creation form in the SwissPKI interface. The form includes the following fields:

- User name\***: admin.admin
- Email\***: pki.admin@gmail.com
- First name**: PKI
- Last name**: Admin
- Title**: MR
- Language**: English
- Permissions template**: Default PKI Administrators

Buttons for 'Back' and 'Create' are visible at the bottom right of the form.

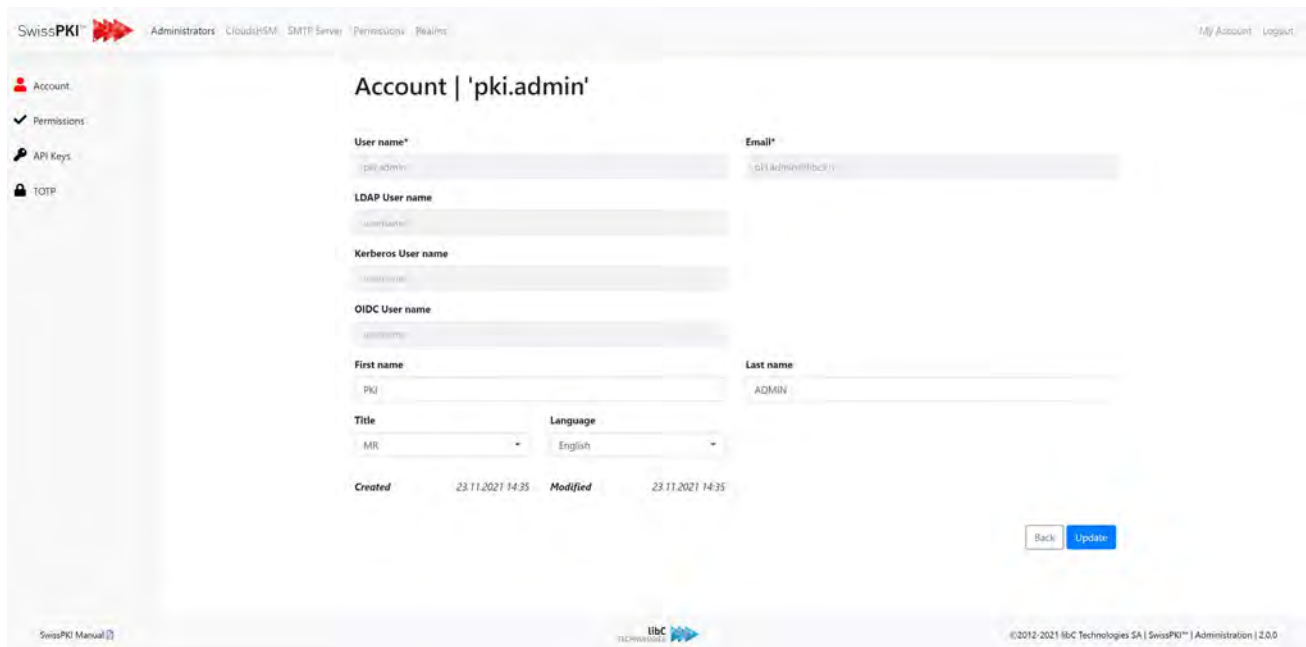
## 11.1.2 Editing PKI Administrators


Editing PKI Administrators lets you accomplish the following functions:

- Edit the PKI Administrator account information
- Edit the PKI Administrator permissions
- Reset the PKI Administrator's TOTP

### 11.1.2.1 Account

Edit the user's information (see 11.1.1 Creating PKI Administrators for filed values).



SWISSPKI  Administrators CloudHSM SMTP Server Permissions Realms My Account Logout

**Account | 'pki.admin'**

**User name\***  **Email\***

**LDAP User name**

**Kerberos User name**


**OIDC User name**

**First name**  **Last name**

**Title**  **Language**

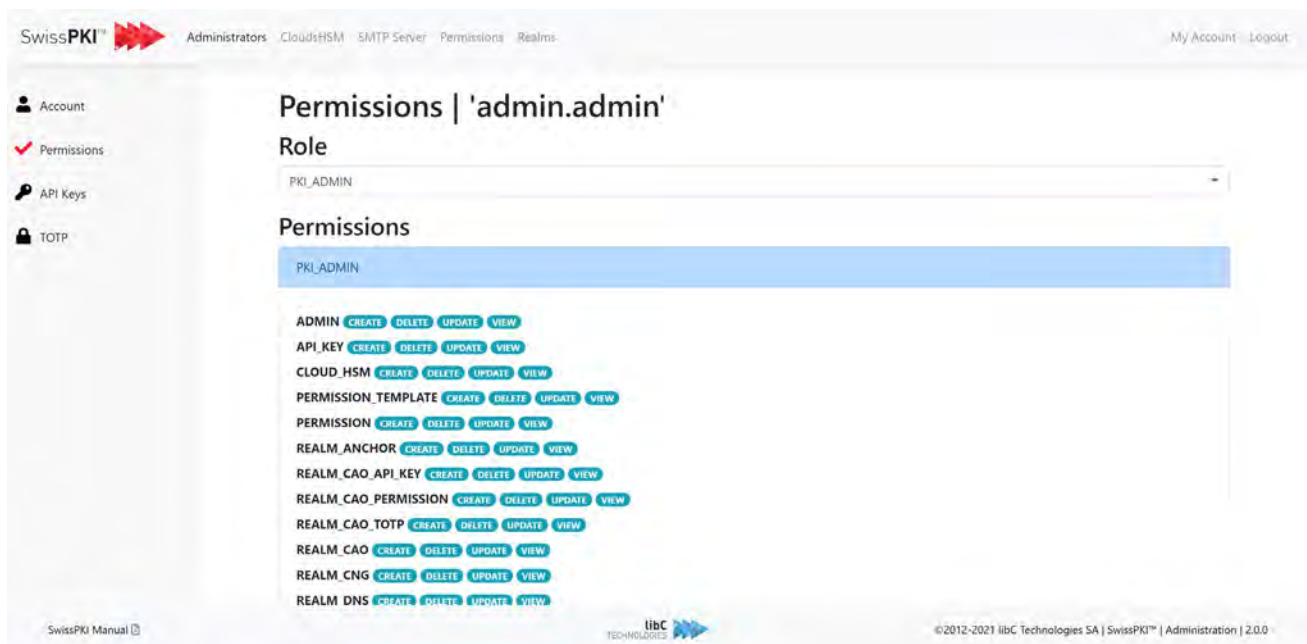
Created	Modified
23.11.2021 14:35	23.11.2021 14:35

[Back](#) [Update](#)

SwissPKI Manual  ©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0

### 11.1.2.2 Permissions

View and/or reset the PKI Administrator's Permission Template. Note that you cannot modify your own 'Permission Template.'



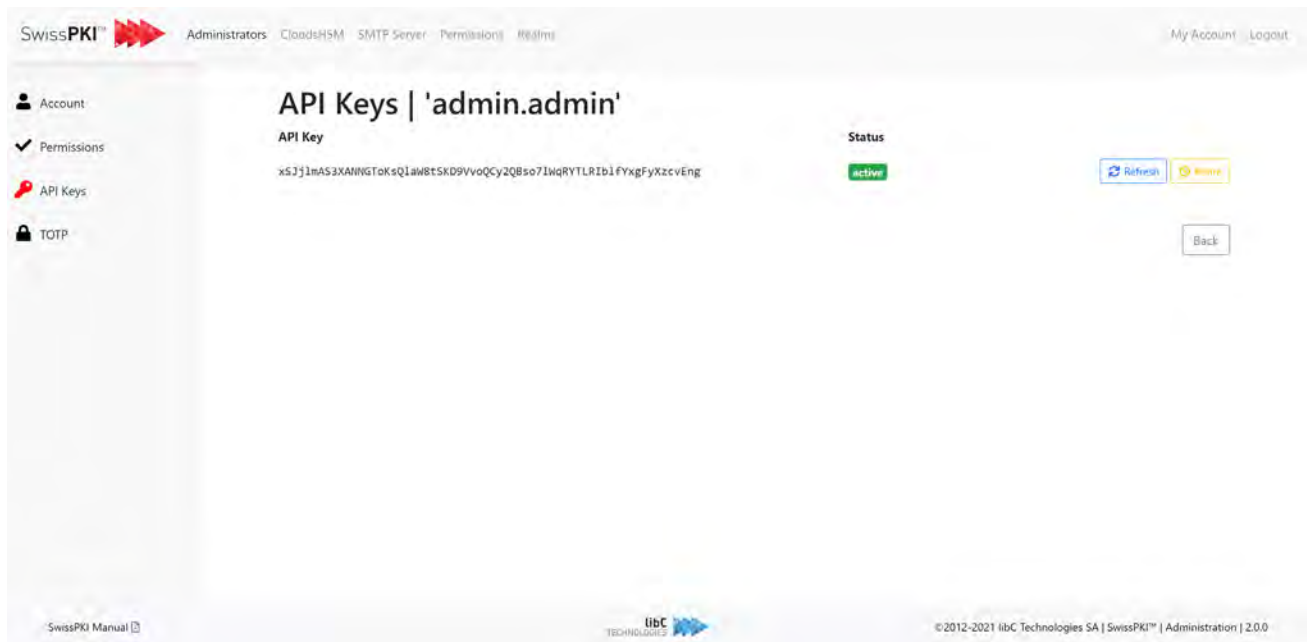
The screenshot shows the SwissPKI web interface. The breadcrumb navigation includes 'Administrators', 'CloudsHSM', 'SMTP Server', 'Permissions', and 'Realms'. The main heading is 'Permissions | admin.admin'. Below this, a dropdown menu shows the selected role 'PKI\_ADMIN'. A table lists various permissions with their respective actions:

Permission	Actions
ADMIN	CREATE, DELETE, UPDATE, VIEW
API_KEY	CREATE, DELETE, UPDATE, VIEW
CLOUD_HSM	CREATE, DELETE, UPDATE, VIEW
PERMISSION_TEMPLATE	CREATE, DELETE, UPDATE, VIEW
PERMISSION	CREATE, DELETE, UPDATE, VIEW
REALM_ANCHOR	CREATE, DELETE, UPDATE, VIEW
REALM_CAO_API_KEY	CREATE, DELETE, UPDATE, VIEW
REALM_CAO_PERMISSION	CREATE, DELETE, UPDATE, VIEW
REALM_CAO_TOTP	CREATE, DELETE, UPDATE, VIEW
REALM_CAO	CREATE, DELETE, UPDATE, VIEW
REALM_CNG	CREATE, DELETE, UPDATE, VIEW
REALM_DNS	CREATE, DELETE, UPDATE, VIEW

At the bottom of the page, there is a footer with 'SwissPKI Manual', the libC Technologies logo, and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0'.

### 11.1.2.3 API Keys

If you have the permission enabled to manage API Keys, you can generate and/or reset PKI Administrator 'API Keys' through the "API Keys" tab. Resetting (deleting) an API Key will immediately disable access via OpenAPI for the selected PKI Administrator.



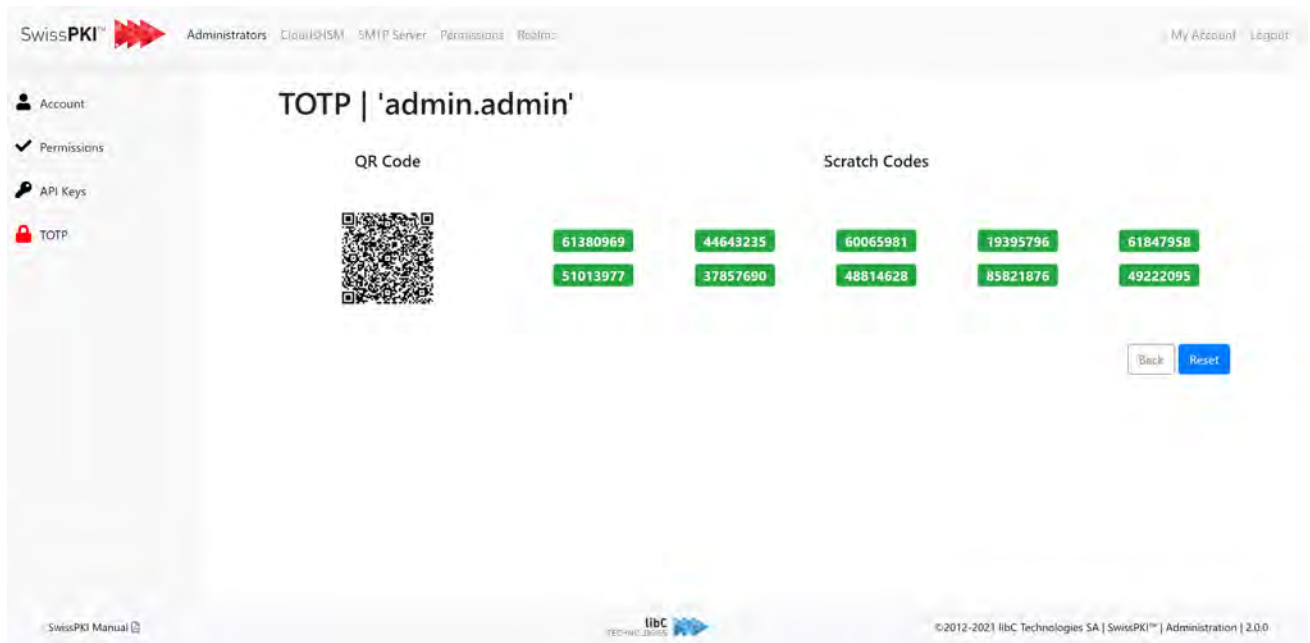
The screenshot shows the SwissPKI web interface. The top navigation bar includes 'SwissPKI' and a menu with 'Administrators', 'CloudsHSM', 'SMTP Server', 'Permissions', and 'Realms'. The user is logged in as 'admin.admin'. The left sidebar has 'Account', 'Permissions', 'API Keys', and 'TOTP'. The main content area is titled 'API Keys | 'admin.admin'' and shows a table with the following data:

API Key	Status
xSj1mAS3XANNGToksQ1aW8tSKD9VvoQCy2QBso71nqRYTLR1b1FYxgFyXzcVEng	active

Buttons for 'Refresh', 'More', and 'Back' are visible next to the table entry. The footer contains 'SwissPKI Manual', the libC Technologies logo, and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0'.

### 11.1.2.4 TOTP

View or reset the PKI Administrator's QR code as well as the scratch codes. Resetting the QR Code will send an Email to the user with the new values.



The screenshot shows the SwissPKI Administration interface. The top navigation bar includes 'Administrators', 'CloudHSM', 'SMTP Server', 'Permissions', and 'Realm'. The user is logged in as 'My Account' and can 'Logout'. The left sidebar shows navigation options: 'Account', 'Permissions', 'API Keys', and 'TOTP'. The main content area is titled 'TOTP | 'admin.admin'' and displays a 'QR Code' and 'Scratch Codes'. The QR code is a standard black and white square. The scratch codes are displayed in a grid of green boxes with white text. Below the scratch codes are 'Back' and 'Reset' buttons. The footer contains the 'SwissPKI Manual' link, the libC Technologies logo, and the copyright notice: '©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0'.

Scratch Codes
61380969
44643235
60065981
19395796
61847958
51013977
37857690
48814628
85821876
49222095

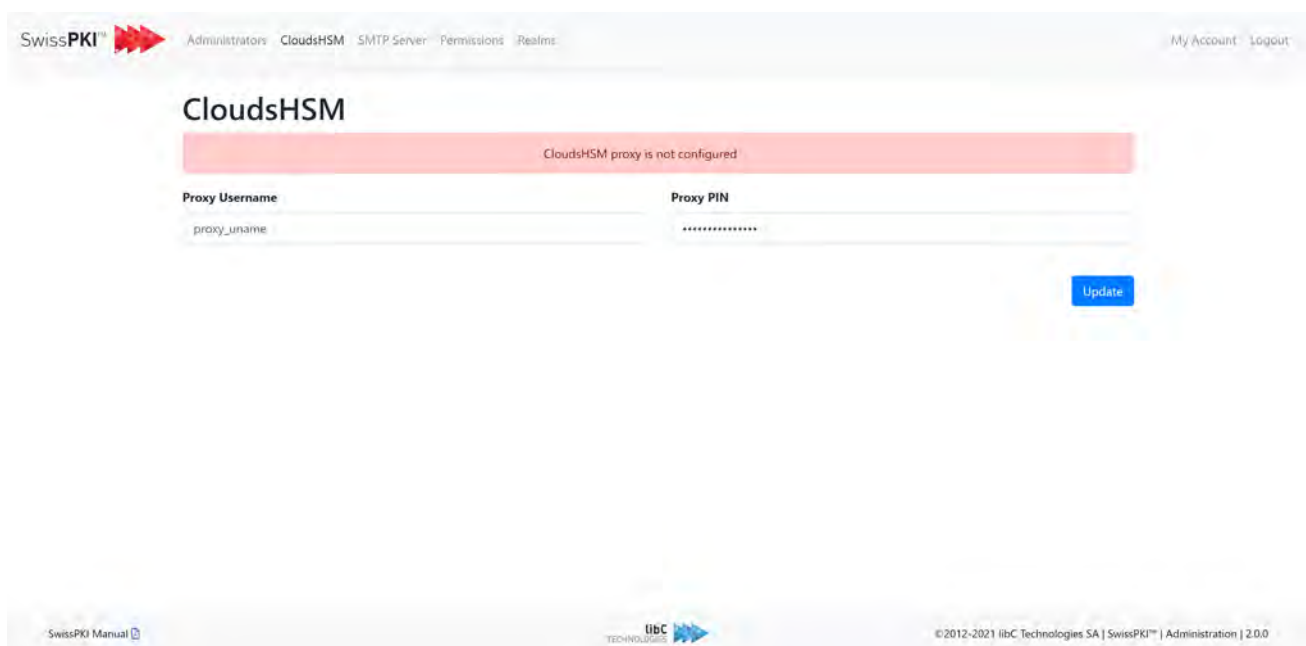


## 11.2 CloudHSM

This tab allows you to set up the CloudsHSM. To do so, the proxy username and PIN for the connection to the HSM Cloud proxy are required. To enable specific partitions to connect through the CloudHSM, you need to enable the proxy connection on the selected HSM partition. For more details, please refer to section 12.2.7 *HSMs*.

Fields	Description
Proxy Username	CloudHSM Proxy Username
Proxy Pin	CloudHSM Proxy PIN

Settings when the CloudHSM is not configured



**Note:** CloudsHSM proxy configuration is only applicable to Primus HSMs <sup>11</sup>.

<sup>11</sup> Requires a Primus CloudsHSM account

Settings when the CloudHSM is configured (the PIN is not displayed)

## CloudsHSM

CloudsHSM proxy is configured

Proxy Username\*

cloud.user

Proxy PIN\*

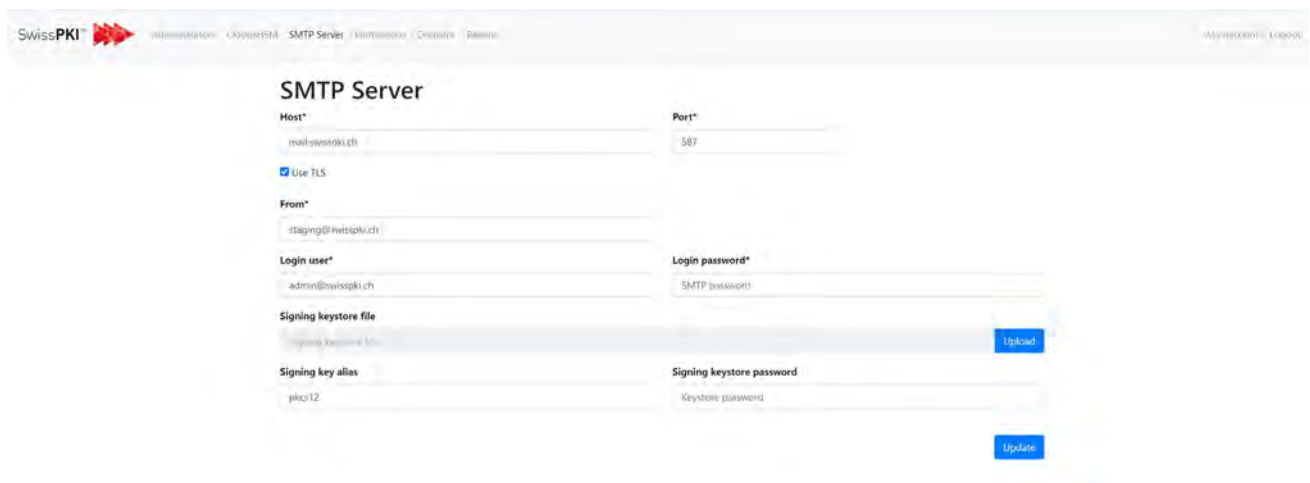
proxy password

Update

### 11.3 SMTP Server

This tab allows you to update the SMTP server details you entered during the initialization. Changes are saved after you click on the update button. This SMTP connection is used to send notifications to PKI Administrators. For Realm SMTP connections, please refer to section *11.5.4 Realm SMTP Server*

Fields	Description
<b>Host</b>	The SMTP Server Host
<b>Port</b>	The SMTP Server Port
<b>TLS</b>	Activate or not the use of TLS
<b>From</b>	The email sender
<b>Login User</b>	The user used to log into your SMTP Server
<b>Password</b>	The Login User's Password
<b>Signing keystore file</b>	An optional PKCS#12 S/MIME certificate including full certificate chain and private key
<b>Signing key alias</b>	The private key alias to use
<b>Signing keystore password</b>	The PKCS#12 password to unlock the signing private key



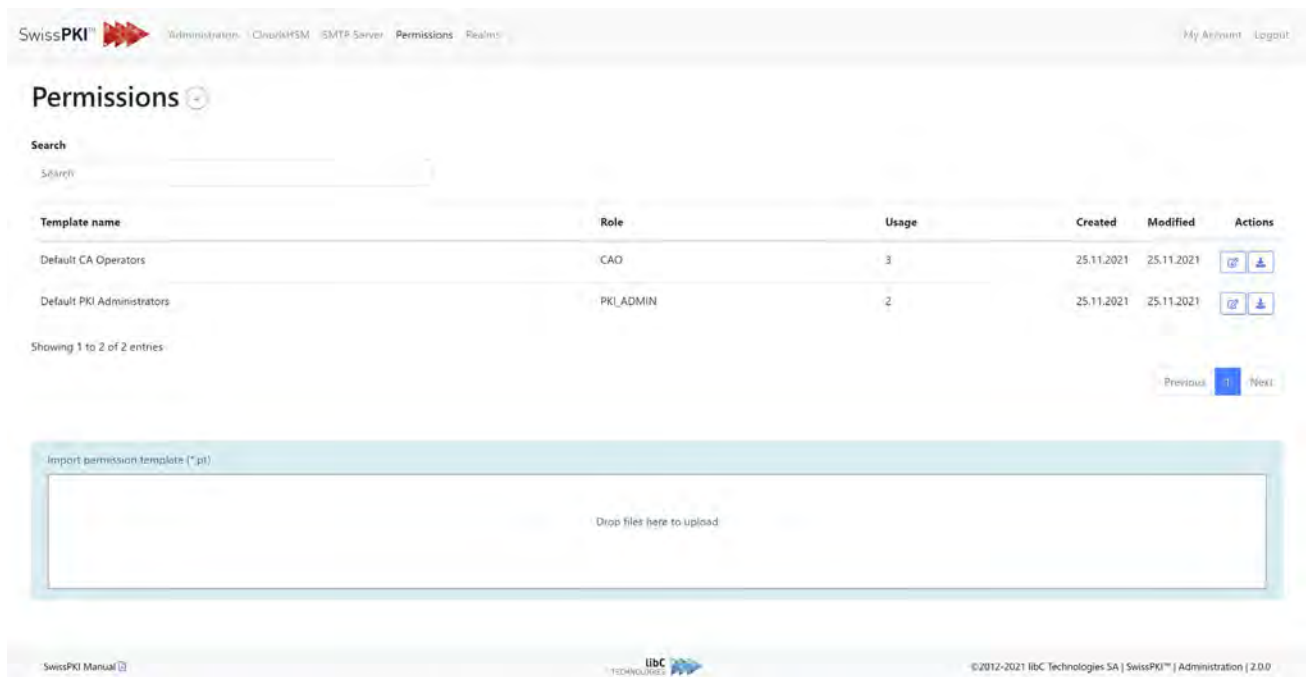
The screenshot shows the 'SMTP Server' configuration page. The fields and their values are as follows:

- Host\***: mail-swisspki.ch
- Port\***: 587
- Use TLS**:
- From\***: staging@swisspki.ch
- Login user\***: admin@swisspki.ch
- Login password\***: SMTP (password)
- Signing keystore file**: Signing keystore file (with an 'Upload' button)
- Signing key alias**: pkcs12
- Signing keystore password**: Keystore password (with an 'Update' button)

## 11.4 Permission Templates

On initial setup, two permission templates ‘**Default CAO Operators**’ and ‘**Default PKI Administrators**’ are generated. The initial PKI Administrator created during initialization has the permission template ‘All PKI Admins’ associated to its user account. To modify the initial PKI Administrator’s permissions, create a new PKI Administrator. The new PKI Administrator with a new permission template with the permission to modify other administrators’ permission settings.

- To create a new permission template, click on ‘+’ icon
- To export an existing permission template, click on the ‘download button
- To edit an existing permission template, click on the ‘edit’ button
- To delete an existing permission template, click on the ‘delete’ button
- To import a permission template, drag & drop an exported policy template file. You can only import permission templates with role PKI\_ADMIN or CAO



The screenshot shows the 'Permissions' section of the SwissPKI Administration interface. At the top, there is a navigation bar with 'SwissPKI' and menu items: Administration, Client/MSM, SMTP Server, Permissions, and Realtime. On the right, there are links for 'My Account' and 'Logout'. Below the navigation bar, the 'Permissions' title is followed by a search bar. A table lists two permission templates:

Template name	Role	Usage	Created	Modified	Actions
Default CA Operators	CAO	3	25.11.2021	25.11.2021	[edit] [download]
Default PKI Administrators	PKI_ADMIN	2	25.11.2021	25.11.2021	[edit] [download]

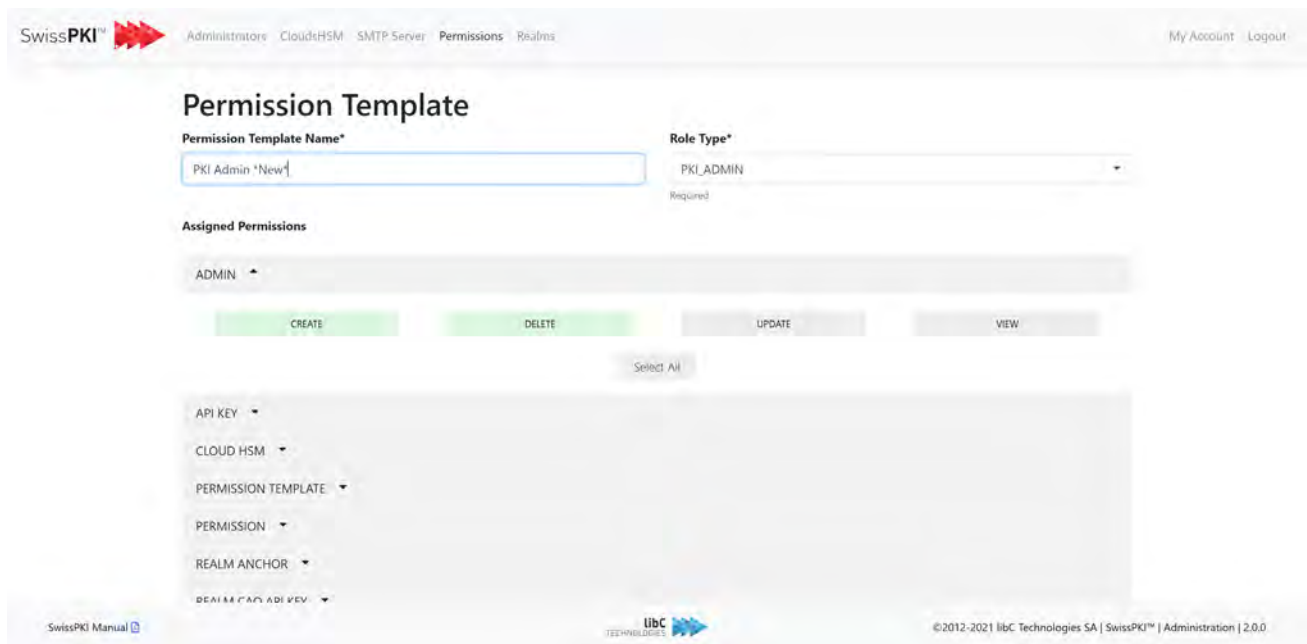
Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom right of the table area, there are 'Previous' and 'Next' navigation buttons. Below the table is a large light blue box for importing templates, titled 'Import permission template (\*.p1)'. Inside this box, there is a text prompt: 'Drop files here to upload.' At the very bottom of the page, there is a footer with 'SwissPKI Manual', the libC TECHNOLOGIES logo, and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Administration | 2.0.0'.

**Note:** Please carefully read section 8.3 *End User Login Options* if you plan to rename the default permission templates.

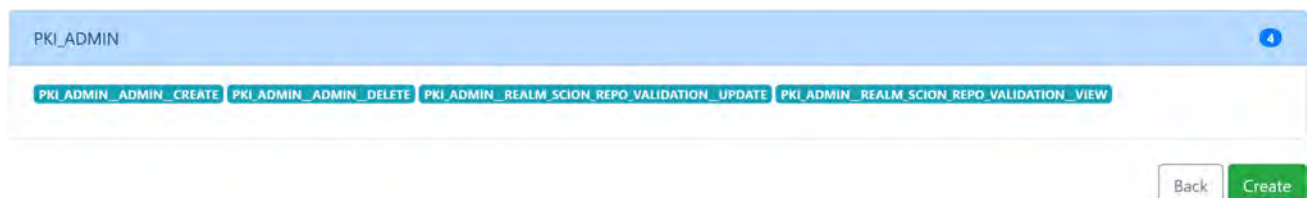
### 11.4.1 Creating Permission Templates

As a PKI Administrator, you have the possibility to create permission templates for either PKI Administrator or CA Operator roles.

Field	Description
<b>Permission Template Name</b>	The name of the permission template
<b>Role Permissions Type</b>	List of available roles (PKI Administrator or CA Operator)  For permission details, please refer to 7.5.2 <i>Permissions</i>



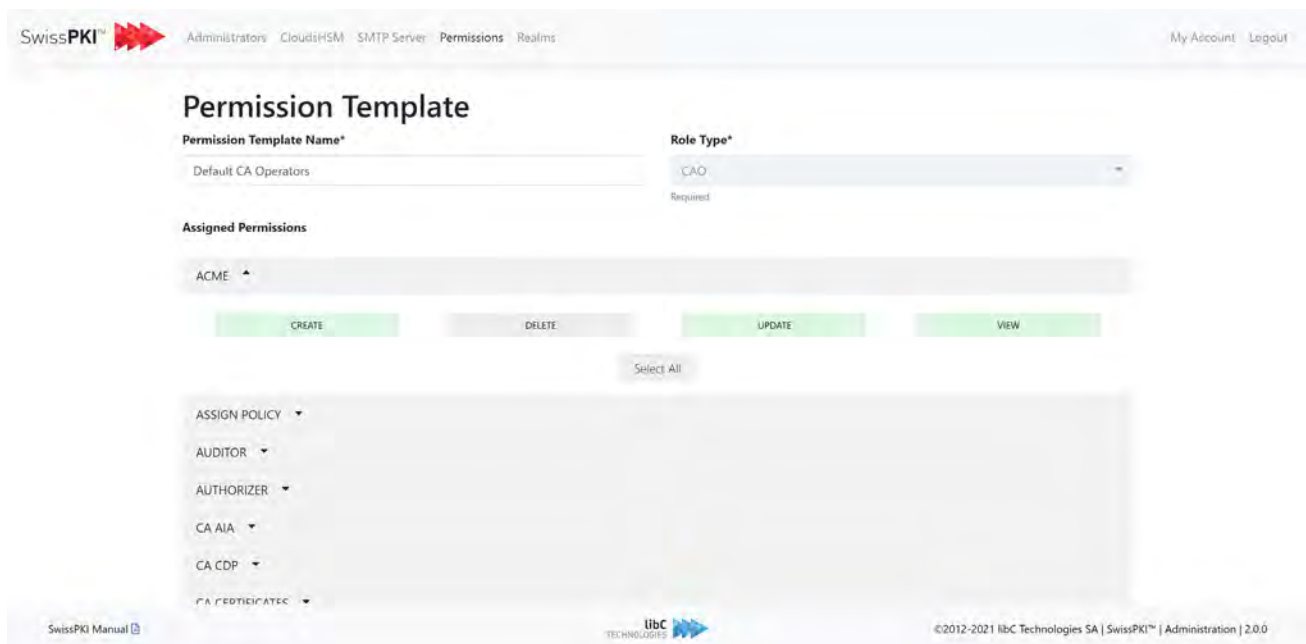
Selecting the permissions, you plan to grant to the permission template. Adding permissions will display at the bottom of the permission selections:



Click 'Create' to save the permission template

### 11.4.2 Editing Permission Templates

Select the permission template you wish to modify and select/unselect the permissions to remove/add to the permission template. Click ‘Save’ to save your modifications.



Modifying permission templates updates the permissions of the roles associated with it. If you modify a permission template, end users must login anew for the changes to take effect.

### 11.4.3 Deleting Permission Templates

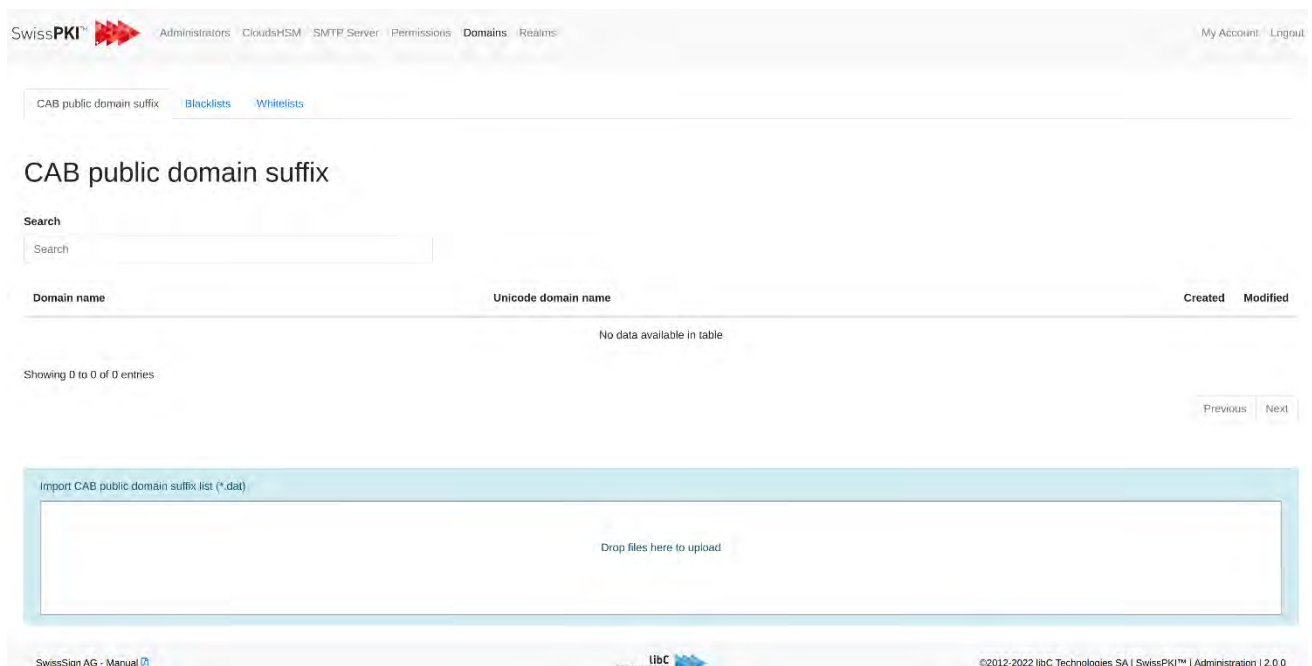
Deleting permission templates is only possible if no user role is assigned to the permission template.

## 11.5 Domains

This ‘Domains’ tab allows you to manage different blacklists.


### 11.5.1 CAB Public Domain Suffix

This tab allows you to upload and update the CAB public domain suffix list. This list can be found under the following link: [https://publicsuffix.org/list/public\\_suffix\\_list.dat](https://publicsuffix.org/list/public_suffix_list.dat). This list must be downloaded manually and uploaded into the app via the drop zone shown below. Once uploaded, the list is displayed in the table. Notice that uploading a new list overwrites the existing list. It is not possible to upload multiple lists.



### 11.5.2 Blacklists

The blacklists tab allows you to import other types of blacklists. Currently there are only 2 available types of blacklists: Embargo and Alexa. To create a new blacklist, provide a name and a type. The newly created list will then appear in the list of blacklists.

SwissPKI™  Administrators CloudsHSM SMTP Server Permissions Domains Realms My Account Logout

[CAB public domain suffix](#) [Blacklists](#) [Whitelists](#)


## Blacklists +

Search

List name	admin.list.type	Created	Modified	Actions
Blacklist1	ALEXA	24.03.2022	24.03.2022	<a href="#">✎</a> <a href="#">🗑</a>
Blacklist2	EMBARGO	24.03.2022	24.03.2022	<a href="#">✎</a> <a href="#">🗑</a>


Showing 1 to 2 of 2 entries

Previous [1](#) Next

SwissSign AG - Manual  libC TECHNOLOGIES ©2012-2022 libC Technologies SA | SwissPKI™ | Administration | 2.0.0

### 11.5.3 Whitelists

The whitelists tab is remarkably like the blacklists tab. The only difference is that there is a single option for the type of the list: Whitelist.

SwissPKI™  Administrators CloudsHSM SMTP Server Permissions Domains Realms My Account Logout

[CAB public domain suffix](#) [Blacklists](#) [Whitelists](#)


## Whitelists +

Search

List name	admin.list.type	Created	Modified	Actions
Whitelist	WHITELIST	24.03.2022	24.03.2022	<a href="#">✎</a> <a href="#">🗑</a>

Showing 1 to 1 of 1 entries

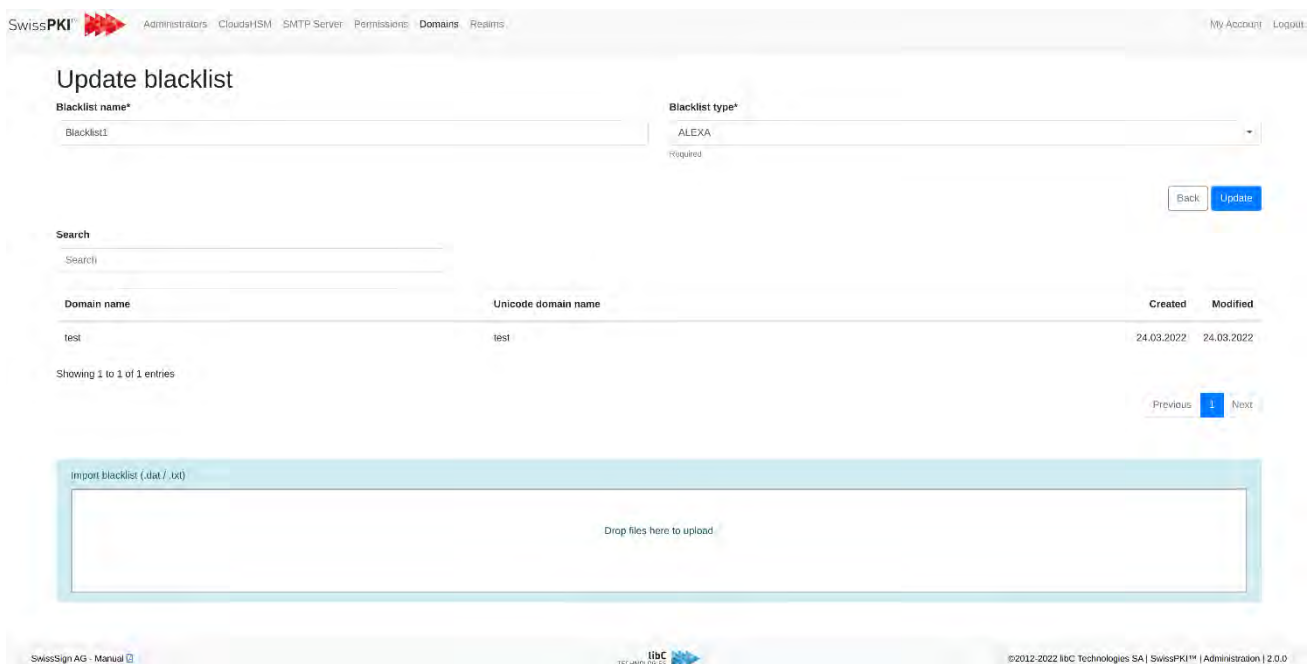
Previous [1](#) Next

SwissSign AG - Manual  libC TECHNOLOGIES ©2012-2022 libC Technologies SA | SwissPKI™ | Administration | 2.0.0



### 11.5.4 Editing black and white lists

To edit a list, click on the edit icon next to the list you wish to modify. From there, you can import a new list via the drop zone. Similarly, to the CAB public domain suffix, uploading a new list overwrites the existing list. The image below shows a screen to edit a blacklist. The same page appears for whitelists with different type options.



### 11.6 Realms

As a PKI Administrator, you manage Realms. A Realm is a tenant and SwissPKI supports multiple Realms (multi-tenant) per deployment. PKIs along with the Certification Authorities, certificates, users, and clients are deployed within Realms. PKIs deployed within a Realm cannot cross their Realm boundary except if you decide to cross sign Certification Authorities between Realms. Additionally, users created within one Realm cannot access PKI entities deployed in another Realm. You need to create separate users in each Realm if you plan to have one ‘physical’ person accessing different PKIs deployed in different Realms.

On Realms tab, you access the list of all deployed Realms. To create a new Realm, click on the ‘+’ link right of the page title. To edit a Realm, click on the ‘edit’ button in the far right of the table. To delete a Realm, click on the ‘delete’ button in the far right of the table. Note that deleting a Realm will mark it as deleted in the database and not effectively drop the records from the database.

## Realms

Search

Created	Modified	Realm	Actions
01.12.2021	01.12.2021	Realm	 

Showing 1 to 1 of 1 entries

Previous:  Next

### 11.6.1 Add Realm

After clicking on the add realm button, provide a name for your realm, and confirm its creation by clicking on the 'create' button.

#### Realm

name\*

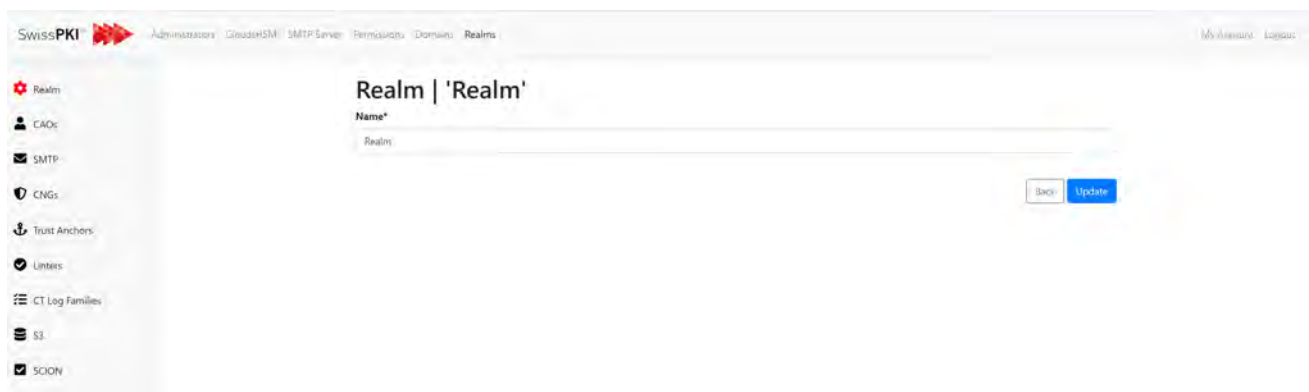
Back

Create

## 11.6.2 Edit Realm

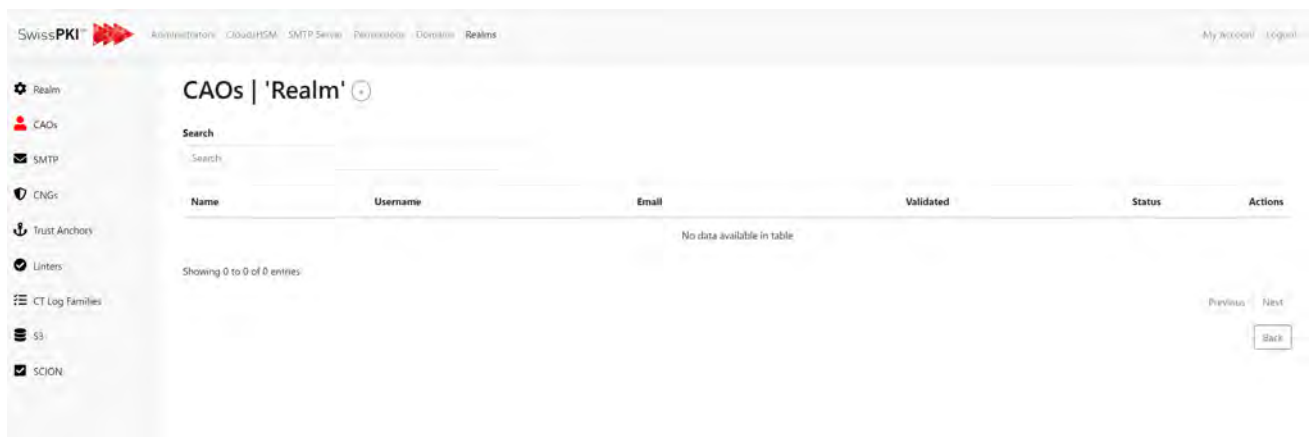
By clicking on a Realm's 'edit' button, you access its configuration. Configuring a Realm allows you to:

- Rename the Realm
- Edit CA Operators
- Edit SMTP server information
- Edit DNS server information
- Edit Microsoft CNG information
- Edit Trust Anchors
- Edit Linters
- Edit CT Log Families
- Edit S3 object store
- Edit SCION Identity Repository Validation Service settings



### 11.6.2.1 CAOs

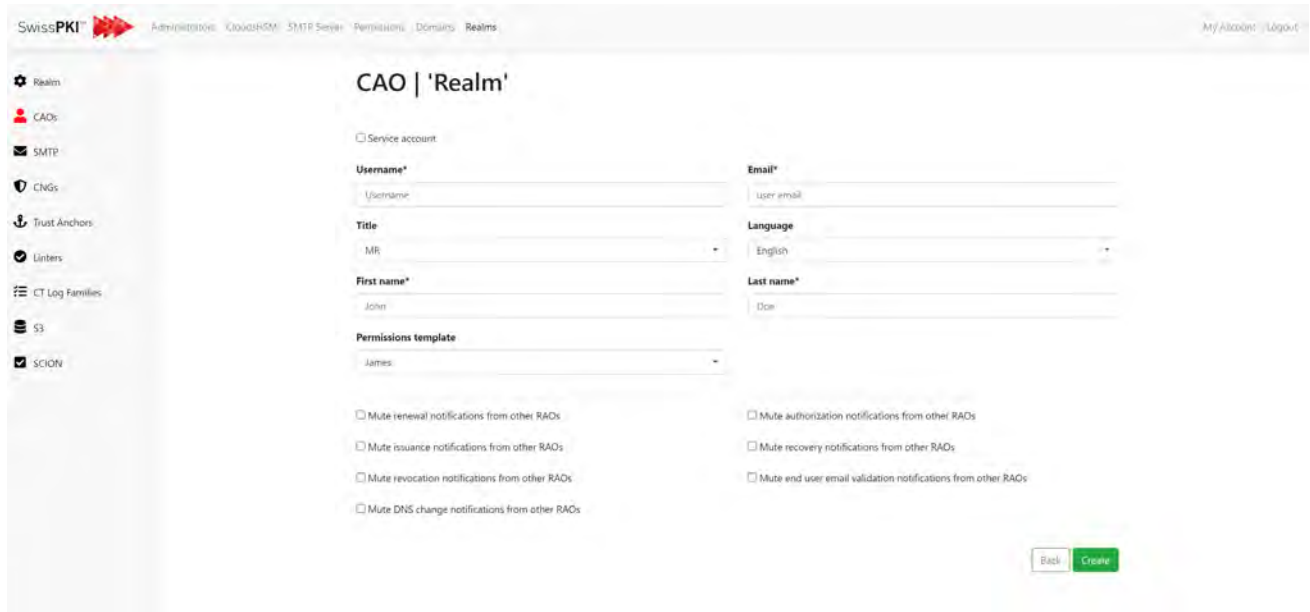
A list of all created CAOs for a realm is found under its CAOs tab. This tab allows you to create new CAOs by clicking on the add button located on the right of the page title. Additionally, you can edit or delete existing CAOs by clicking on the buttons located in the action column of the table. CAOs can only be created and managed by administrators.



### 11.6.2.1.1 Create CAO

To create a new CAO, you need to provide the following information:

Fields	Description
<b>Service account</b>	TBD
<b>User name</b>	<p>The CAO's user name (must be unique).</p> <ul style="list-style-type: none"> <li>At least 8 characters</li> </ul> <p>Cannot contain spaces</p>
<b>Email</b>	The CAO's email
<b>Fist Name</b>	The CAO's first name
<b>Last Name</b>	The CAO's last name
<b>Title</b>	The CAO's title
<b>Language</b>	<p>The language used by the CAO. Three choices are available:</p> <ol style="list-style-type: none"> <li>English</li> <li>French</li> <li>German</li> </ol>
<b>Permission template</b>	The permission template attributed to the CAO. These permissions define the permissions assigned to CAO.
<b>Mute notification</b>	<p>If the CAO is assigned RAO role at a later stage, you may optionally set notification muting.</p> <p>Refer to <i>12.2.5.1 Notifications and Recipients</i></p>



After clicking on the create button, a user registration email is sent to the email you provided <sup>12</sup>. In this email, you will find:

4. A link to confirm the email address.
5. A step-by-step guide on how to configure two-factor authentication for this user.
6. The two-factor authentication's QR Code
7. The two-factor authentication scratch codes.

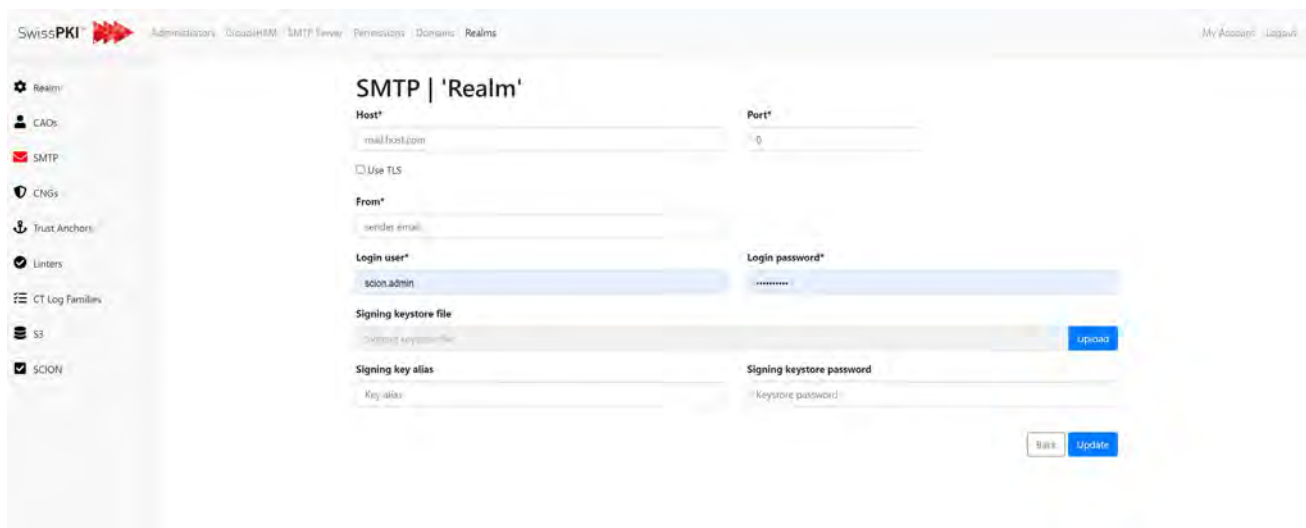
Once the CAO confirmed his email address, he will receive a second email allowing him to configure his password. Then one last email informing that the password was configured is sent.

<sup>12</sup> Notification is sent when Username/Password with TOTP authentication is activated

### 11.6.2.2 Realm SMTP Server

A dedicated SMTP server configuration is available for each Realm. If the configuration is left empty, the main SMTP server is used to send notifications to Realm users.

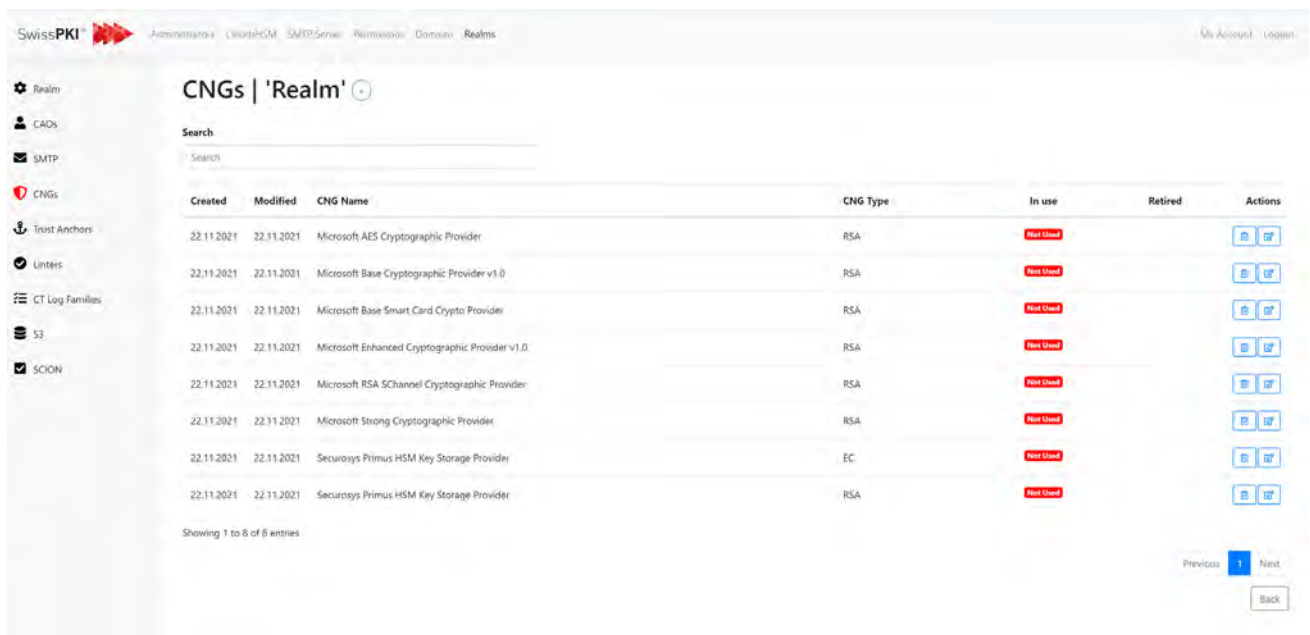
Fields	Description
<b>Host</b>	The SMTP Server Host
<b>Port</b>	The SMTP Server Port
<b>TLS</b>	Activate or not the use of TLS
<b>From</b>	The email sender
<b>Login User</b>	The user used to log into your SMTP Server
<b>Password</b>	The Login User's Password
<b>Signing keystore file</b>	An optional PKCS#12 S/MIME certificate including full certificate chain and private key
<b>Signing key alias</b>	The private key alias to use
<b>Signing keystore password</b>	The PKCS#12 password to unlock the signing private key





















### 11.6.2.3 Realm CNGs

Displays the list of usable Microsoft crypto providers. These crypto providers are available when issuing user or system certificates via Microsoft's auto enrolment. As a CA Operator, when creating a certificate policy template, you can then force the end user machine to use a preselected Microsoft crypto provider.



The screenshot shows the SwissPKI web interface. The main content area is titled 'CNGs | 'Realm'' and contains a table of cryptographic providers. The table has the following columns: Created, Modified, CNG Name, CNG Type, In use, Retired, and Actions. All providers are marked as 'Not Used' in the 'In use' column. The 'Actions' column contains icons for edit and delete.

Created	Modified	CNG Name	CNG Type	In use	Retired	Actions
22.11.2021	22.11.2021	Microsoft AES Cryptographic Provider	RSA	Not Used		 
22.11.2021	22.11.2021	Microsoft Base Cryptographic Provider v1.0	RSA	Not Used		 
22.11.2021	22.11.2021	Microsoft Base Smart Card Crypto Provider	RSA	Not Used		 
22.11.2021	22.11.2021	Microsoft Enhanced Cryptographic Provider v1.0	RSA	Not Used		 
22.11.2021	22.11.2021	Microsoft RSA SChannel Cryptographic Provider	RSA	Not Used		 
22.11.2021	22.11.2021	Microsoft Strong Cryptographic Provider	RSA	Not Used		 
22.11.2021	22.11.2021	Securosys Primus HSM Key Storage Provider	EC	Not Used		 
22.11.2021	22.11.2021	Securosys Primus HSM Key Storage Provider	RSA	Not Used		 

Showing 1 to 8 of 8 entries

Previous **1** Next  
Back

### 11.6.2.3.1 Add Realm CNG

Adding a new CNG is done by clicking on the add button located to the right of the page title. You are redirected to a form where you need to provide the following information:

Fields	Description
<b>CNG Name</b>	The exact CNG name published in the end user's Windows registry.
<b>CNG Type</b>	Two types are available: <ul style="list-style-type: none"> <li>• RSA</li> <li>• EC</li> </ul>

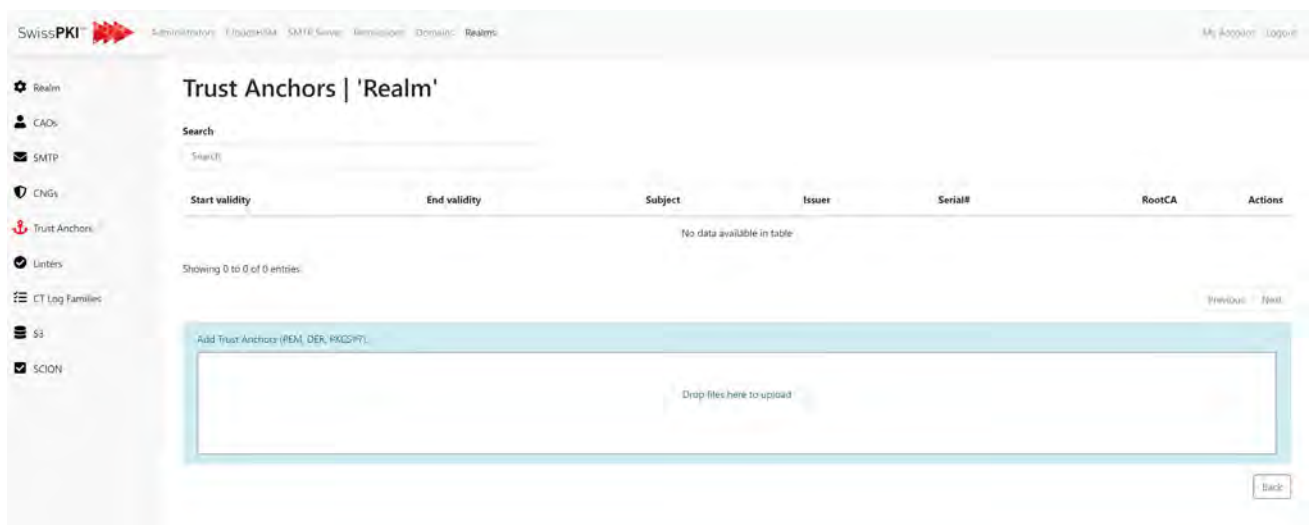


The screenshot shows the SwissPKI administration interface. The breadcrumb trail is: Administration > CloudPKM > SMTP Server > Renewalcert > Domain > Realms. The left sidebar contains a menu with items: Realm, CADs, SMTP, CNGs, Trust Anchors, Linters, CT Log Families, S3, and SCION. The main content area is titled 'CNG | 'Realm'' and contains a form with two fields: 'CNG Name\*' with a text input field containing 'CNG name', and 'CNG Type\*' with a dropdown menu showing 'EC'. At the bottom right of the form are 'Back' and 'Create' buttons.

### 11.6.2.4 Realm Trust Anchor

Allows you to import Root and Subordinate Certificate Authority chains. The trust anchors are used to validate client's CMP protocol certificates if they were issued by another PKI. Additionally, Realm trust anchors are also used in the SCION context to validate renewal requests from external Certification Authorities.

To add a new trust anchor, simply drag and drop a PKCS#7 certificate chain file in the box at the bottom of the page.



### 11.6.2.5 Realm Linters

Linters are Web Service URLs used to inspect certificate content. Linters are specifically used in the context of public trust certificate issuance. We provide Web Services for the standard CertLint, X509Lint and ZLint tools.

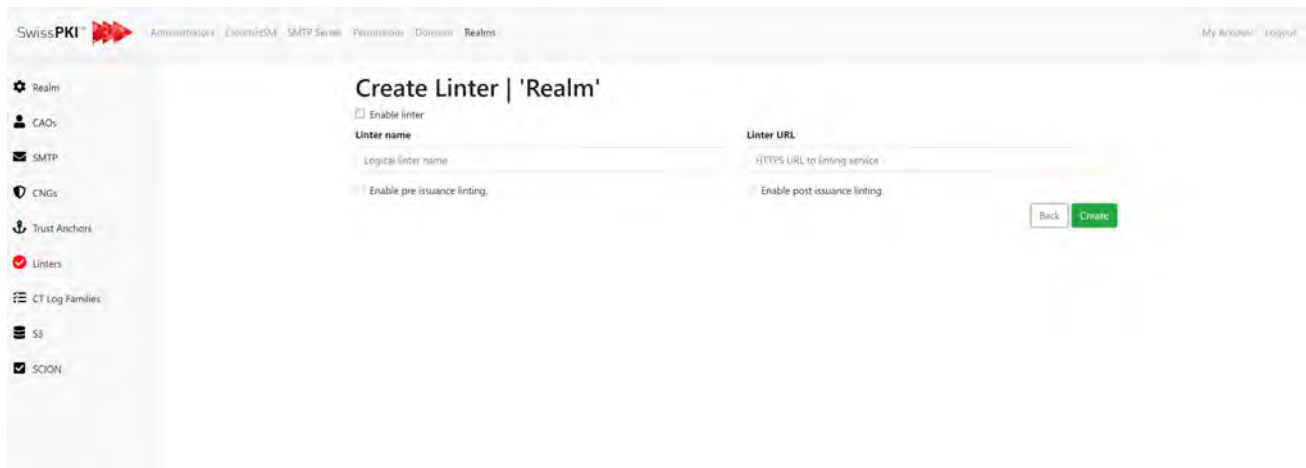
A list of all linters created for your realm is available on the realm linters tab. Linters are then associated to a Realm Certificate Policy Template.



### 11.6.2.5.1 Create Linter

Creating a linter is done by clicking on the add button located on the right of the linter's list page title. You are then redirected to a form where you must provide the following information:

Fields	Description
<b>Enable linter</b>	This checkbox allows you to enable or not the linter. This will enable the pre and post issuance linting checkboxes.
<b>Linter name</b>	The linter's logical name
<b>Linter URL</b>	The linter's URL
<b>Enable pre issuance linting</b>	Allows you to enable or not pre issuance linting
<b>Enable post issuance linting</b>	Allows you to enable or not post issuance linting

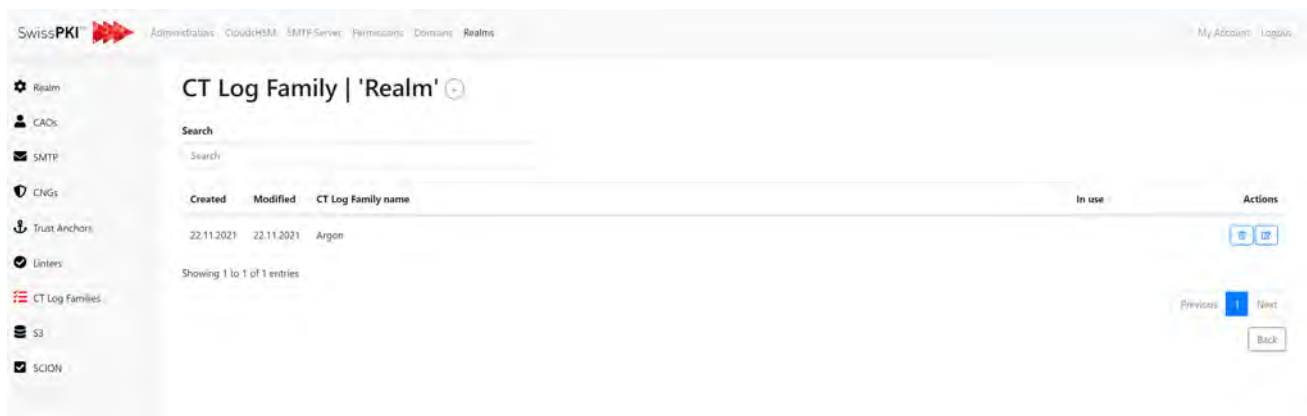


### 11.6.2.6 CT Log Families

Certificate Transparency <sup>13</sup> is used in combination with public trust certificates. When a CA receives a request for a certificate from a domain owner. It checks that the domain owner has the right to request the certificate, and creates a precertificate, which ties the domain to a public key. A precertificate contains all the information a certificate does. It also has a poison extension so that user agents will not accept it. Before a CA can log a certificate, the certificate needs an SCT (Signed Certificate Timestamp). But for the certificate to get an SCT, it needs to have been submitted to a log.

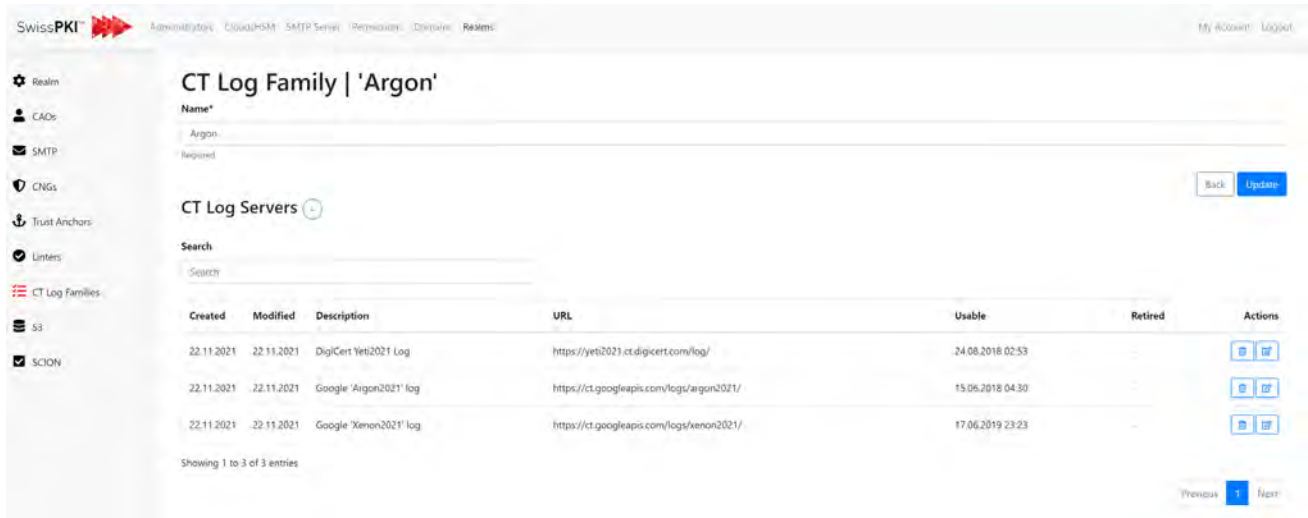
For each Realm, you can configure several CT Log Families which are referenced in Certificate Policy Templates of that Realm. The CA Operator defines which CT Log Families are used when issuing public trust SSL/TLS certificates in precertificate or OCSP stapling mode.

Create CT Log Families using a user defined logical CT Log Family name of your choice and click 'edit' to configure the CT Log Family. In this sample, we have used 'Argon' as the logical CT Log Family name



Click on 'Create' button to add or chose from existing log families

<sup>13</sup> <https://certificate.transparency.dev/howctworks/>



SwissPKI Administration | CA/RA/PSM | SMTP Server | Permissions | Domains | Realms

My Account | Logout

### CT Log Family | 'Argon'

Name\*  
Argon  
Required

CT Log Servers

Created	Modified	Description	URL	Usable	Retired	Actions
22.11.2021	22.11.2021	DigiCert Yeti2021 Log	https://yeti2021.ct.digicert.com/log/	24.08.2018 02:53	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
22.11.2021	22.11.2021	Google 'Argon2021' log	https://ct.googleapis.com/logs/argon2021/	15.06.2018 04:30	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
22.11.2021	22.11.2021	Google 'Xenon2021' log	https://ct.googleapis.com/logs/xenon2021/	17.06.2019 23:23	-	<input type="button" value="edit"/> <input type="button" value="delete"/>

Showing 1 to 3 of 3 entries

Previous 1 Next

Preselected CT Log Families are available from the drop-down menu and will fill in all fields based on the log ser's settings. Note that you can also edit manually the CT Log Family record.



SwissPKI Administration | CA/RA/PSM | SMTP Server | Permissions | Domains | Realms

My Account | Logout

### CT Log Server | 'Google Argon2021' log

Description\*  
Google 'Argon2021' log

CT URL\*  
https://ct.googleapis.com/logs/argon2021/

CT Log Id  
9jUL9F3MCIUVBgiMjRWjwVNEkzv98MlyALzE7xZOM+

CT Log Key  
MFlxvEwYHkaZizp0CAQYIKaZizp0DAQCDQgA5Te8mZorZko4Ykx9gl20iEw3cw/tbr5xkoQlm#B18akf5D+MnllgGNl0F0m0eYGrFV65wLRIORX6k6xw==

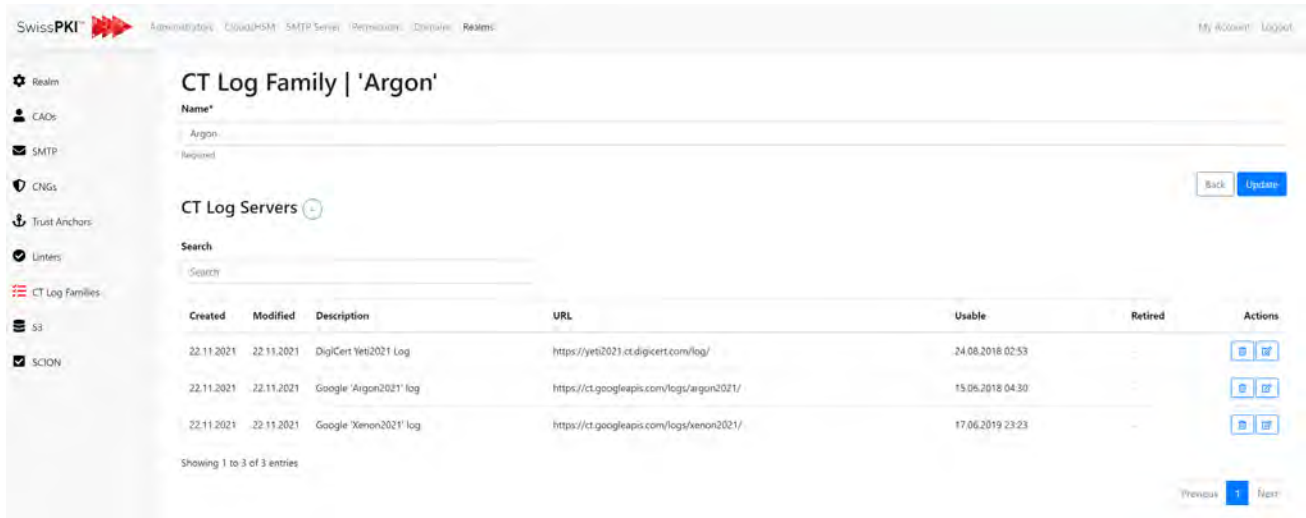
Usable  
15.06.2018 04:30:13

Retired  
15.06.2018 04:30:13







Start Inclusive  
01.01.2021 01:00:00

End Exclusive  
01.01.2022 01:00:00

Click 'Create' to add the edit/select CT Log Family to the Realm. For public trust, you will need to include at the minimum three CT Log Families. Once created, the list of CT Log Servers is displayed on the main page of the CT Log Family:



The screenshot shows the 'SwissPKI' web interface. The main content area is titled 'CT Log Family | 'Argon''. Below the title, there is a 'Name' field containing 'Argon' and a 'Required' checkbox. A 'Back' button and an 'Updates' button are visible. Below this is a section for 'CT Log Servers' with a search bar. A table lists the configured CT Log Servers:

Created	Modified	Description	URL	Usable	Retired	Actions
22.11.2021	22.11.2021	DigiCert Yeti2021 Log	https://yeti2021.ct.digicert.com/log/	24.08.2018 02:53	-	 
22.11.2021	22.11.2021	Google 'Argon2021' log	https://ct.googleapis.com/logs/argon2021/	15.06.2018 04:30	-	 
22.11.2021	22.11.2021	Google 'Xenon2021' log	https://ct.googleapis.com/logs/xenon2021/	17.06.2019 23:23	-	 

Showing 1 to 3 of 3 entries

The certificate issuance process with CT Log **'enabled'** on a Certificate Policy Template will create the SCT in precertificate or OCSP stapling depending on the settings and use the log server of the corresponding year. Adding subsequent years to the CT Log Family will get automatically picked up when changing into a new year. Note that certificate issuance will fail if you do not have at least three valid CT Log Server in a CT Log Family. Because CT Log Servers are not always available, we recommend that you create CT Log Families with at least 5 CT Log Servers to avoid certificate issuance failure when one of the CT Log Server does not reply during the issuance process.



### 11.6.2.7 S3 Object Store

The S3 object store may be used to store certificate registration documents to offload the amount of data stored in the PostgreSQL database. When enabled, this option is used in conjunction with the Realm’s Registration Rule (see section 12.2.4.1 Registration Rules for details).

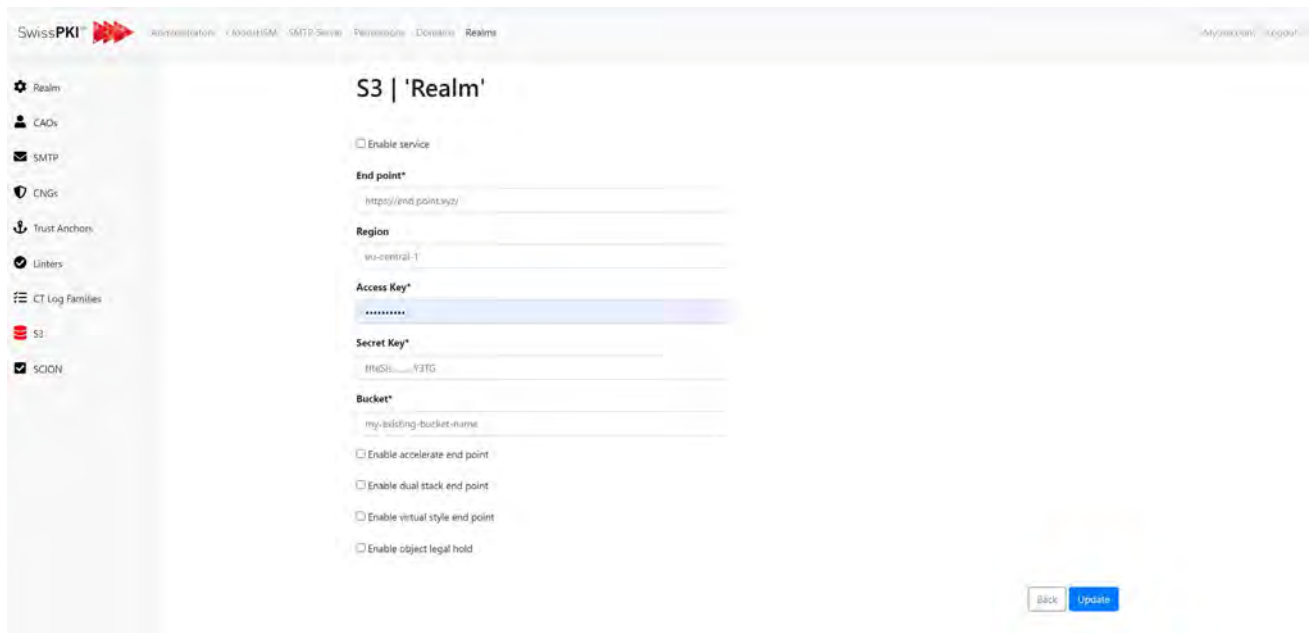
Fields	Description
<b>Enable service</b>	Enables/disables the service. See section 8.14 Scheduler for scheduling details
<b>End point</b>	URL to the S3 service
<b>Region</b>	For AWS S3, the AWS region. For non AWS service such as minio set the value to <i>eu-central-1</i>
<b>Access key</b>	S3 access key or username for minio
<b>Secret key</b>	S3 account secret key
<b>Bucket</b>	Bucket name where the registration documents are stored and retrieved
<b>Enable accelerate endpoint</b>	AWS feature
<b>Enable dual stack endpoint</b>	AWS feature
<b>Enable virtual style end point</b>	AWS feature
<b>Enable object legal hold</b>	AWS feature

**Note:** Document file path is composed of

- Realm UUID followed by
- /registration/document/ followed by
- Certificate common name followed by
- Certificate Order UUID followed by
- Document UUID and file extension (pdf/jpg)

Each document has associated tags:

- order-reference: *ord-uuid* – the UUID of the certificate order
- created-by: *issuer full name* – the name of the Registration Officer
- certificate-serial: *serial number* – the certificate serial number in HEX uppercase
- certificate-subject-cn: *Subject Common Name* – certificate subject common name (if available)
- file-name: *document name* – the document name
- rri: *rri-uuid* – the registration document UUID



### 11.6.2.8 SCION

When enabled, you configure the Realm's Web Service to perform SCION Identity Repository certificate content validation when integrating the SwissPKI SCION PKI Adapter. For detailed information about SCION please refer to <https://www.scion-architecture.net/pdf/SCION-book.pdf>.

Fields	Description
<b>Validation URL</b>	Define the SCION Identity Repository validation service to validate SCION AS Identity certificate content
<b>Shared Secret</b>	Contains the shared secret used for HMAC256 JWT for authentication.



## 12 Operator UI

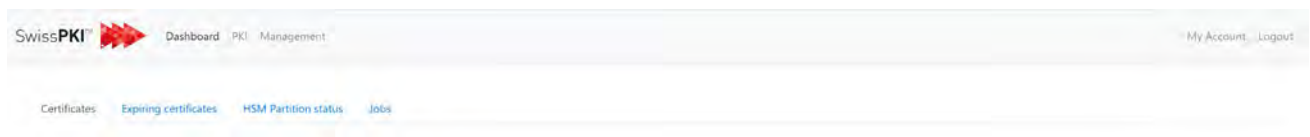
The Operator UI is accessible at the deployed URL **Error! Hyperlink reference not valid.** or `DNS>/operator/` to registered CA Operator and/or Auditor roles. As a CA Operator, you can:

- Access the Realm's Dashboard
- Manage the Realm's settings
- Manage the Realm's PKI entities

### 12.1 Dashboard

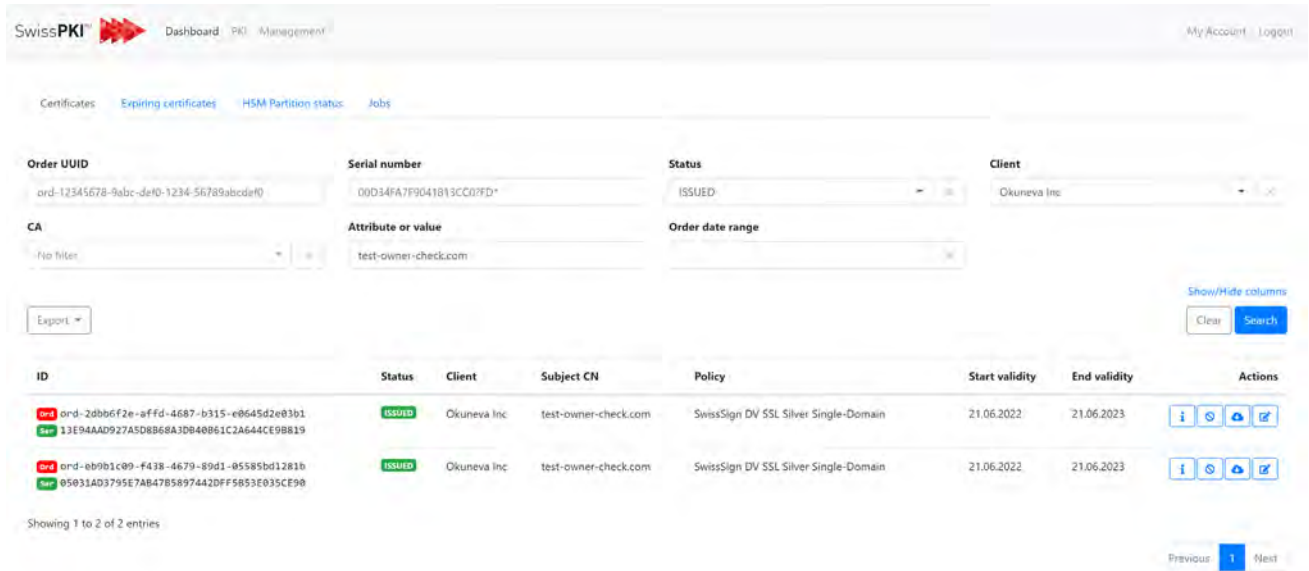
The Operator UI Dashboard gives you an overview of:

- Issued certificates
- Expiring certificates
- HSM partition status
- Job status



## 12.1.1 Issued certificates

Search certificates and certificate orders for all Clients and Certification Authorities within the Realm



SwissPKI Dashboard - PKI Management

Certificates Expiring certificates HSM Partition status Jobs

Order UUID: ord-12345678-9abc-def0-1234-56789abcde0

Serial number: 00D34FA7F9041813CC03FD\*

Status: ISSUED

Client: Okuneva Inc

CA: No filter

Attribute or value: test-owner-check.com

Order date range: [ ]

Export [ ] Show/Hide columns [ ] Clear [ ] Search [ ]

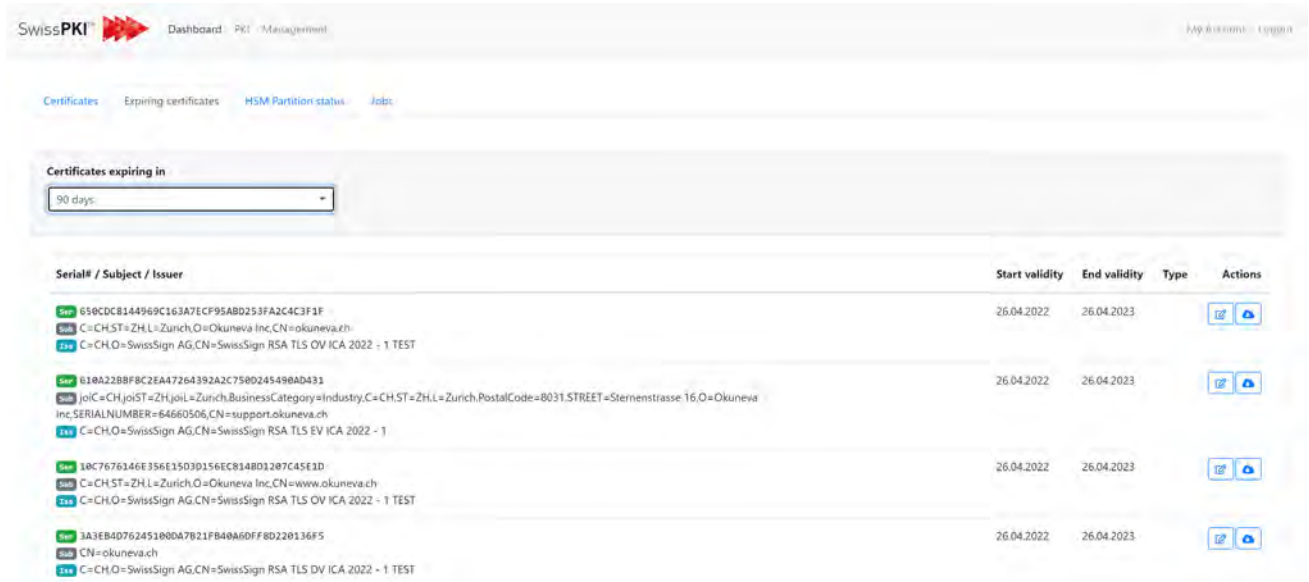
ID	Status	Client	Subject CN	Policy	Start validity	End validity	Actions
ord-20bb6f2e-afFd-4687-b315-e8645d2e03b1 13E94AAD927A5D8868A3DB40861C2A644CE9B819	ISSUED	Okuneva Inc	test-owner-check.com	SwissSign DV SSL Silver Single-Domain	21.06.2022	21.06.2023	[i] [S] [C] [E]
ord-eb9b1c09-f438-4679-89d1-85585bd1281b 05031AD3795E7AB4785897442DFF5853E035CE90	ISSUED	Okuneva Inc	test-owner-check.com	SwissSign DV SSL Silver Single-Domain	21.06.2022	21.06.2023	[i] [S] [C] [E]

Showing 1 to 2 of 2 entries

Previous [ 1 ] Next

## 12.1.2 Expiring certificates

List expiring certificates in 15, 30, 45, 60, 75 or 90 days for the logged in Realm.



SwissPKI Dashboard - PKI Management

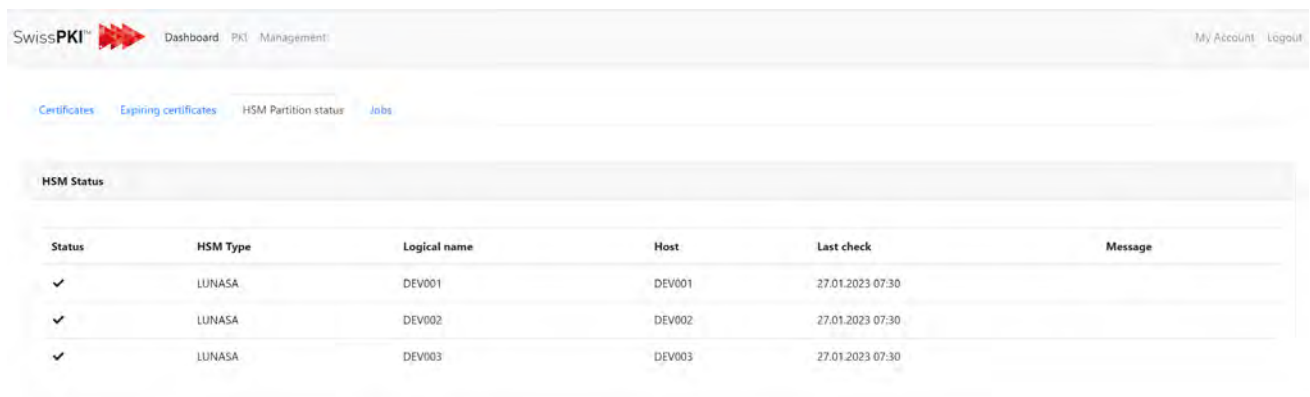
Certificates Expiring certificates HSM Partition status Jobs

Certificates expiring in: 90 days

Serial# / Subject / Issuer	Start validity	End validity	Type	Actions
650CDB144969C163A7ECP95ABD253FA2C4C3F1F C=CH,ST=ZH,L=Zurich,O=Okuneva Inc,CN=okuneva.zh C=CH,O=SwissSign AG,CN=SwissSign RSA TLS OV ICA 2022 - 1 TEST	26.04.2022	26.04.2023	[E] [C]	
310A228BF8C2EA47264392A2C758D245498AD431 [o]C=CH,[o]ST=ZH,[o]L=Zurich,[o]BusinessCategory=Industry,C=CH,ST=ZH,L=Zurich,PostalCode=8031,STREET=Sternenstrasse 16,O=Okuneva Inc,SERIALNUMBER=64660506,CN=support.okuneva.ch C=CH,O=SwissSign AG,CN=SwissSign RSA TLS EV ICA 2022 - 1	26.04.2022	26.04.2023	[E] [C]	
30C7676346E35E13503D156EC83ABD1207C45E1D C=CH,ST=ZH,L=Zurich,O=Okuneva Inc,CN=www.okuneva.ch C=CH,O=SwissSign AG,CN=SwissSign RSA TLS OV ICA 2022 - 1 TEST	26.04.2022	26.04.2023	[E] [C]	
3A3EB4D76245180DA7B21FB40A6DFF8D228136F5 CN=okuneva.ch C=CH,O=SwissSign AG,CN=SwissSign RSA TLS DV ICA 2022 - 1 TEST	26.04.2022	26.04.2023	[E] [C]	

### 12.1.3 HSM partition status

Active HSM partition status is updated every 15 minutes. If one of the partitions is unavailable or has missing key alias references an error message is display for this HSM partition.



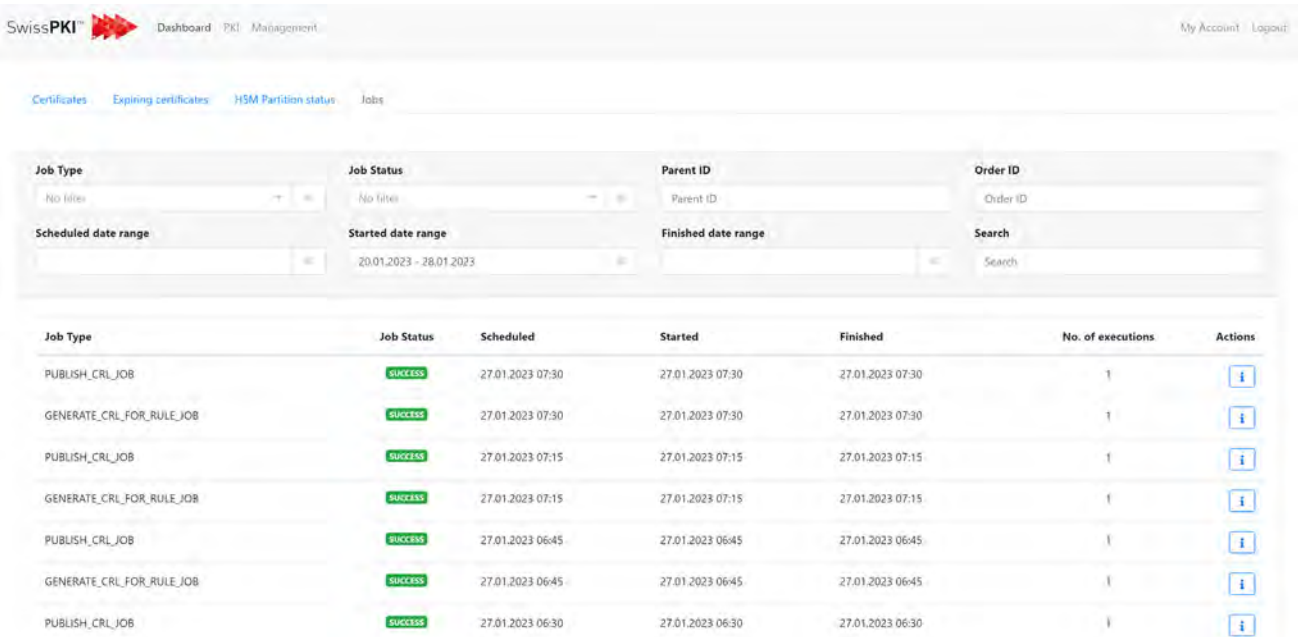
The screenshot shows the 'HSM Status' page in the SwissPKI management interface. The page has a navigation bar with 'Dashboard', 'PKI Management', 'My Account', and 'Logout'. Below the navigation bar, there are tabs for 'Certificates', 'Expiring certificates', 'HSM Partition status', and 'Jobs'. The main content area is titled 'HSM Status' and contains a table with the following data:

Status	HSM Type	Logical name	Host	Last check	Message
✓	LUNASA	DEV001	DEV001	27.01.2023 07:30	
✓	LUNASA	DEV002	DEV002	27.01.2023 07:30	
✓	LUNASA	DEV003	DEV003	27.01.2023 07:30	

**Note:** If a PKI is in the state 'disabled' and is using an HSM partition, then the HSM partition status is not checked unless the HSM partition is referenced by another PKI entity which also uses the HSM partition. Also, the HSM Partition status check signs a random 16 bytes data with each active private key in your Realm.

## 12.1.4 Job Status

Because some of the certificate processing tasks may take some time, SwissPKI uses an asynchronous processing for issuing, revoking, or renewing certificates during workflow execution. The Job Status view lets you search for related Jobs and display the details of its status and content.



The screenshot shows the SwissPKI Job Status interface. At the top, there is a navigation bar with 'SwissPKI' logo, 'Dashboard', 'PKI Management', 'My Account', and 'Logout'. Below the navigation bar, there are tabs for 'Certificates', 'Expiring certificates', 'HSM Partition status', and 'Jobs'. The 'Jobs' tab is active.

The main content area contains a search and filter section with the following fields:

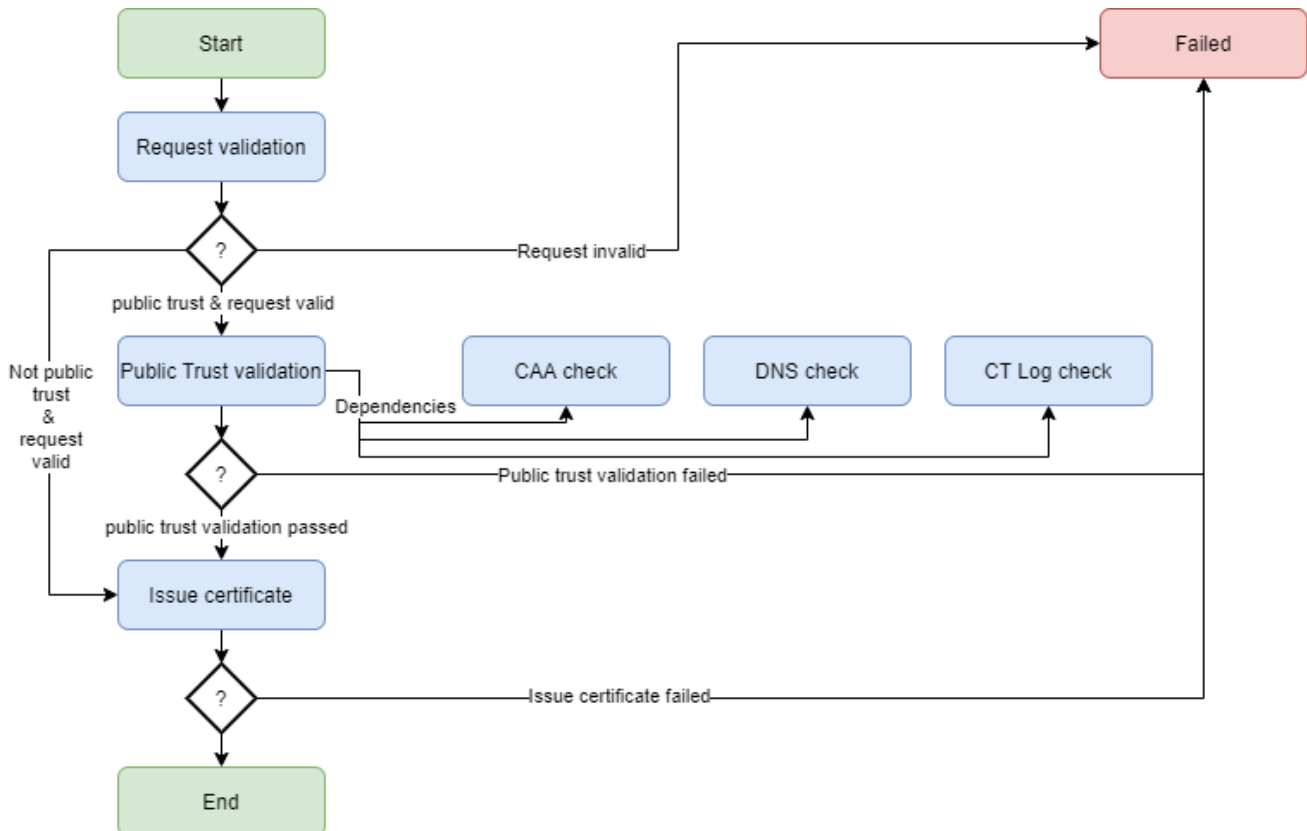
- Job Type:** No filter
- Job Status:** No filter
- Parent ID:** Parent ID
- Order ID:** Order ID
- Scheduled date range:** [Empty]
- Started date range:** 20.01.2023 - 28.01.2023
- Finished date range:** [Empty]
- Search:** Search

Below the search section is a table with the following columns: Job Type, Job Status, Scheduled, Started, Finished, No. of executions, and Actions.

Job Type	Job Status	Scheduled	Started	Finished	No. of executions	Actions
PUBLISH_CRL_JOB	SUCCESS	27.01.2023 07:30	27.01.2023 07:30	27.01.2023 07:30	1	<a href="#">i</a>
GENERATE_CRL_FOR_RULE_JOB	SUCCESS	27.01.2023 07:30	27.01.2023 07:30	27.01.2023 07:30	1	<a href="#">i</a>
PUBLISH_CRL_JOB	SUCCESS	27.01.2023 07:15	27.01.2023 07:15	27.01.2023 07:15	1	<a href="#">i</a>
GENERATE_CRL_FOR_RULE_JOB	SUCCESS	27.01.2023 07:15	27.01.2023 07:15	27.01.2023 07:15	1	<a href="#">i</a>
PUBLISH_CRL_JOB	SUCCESS	27.01.2023 06:45	27.01.2023 06:45	27.01.2023 06:45	1	<a href="#">i</a>
GENERATE_CRL_FOR_RULE_JOB	SUCCESS	27.01.2023 06:45	27.01.2023 06:45	27.01.2023 06:45	1	<a href="#">i</a>
PUBLISH_CRL_JOB	SUCCESS	27.01.2023 06:30	27.01.2023 06:30	27.01.2023 06:30	1	<a href="#">i</a>

### 12.1.4.1 Process Orchestration

Simplified example of the Certificate Issuance workflow:



- Process Orchestration via Jobs
  - Modeling the whole workflow using jobs ( tasks )
  - Every job implements a specific task
  - Failed jobs may be retried 0-n times before failing for good
  - The next job(s) are scheduled depending on the output of the previous job
  - A job can have multiple child jobs. The parent job gets executed once all child jobs finished executing



- Dispatching jobs
  - Rabbit MQ is used to schedule jobs on job specific request and reply queues
  - Job runners are picking up the jobs from the queue, processing them and sending the response back to a reply queue
  - An orchestration service will read the reply and decide what job needs to be executed next
  - Job runners are scalable. By using the messaging system, we can ensure that only one job runner is able to pick up a job

### 12.1.4.2 Job Types

Job	Description
<b>REVOKE_CERTIFICATE_JOB</b>	Job is started upon RA or Operator certificate revocation
<b>MANUAL_PUBLISH_CERTIFICATE_JOB</b>	RA/CA Operator manually request a certificate publication
<b>PUBLISH_CERTIFICATE_ORDER_JOB</b>	CA requests a certificate publication
<b>MANUAL_UNPUBLISH_CERTIFICATE_JOB</b>	RA/CA Operator manually request a certificate publication
<b>PUBLISH_CRL_JOB</b>	Job is started after a CRL/ARL is generated
<b>GENERATE_CRL_JOB</b>	Job is started when a CAO manually generates a CRL/ARL
<b>GENERATE_LAST_CRL_JOB</b>	Job is started when a CAO manually generates a Last CRL/ARL
<b>GENERATE_CRL_FOR_RULE_JOB</b>	Job is started when the Scheduler generates time/day based CRL/ARL
<b>REGISTER_CRL_FOR_RULE_JOB</b>	Job is started when a new CRL publication rule is created by a CAO
<b>UPDATE_CRL_FOR_RULE_JOB</b>	Job is started when a CRL publication rule is updated by a CAO
<b>UNREGISTER_CRL_FOR_RULE_JOB</b>	Job is started when a CRL publication rule is deleted by a CAO
<b>RA_CREATE_CERT_COMMENT_JOB</b>	Job is started when a RAO creates a comment for an issued certificate
<b>RA_CREATE_CERT_REG_DOCUMENT_JOB</b>	Job is started when a RAO creates a registration document for an issued certificate
<b>RA_DELETE_CERT_REG_DOCUMENT_JOB</b>	Job is started when a RAO deletes a registration document for an issued certificate

<b>RA_CREATE_CERT_RENEWAL_EMAIL_JOB</b>	Job is started when a RAO add an email recipient to an issued certificate renewal rule
<b>RA_DELETE_CERT_RENEWAL_EMAIL_JOB</b>	Job is started when a RAO deletes an email recipient to an issued certificate renewal rule
<b>RA_AUTHORIZE_CERT_REVOKE_JOB</b>	The job is started upon every RAO revocation request
<b>RA_NOTIFY_AUTHORIZE_CERT_REVOKE_JOB</b>	Job is started upon every RAO revocation request to notify potential Authorizers
<b>RA_CERT_IMPORT_JOB</b>	Job is started when a RAO imports a certificate for an 'External' CA
<b>ISSUE_SUBMIT_CERTIFICATE_ORDER_JOB</b>	Job is started for every certificate renewal issuance with rekeying. Validates the renewal request and optionally sets the processing in WAITING if the renewal requires a new CSR from its recipient.
<b>ISSUE_CERTIFICATE_RENEWAL_VALIDATION_JOB</b>	Job is started for every certificate issuance. It creates an initial 'empty' certificate order. The Certificate Order UUID can be used to search for Jobs related to the certificate issuance workflow.
<b>ISSUE_KEY_VALIDATION_JOB</b>	After creation of a Certificate Order, a key validation Job validates the requested public key prior to pre validation tasks.
<b>ISSUE_PRE_VALIDATION_JOB</b>	The job is the parent job for all pre validation tasks
<b>ISSUE_GENERATE_TBS_JOB</b>	Job is started to generate a TBS which will get signed when all pre validation tasks are successfully executed
<b>ISSUE_POLICY_VALIDATION_JOB</b>	Job is started to validate the certificate policy against the requested TBS certificate
<b>ISSUE_CAA_CHECK_VALIDATION_JOB</b>	Job is executed when a CAA check is enabled on the certificate policy template for the requested certificate issuance.

<b>ISSUE_DOMAIN_OWNER_CHECK_VALIDATION_JOB</b>	Job is executed when a DNS Owner check is enabled on the certificate policy template for the requested certificate issuance.
<b>ISSUE_PRE_LINTING_JOB</b>	Job is executed when a TBS certificate is ready for issuance.
<b>ISSUE_PRE_ISSUE_CERTIFICATE_JOB</b>	The parent Job for all pre issuance tasks
<b>ISSUE_CT_LOG_PRE_CERT_PUBLICATION_JOB</b>	Job is executed when a CT log publication is required, the poison pill being removed from the TBS certificate structure
<b>ISSUE_ISSUE_CERTIFICATE_JOB</b>	Job is executed when signing the TBS certificate and produce the final certificate
<b>ISSUE_POST_ISSUE_CERTIFICATE_JOB</b>	The parent Job for all post issuance tasks
<b>ISSUE_POST_LINTING_CERTIFICATE_JOB</b>	Job is started when post linting is enabled
<b>PUBLISH_POST_CERTIFICATE_JOB</b>	Job is started after issuance of the certificate. Clean up and publishing actions are taken during this step.
<b>ISSUE_CT_LOG_PUBLICATION_JOB</b>	Job started during certificate pre issuance to obtain the CT log to include in a certificate extension
<b>ISSUE_AUTHORIZATION_JOB</b>	Job is started when an authorization is executed (one of <i>accept</i> or <i>reject</i> )
<b>ISSUE_NOTIFY_ISSUED_JOB</b>	Job is started for each certificate issuance. Notifies recipients that the certificate is issued based on the associated notification rule.
<b>ISSUE_NOTIFY_RENEWAL_JOB</b>	Job is started for each certificate renewal issuance. Notifies recipients that the certificate is renewed based on the associated notification rule.
<b>ISSUE_UPDATE_RENEWAL_JOB</b>	Job is started for each certificate renewal issuance if there is an auto-revoke rule on the renewal rule and its value equals 0.

<b>ISSUE_REVOKE_RENEWED_CERTIFICATE_JOB</b>	Job is started for each certificate renewal issuance. Updates the order information with the previous order.
<b>ISSUE_NOTIFY_P12_RETRIEVAL_JOB</b>	
<b>ISSUE_NOTIFY_HSM_RETRIEVAL_JOB</b>	
<b>ISSUE_SET_P12_PIN_JOB</b>	
<b>SCEP_PKI_OPERATION_JOB</b>	Job is started when a SCEP Operation request is received to via the SCEP Protocol handler
<b>MICROSOFT_CES_REQUEST_JOB</b>	Job is started when a Microsoft CES PKCS#10 request for autoenrollment is received from a Microsoft Domain
<b>MICROSOFT_CES_STATUS_JOB</b>	Job is started when a Microsoft CES request for autoenrollment is received from a Microsoft Domain
<b>MICROSOFT_ENROLMENT_POLICIES_JOB</b>	Job is started when a Microsoft request for autoenrollment is received from a Microsoft Domain
<b>MICROSOFT_CES_QUERY_STATUS_JOB</b>	Job is started when a Microsoft Certificate Status request for autoenrollment is received from a Microsoft Domain
<b>MICROSOFT_CES_KET_JOB</b>	Job is started when a Microsoft Key Exchange request for autoenrollment is received from a Microsoft Domain
<b>MICROSOFT_CES_UNKNOWN_JOB</b>	Job is started when an unknown Microsoft request for autoenrollment is received from a Microsoft Domain
<b>SEND_EMAIL_JOB</b>	Job started when email notification is executed.
<b>GENERATE_CROSS_SIGNED_CSR_JOB</b>	Job started when a CAO generates a cross signed requests for a select CA.
<b>AUTOMATIC_RENEW_TSA_JOB</b>	The Scheduler starts this Job when the TSA certificates are ready for automatic renewals.

<b>AUTOMATIC_RENEW_DSS_JOB</b>	The Scheduler starts this Job when the DSS certificates are ready for automatic renewals.
<b>AUTOMATIC_RENEW_OCSP_JOB</b>	The Scheduler starts this Job when the OCSP certificates are ready for automatic renewals.
<b>AUTOMATIC_RENEW_CMP_JOB</b>	The Scheduler starts this Job when the CMP certificates are ready for automatic renewals.
<b>HSM_PIN_RESET</b>	HSM PIN Reset jobs for CA and Scheduler processes when an CA Operator updates HSM partition PINs
<b>AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB</b>	Air gapped CA certificate issuance request
<b>AIR_GAPED_OFFLINE_CA_SUB_CA_ISSUANCE_JOB</b>	Air gapped Sub CA certificate issuance request
<b>AIR_GAPED_OFFLINE_CA_XSIGN_JOB</b>	Air gapped CA cross signed request
<b>AIR_GAPED_OFFLINE_CA_CRL_JOB</b>	Air gapped CRL request
<b>AIR_GAPED_OFFLINE_CA_LAST_CRL_JOB</b>	Air gapped Last CRL request
<b>PROBE_*</b>	Probe roundtrips per deployed process

### 12.1.4.3 Job Status

Status	Description
<b>WAITING</b>	Job is in waiting status (e.g., waits for its children to end processing or input from an external event)
<b>PENDING</b>	Job is sent to the queue and ready to process
<b>PROCESSING</b>	Job is processing
<b>SUCCESS</b>	Job is successful
<b>FAILED</b>	Job failed
<b>SCHEDULE_REQUEST</b>	Job is scheduled but not yet sent to a queue (e.g., connection error)
<b>SCHEDULE_RESPONSE</b>	Job is scheduled but not yet sent to a queue (e.g., connection error)
<b>RETRY</b>	Job is marked for retry

## 12.2 Manage

From the 'Manage' main menu, you define the overall Realm configuration:

1. Manage Users  
Create, edit, activate, deactivate, and delete users
2. Manage Auditors  
Assign Auditor roles to existing users
3. Manage Clients  
Create and edit the Clients which have access to the Registration Authority
4. Manage Rules  
Define registration and authorization rules
5. Manage Notifications  
Define notification content to send to recipients based on specific workflow events
6. Manage Registration Sources  
Create and manage external certificate registration sources
7. Manage HSMs  
Create and manage the HSM partitions used by your PKI entities
8. Manage Permissions  
Create and manage permission templates associated with the PKI roles
9. Access Audit log  
Query and/or export audit events

### 12.2.1 Users

As a CA Operator, you manage the users within your realm by associating them to specific roles along with permissions. Depending on the authentication mechanism you setup (see [8.1.5 Users](#)), users can get onboarded automatically by SwissPKI.

As a CA Operator, your user management tasks are:

1. Create, edit, or delete user information
2. Validate user information depending on the onboarding mechanisms configured at deployment
3. Activate or inactive users
4. Associate roles and permissions to your users



### 12.2.1.1 User Types and Status

Users can be of two types:

User Types	Description
<b>USER</b>	A user who can perform login operations to the RA UI and optionally use the REST API
<b>SERVICE</b>	A user who has no RA UI login capability but can optionally use the REST API Service account users are typically associated to Clients for automation purposes. Typically, users associated to 'physical' persons may come and go and with them come and go their API Keys. To keep your Client's automation processes up and running, you would naturally choose a SERVICE account.

Users have different status:

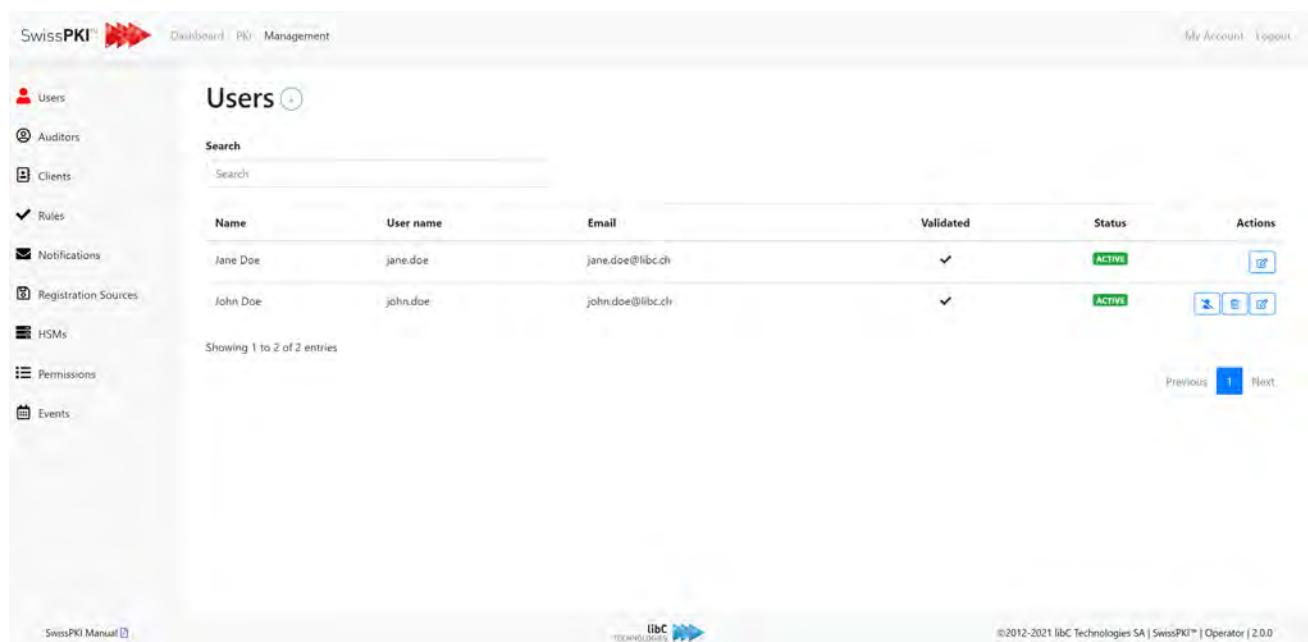
User Status	Description
<b>PENDING_VALIDATION</b>	<p>When you manually create a user in your Realm, its status is set to PENDING_VALIDATION.</p> <p>If the SwissPKI authentication mechanisms <i>Username/Password with TOTP</i> (default) is enabled, a registration email is sent to the created user to validate its email address and configure its password. Upon successful confirmation, the user account is set to validated</p> <p>If <i>Username/Password with TOTP</i> is disabled, the user account stays in status PENDING_VALIDATION until you (or another CA Operator) validate the account.</p> <p>Authentication via LDAP, Kerberos and OIDC with onboarding will automatically set the user account to <b>ACTIVE and VALIDATED</b> upon successful login. If automatic onboarding is disabled, you must manually (or via REST API) validate and activate the user account.</p> <p>It is only when the account is set to <b>ACTIVE and VALIDATED</b> that the user can login via REST API or the RA UI.</p>
<b>ACTIVE</b>	The user account is ACTIVE
<b>INACTIVE</b>	<p>The user account is INACTIVE. The user cannot login to the RA UI or via REST API.</p> <p>The user account has still all its roles associated to it.</p>
<b>DELETED</b>	<p>The user account is DELETED. The user cannot login to the RA UI or via REST API.</p> <p>The user account has no more role associated to it.</p>

A user account **must** be **ACTIVE** and **VALIDATED** to login via RA UI or REST API. Additionally, at least **one** role must be associated with the user account.

As a CA Operator, you can only associate the following roles to the user account within your Realm:

1. Auditor
2. RA Officer
3. Authorizer

For the Auditor, RA Officer, and Authorizer roles, you **must** also associate a Permission Template (see *12.2.8 Permissions*) which defines the operations the role is allowed execute. Permission template selection occurs when associating a user to a Client (see *12.2.3.4 RAOs* and *12.2.3.5 Authorizers*) or as an Auditor (see *12.2.2 Auditors*).



The screenshot shows the 'Users' management page in the SwissPKI interface. The page title is 'Users' and it includes a search bar. Below the search bar is a table with the following columns: Name, User name, Email, Validated, Status, and Actions. Two users are listed: Jane Doe and John Doe. Both are validated and active. The Actions column contains icons for editing and deleting each user.

Name	User name	Email	Validated	Status	Actions
Jane Doe	jane.doe	jane.doe@libc.ch	✓	ACTIVE	[Edit] [Delete]
John Doe	john.doe	john.doe@libc.ch	✓	ACTIVE	[Edit] [Delete]

Showing 1 to 2 of 2 entries

Previous 1 Next

SwissPKI Manual


libC TECHNOLOGIES

©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.1.2 Create New User

Creating a new user is done by clicking on the add button located on the right of the page title. After clicking, you are redirected to a form where you need to provide the following information: Once you clicked on the create button, the new user will receive a confirmation email containing further indication on how to complete the account's configuration.

Fields	Description
<b>Service Account</b>	Indicate if this user is of type service account
<b>User Name</b>	The user's name must be at least 8 characters long and cannot contain spaces.
<b>Email</b>	The user's email
<b>Title</b>	The user's title
<b>First name</b>	The user's first name
<b>Last name</b>	The user's last name
<b>Mute notification</b>	If the user is assigned RAO role at a later stage, you may optionally set notification muting. Refer to <i>12.2.5.1 Notifications and Recipients</i>

SWISSPKI™  Dashboard PKI Management My Account Logout

**Create User**

Service account

**User name\***  
User name   
Required

**Title**  
MR

**First name\***  
First name


Mute renewal notifications from other RAOs  
 Mute issuance notifications from other RAOs  
 Mute revocation notifications from other RAOs  
 Mute DNS change notifications from other RAOs

**Email\***  
Email   
Required

**Language**  
English

**Last name\***  
Last name

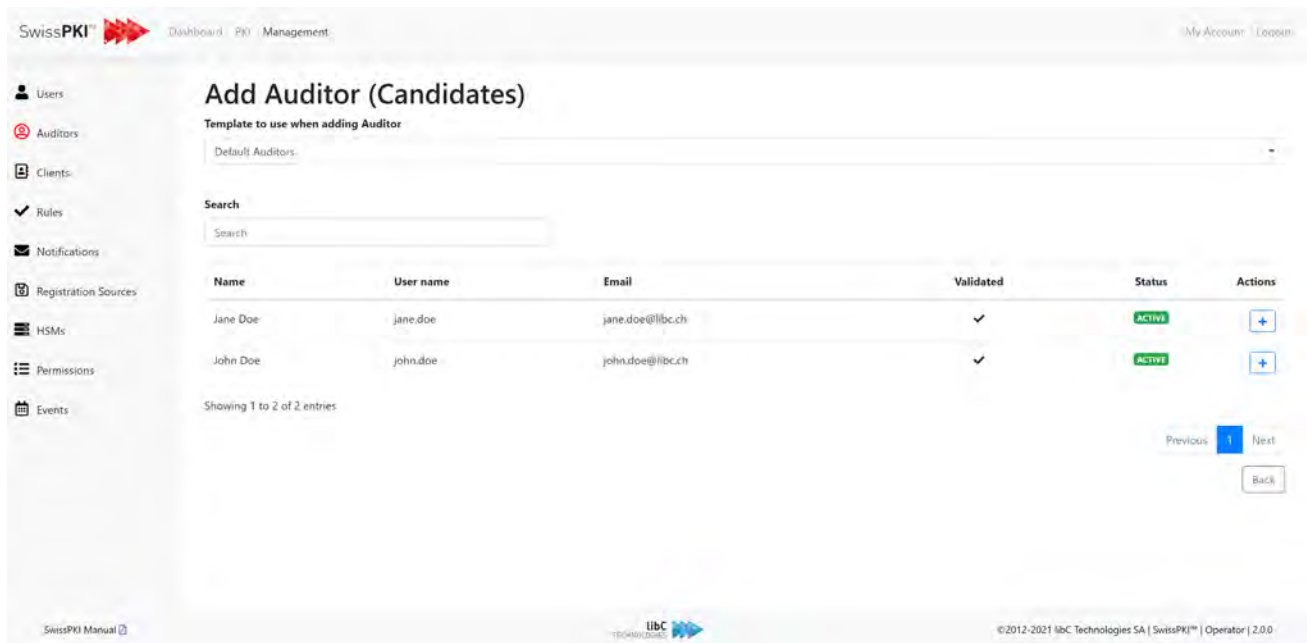
Mute authorization notifications from other RAOs  
 Mute recovery notifications from other RAOs  
 Mute end user email validation notifications from other RAOs

[SwissPKI Manual](#)  ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

## 12.2.2 Auditors

Any user within your Realm can be assigned the role 'Auditor.' The Auditor role gives access to the audit log when logging to the Operator UI. By default, all CA Operators have permissions to access (view) the audit logs.

Select a user from your Realm to assign the role Auditor. Select the permission template to apply to the role from the drop-down menu



The screenshot shows the 'Add Auditor (Candidates)' interface in the libC Operator UI. The page title is 'Add Auditor (Candidates)'. Below the title, there is a dropdown menu for 'Template to use when adding Auditor' with 'Default Auditors' selected. A search bar is present below the dropdown. The main content is a table with the following columns: Name, User name, Email, Validated, Status, and Actions. The table contains two entries:

Name	User name	Email	Validated	Status	Actions
Jane Doe	jane.doe	jane.doe@libc.ch	✓	ACTIVE	[+]
John Doe	john.doe	john.doe@libc.ch	✓	ACTIVE	[+]

Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom right of the table area, there are navigation buttons: 'Previous', '1', 'Next', and 'Back'.

### 12.2.3 Clients

Please refer to *8.1.3 Clients* for a detailed description.

A Client represents a groups of Roles, settings allowed to issue and manage certificates for which certificate policies and associated rules are assigned:

1. RAOs  
RAOs lists the Realm users with an RAO role assigned to the client. RAOs can access the Registration UI and manage certificates for the assigned Clients.
2. Authorizers  
Authorizers lists the Realm users with an Authorizer role assigned to the client. Authorizers can access the Registration UI and manage authorization requests for the assigned Clients.
3. Validation Rules  
Validation rules are external HTTPS services which can be implemented to provide additional <sup>14</sup> certificate content validation when certificates are issued. You implement a REST Web Service which receives, for each certificate issuance (in PRE VALIDATION stage), a callback with the TBS and Policy information.
4. CMP  
Allows to register authorized Signing Certificates and associated Certificate Chain to enable the Client to send signed CMP requests. The certificates are used to validate the CMP signature. Every single CMP certificate policy associated with a Client requires a matching signing certificate to authenticate the client CMP request.
5. ACME Tokens  
If the Client has ACME certificate policies assigned to it, you will find all ACME tokens and DNS issued to this Client.
6. SCEP  
If the Client has SCEP certificate policies assigned to it, then you will find all SCEP registration URLs and PINs made available to the Client.
7. Policies  
Lists the certificate policies, protocols and rules associated with the Client.
8. DNS Server  
Offers a possibility to override <sup>15</sup> the SwissPKI root DNS for the Client. This may cover situations where the deployed SwissPKI may have to rely on custom deployed DNS servers.

---

<sup>14</sup> Additional validation means custom validation implementations in addition to the standard SwissPKI content validation rules

<sup>15</sup> This override does not override the root DNS for CAA Checks

## 9. Domains

For domain validations which do not involve DNS Owner Checks, the domain validation allows you to define RFC822 and/or DNS value to validate when issuing certificates without having to go through the deployment and installation of validation tokens on DNS servers.

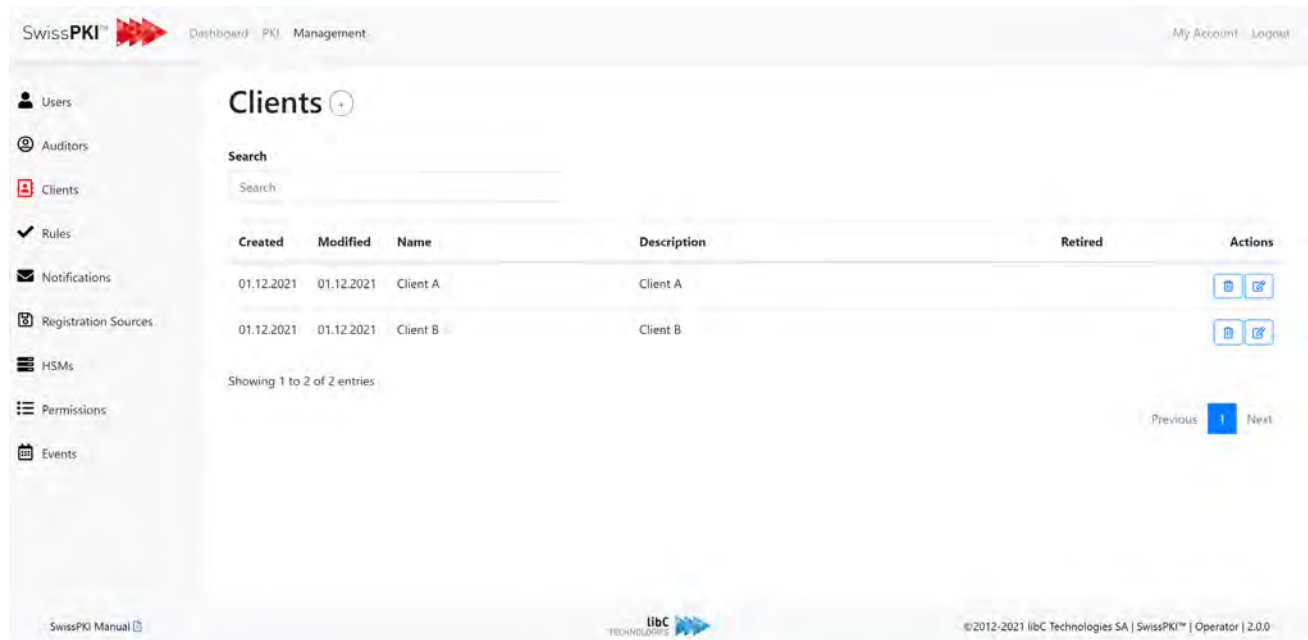
## 10. Technical Contacts


List all Client Technical Contacts. The technical contacts are also notified when DNS Owner Check tokens are sent to the constructed postmaster email addresses (refer to 8.2.2.3 *Constructed Email to Domain Contact*)

## 11. CMC S/N

When CMC is enabled, lists all Client certificate serial numbers authorized to issue, revoke, and search certificates (of type policy type CMC) via CMC Client.





To create a new client, click on the add button located at the right of the page title. Additionally, you can edit or delete each client in the list by clicking on the buttons in the action's column of the table.



SwissPKI  Dashboard PKI Management My Account Logout



**Clients** +

Search

Created	Modified	Name	Description	Retired	Actions
01.12.2021	01.12.2021	Client A	Client A		 
01.12.2021	01.12.2021	Client B	Client B		 

Showing 1 to 2 of 2 entries

Previous **1** Next

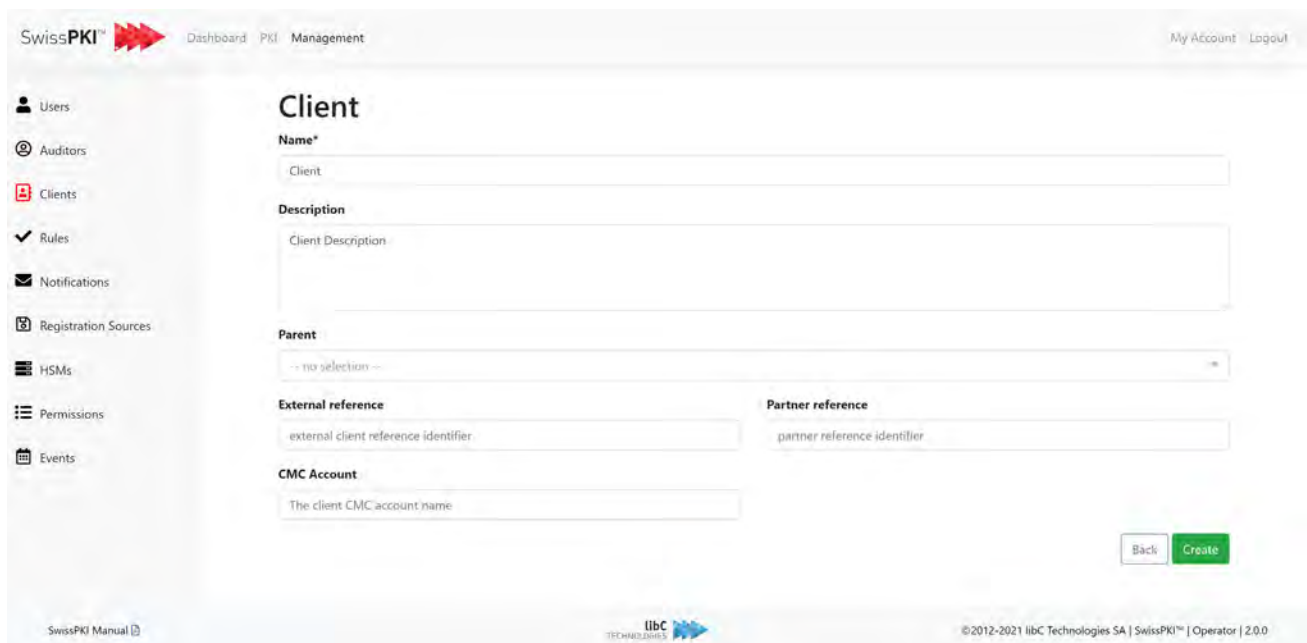
SwissPKI Manual  libC TECHNOLOGIES  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0



### 12.2.3.1 Create Client

After clicking on the add client button, you are redirected to a form. Please fill the following fields and click on the create button to confirm.

Fields	Description
<b>Name</b>	The client's logical name
<b>Description</b>	The client's description
<b>Parent</b>	A parent client can be assigned when creating a new client
<b>External reference</b>	The client's external reference
<b>Partner reference</b>	The client's partner reference
<b>CMC Account</b>	Optional CMC account when CMC option is enabled



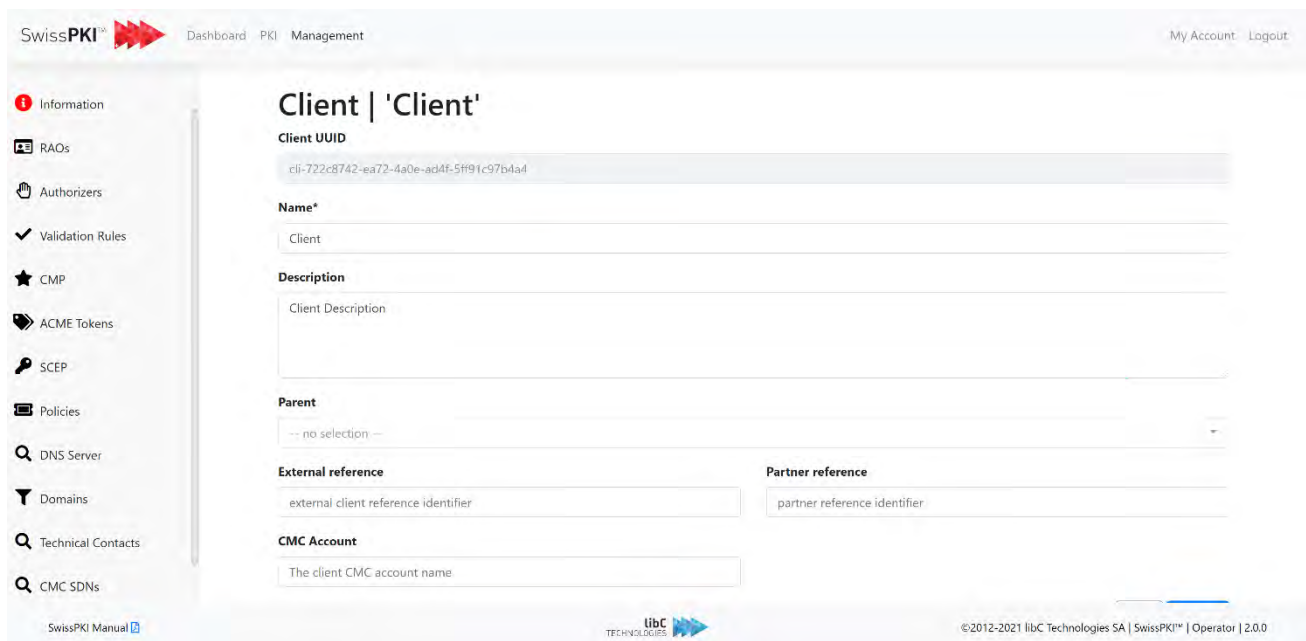
The screenshot shows the 'Client' creation form in the SwissPKI interface. The form is titled 'Client' and contains the following fields:

- Name\***: A text input field with the placeholder 'Client'.
- Description**: A text area with the placeholder 'Client Description'.
- Parent**: A dropdown menu with the placeholder '-- no selection --'.
- External reference**: A text input field with the placeholder 'external client reference identifier'.
- Partner reference**: A text input field with the placeholder 'partner reference identifier'.
- CMC Account**: A text input field with the placeholder 'The client CMC account name'.

At the bottom right of the form, there are two buttons: 'Back' and 'Create'.

### 12.2.3.2 Edit Client

After clicking on the edit button, you are redirected to the client's information page. There, you can update the information you entered during the creation process. Additionally, you can access the different client modules with the side navigation on the left. Each of these modules will be detailed in the next chapters of the documentation.



The screenshot shows the 'Client | 'Client'' edit page in the SwissPKI interface. The page has a header with 'SwissPKI' and 'Dashboard PKI Management' and a user menu with 'My Account' and 'Logout'. A left sidebar contains navigation items: Information, RAOs, Authorizers, Validation Rules, CMP, ACME Tokens, SCEP, Policies, DNS Server, Domains, Technical Contacts, and CMC SDNs. The main content area is titled 'Client | 'Client'' and contains the following fields:

- Client UUID:** cli-722c8742-ed72-4a0e-ad4f-5ff91c97b4d4
- Name\*:** Client
- Description:** Client Description
- Parent:** -- no selection --
- External reference:** external client reference identifier
- Partner reference:** partner reference identifier
- CMC Account:** The client CMC account name

At the bottom, there is a footer with 'libC TECHNOLOGIES' and '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.2.3.3 Delete Client

Deleting a Client will

1. Delete the Client from the DB if the Client has no issued certificate
2. Retire the Client if the Client has issued certificates
3. Disable all access/login for the Client

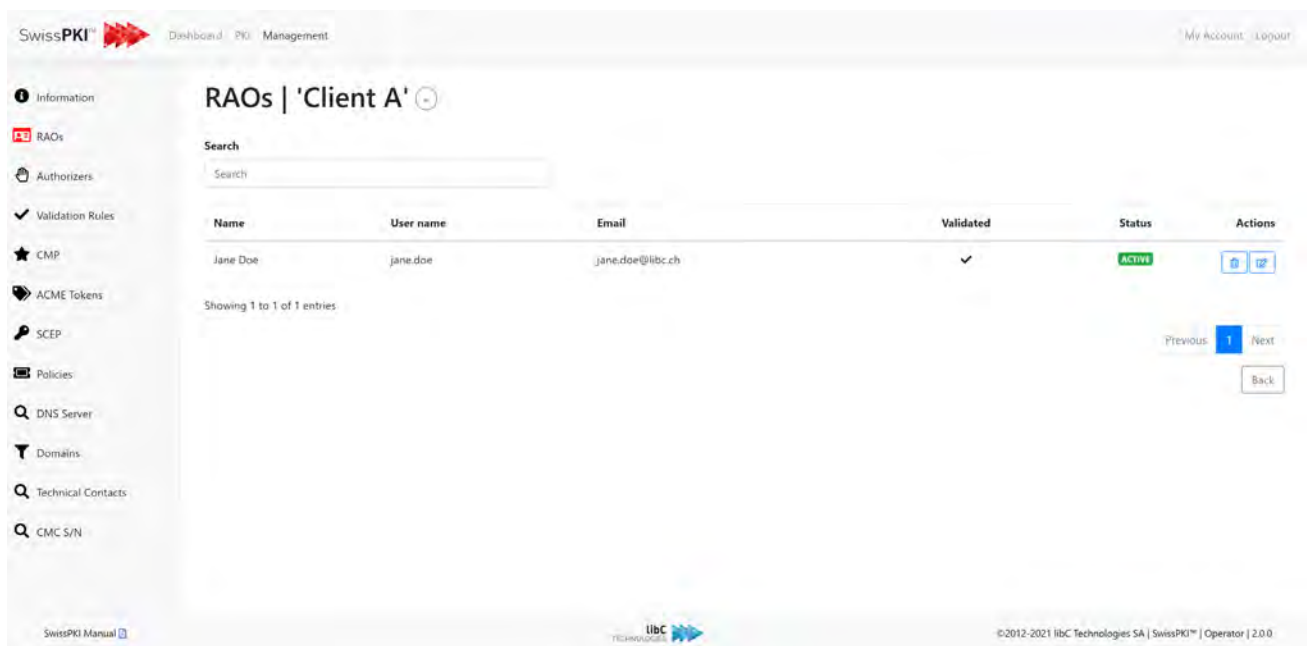
Whether the Client is deleted or retired, then

1. All associated roles (RAO and Authorizer) are removed from the Client
2. All associated certificate mappings are removed from the Client
  - a. If a Client has issued certificates, then the certificate mapping to the certificate policy instance is marked as retired.
  - b. If a Client has no issued certificates, then the certificate mapping to the certificate policy instance is deleted.

### 12.2.3.4 RAOs

Assign or remove Registration Officer roles between the Client and the Realm users.

- Assigning a RAO to the Client will grant access to the Client’s certificate management. The selected user will have access to the Client in the Registration UI. If the user is a SERVICE ACCOUNT, then only REST API (if enabled) is granted
- Removing an RAO will remove the user RAO access to the Client. The selected user will not have access to the Client in the Registration UI.



The screenshot shows the 'RAOs | 'Client A'' management page in the SwissPKI interface. The page includes a search bar, a table of RAOs, and a sidebar with navigation options.

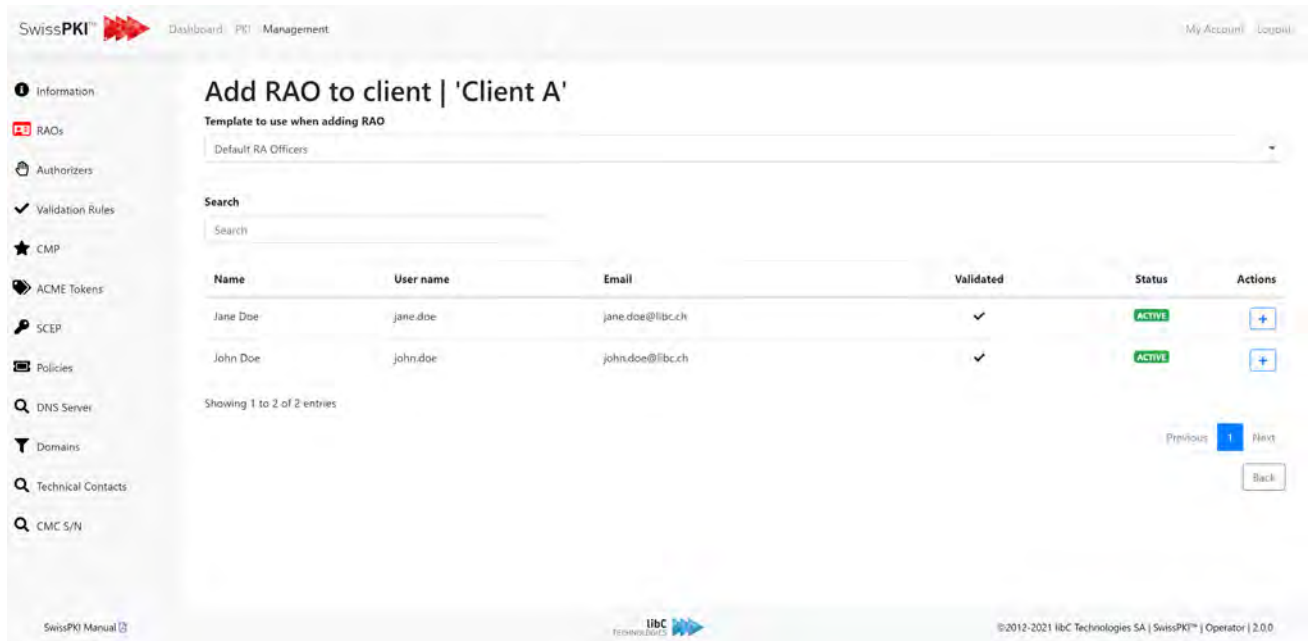
Name	User name	Email	Validated	Status	Actions
Jane Doe	jane.doe	jane.doe@libc.ch	✓	ACTIVE	[Edit] [Delete]

Showing 1 to 1 of 1 entries

Navigation: Previous | 1 | Next | Back

### 12.2.3.4.1 Add RAO

Adding a new RAO is done by clicking on the add button located on the right of the page title. You are redirected to a list of available users. Simply click on the add button located in the action column of the desired user.



SwissPKI™ Dashboard: PKI Management My Account | Logout

## Add RAO to client | 'Client A'


Template to use when adding RAO  
Default RA Officers

Search  
Search

Name	User name	Email	Validated	Status	Actions
Jane Doe	jane.doe	jane.doe@libc.ch	✓	ACTIVE	<a href="#">+</a>
John Doe	john.doe	john.doe@libc.ch	✓	ACTIVE	<a href="#">+</a>

Showing 1 to 2 of 2 entries

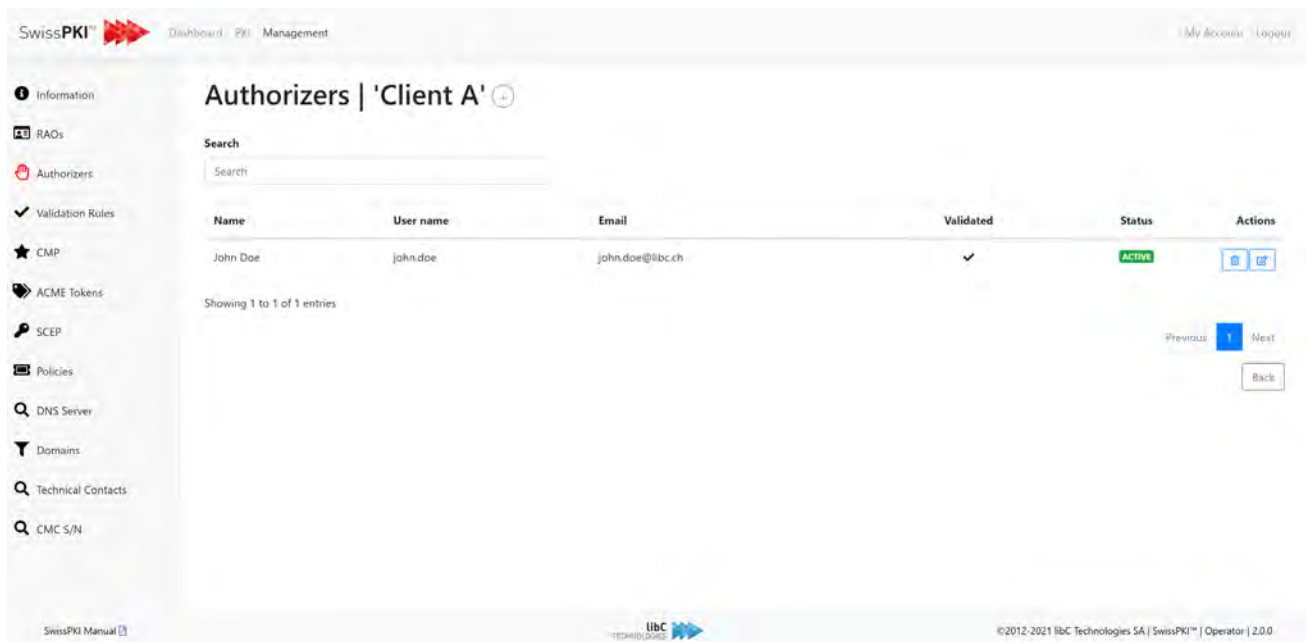
Previous **1** Next  
[Back](#)

SwissPKI Manual  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.3.5 Authorizers

Assign or remove Authorizer roles between the Client and the Realm users.

- Assigning an Authorizer to the Client will grant access to the Client’s certificate authorization management. The selected user will have access to the Client in the Registration UI. If the user is a SERVICE ACCOUNT, then only REST API (if enabled) is granted
- Removing an Authorizer will remove the user’s Authorizer access to the Client. The selected user will not have access to the Client in the Registration UI.
- 



The screenshot shows the 'Authorizers | 'Client A'' page in the SwissPKI management console. The page includes a search bar, a table of authorizers, and navigation controls. The table contains one entry for 'John Doe' with a validated status and an active status.

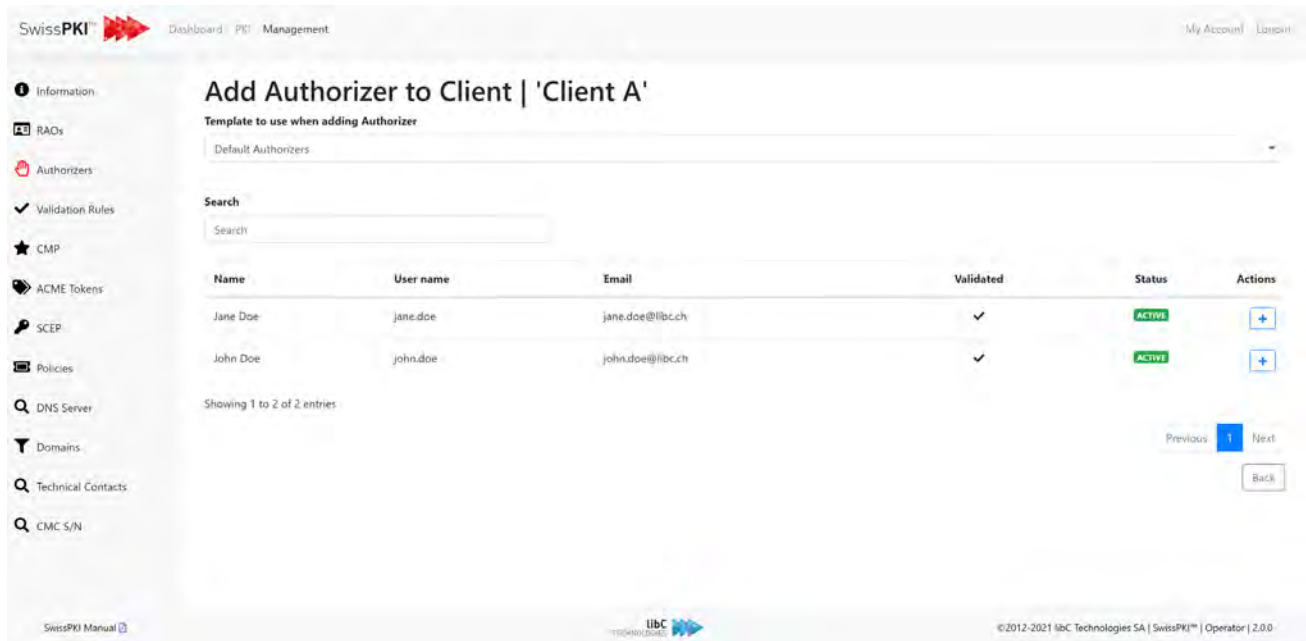
Name	User name	Email	Validated	Status	Actions
John Doe	john.doe	john.doe@libc.ch	✓	ACTIVE	<a href="#">Edit</a> <a href="#">Delete</a>


Showing 1 to 1 of 1 entries

Navigation: Previous 1 Next, Back

### 12.2.3.5.1 Add Authorizers

Adding a new authorizer is done by clicking on the add button located on the right of the page title. You are redirected to a list of available users. Simply click on the desired user's add button.



SwissPKI  Dashboard PKI Management My Account Logout

## Add Authorizer to Client | 'Client A'


Template to use when adding Authorizer  
Default Authorizers

Search  
Search

Name	User name	Email	Validated	Status	Actions
Jane Doe	jane.doe	jane.doe@libc.ch	✓	ACTIVE	+
John Doe	john.doe	john.doe@libc.ch	✓	ACTIVE	+

Showing 1 to 2 of 2 entries

Previous **1** Next  
Back

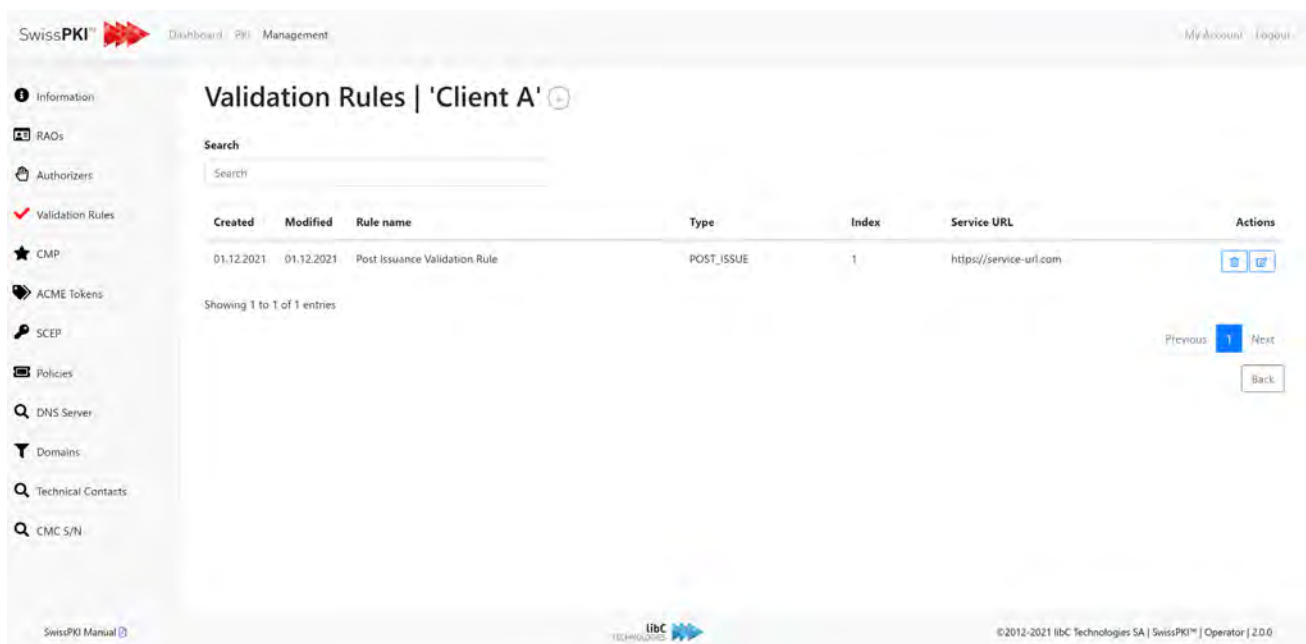
SwissPKI Manual  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.3.6 Client Validation Rules

Validation rules are used to validate the content of a certificate. There are two types of validation rules:

- Pre Validation
- Post Validation

To create a new validation rule, click on the add button located next to the title. Additionally, you can edit or delete an existing validation rule by clicking on the buttons in the table's actions column.



The screenshot shows the 'Validation Rules | Client A' page in the SwissPKI Management interface. The page includes a search bar, a table with one entry, and navigation controls.

Created	Modified	Rule name	Type	Index	Service URL	Actions
01.12.2021	01.12.2021	Post Issuance Validation Rule	POST_ISSUE	1	https://service-url.com	[Edit] [Delete]

Showing 1 to 1 of 1 entries

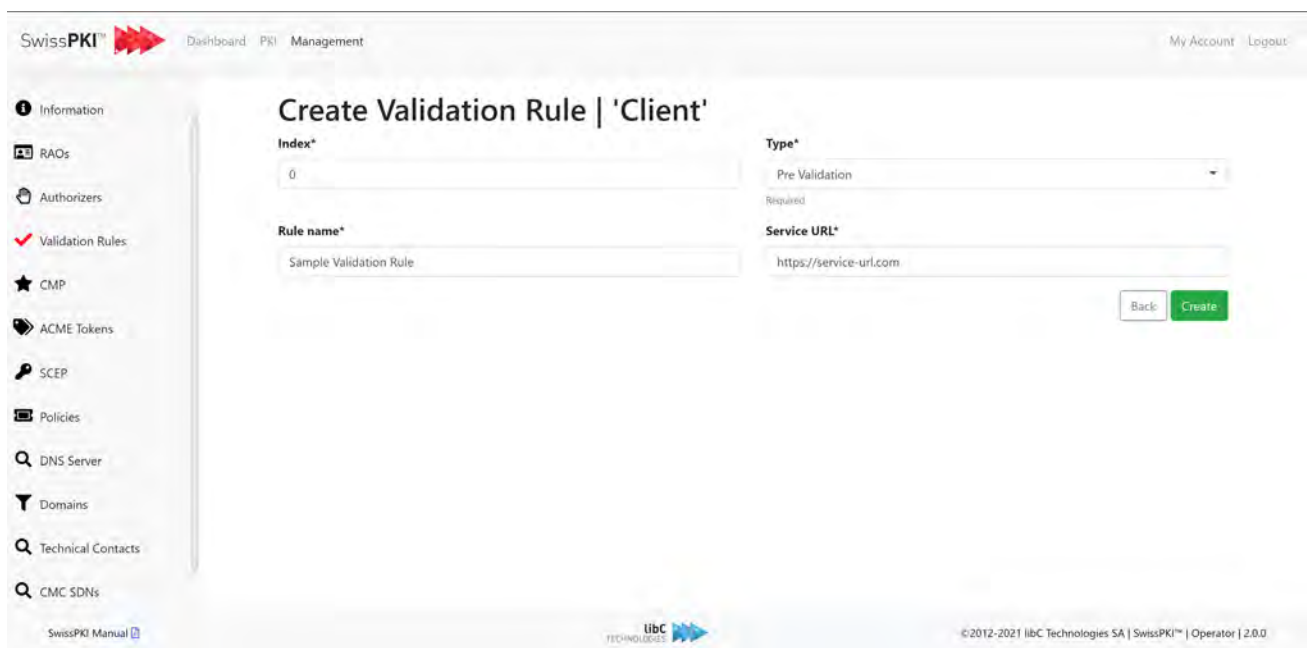
Navigation: Previous 1 Next Back

**Note:** for external validation services implementation, please contact [support@swisspki.com](mailto:support@swisspki.com).

### 12.2.3.6.1 Create Client Validation Rule

To create a new validation rule, simply fill the fields described below and click on the create button at the bottom of the page.

Fields	Description
<b>Index</b>	Validation rule index
<b>Type</b>	Validation rule type <ul style="list-style-type: none"> <li>• Pre validation</li> <li>• Post validation</li> </ul>
<b>Rule name</b>	Validation rule logical name
<b>Service URL</b>	Your defined service HTTPS URL



The screenshot shows the 'Create Validation Rule | 'Client'' page in the SwissPKI interface. The page includes a sidebar with navigation options like Information, RAOs, Authorizers, Validation Rules (highlighted), CMP, ACME Tokens, SCEP, Policies, DNS Server, Domains, Technical Contacts, and CMC SDNs. The main form contains the following fields:

- Index\***: A text input field containing the value '0'.
- Type\***: A dropdown menu with 'Pre Validation' selected. A 'Required' label is present below the dropdown.
- Rule name\***: A text input field containing 'Sample Validation Rule'.
- Service URL\***: A text input field containing 'https://service-url.com'.

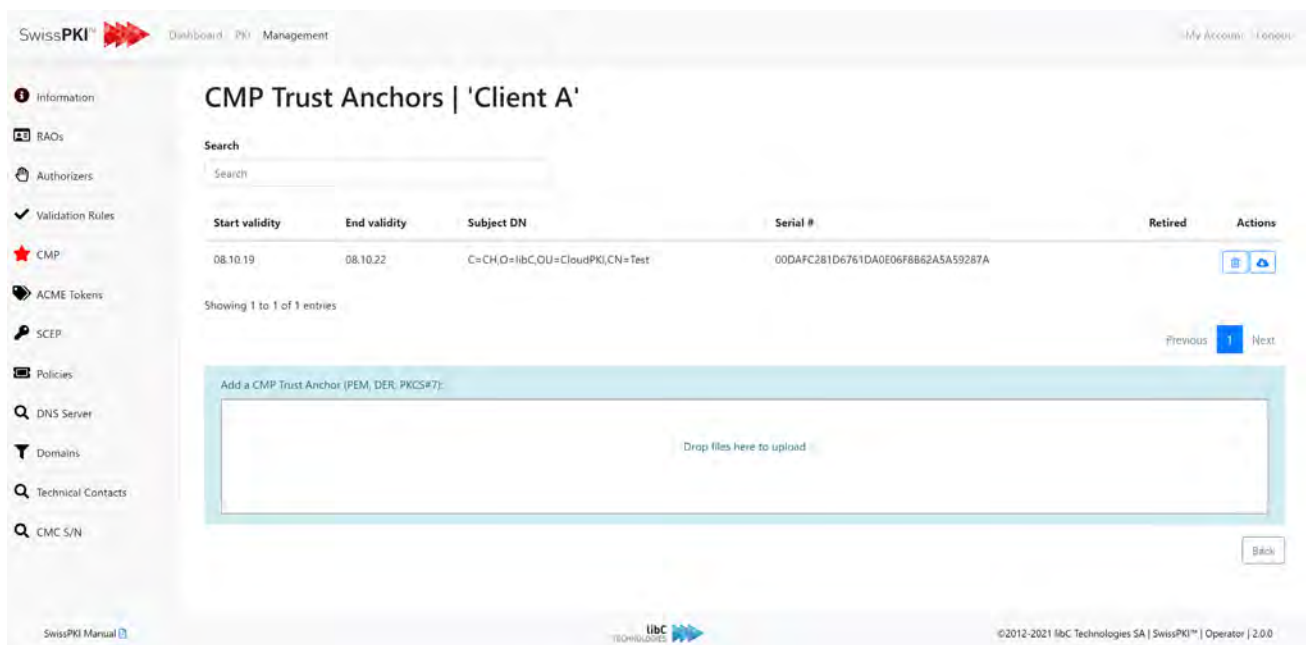
At the bottom right of the form, there are two buttons: 'Back' and 'Create'.





### 12.2.3.7 Certificate Management Protocol

Register Client certificate chain which are used with the CMP client SDK. The certificate chain must contain an end user certificate with the key usage Digital Signature for it to be a valid signing certificate. You can use certificates you issue through SwissPKI or any other end user certificate from another issuing certificate authority. In this case, register the CA trust anchor with your Realm such the validation of the end user requests using the third party issued certificate validate up to the trust anchor.

The uploaded file format must be PKCS#7. Once uploaded, the certificate is ready to use in the CMP Policy Mapping for the Client (please refer to [12.3.1.1.1.2.3 Policy instance mappings](#))



The screenshot shows the SwissPKI web interface. The main heading is 'CMP Trust Anchors | 'Client A''. Below the heading is a search bar. A table displays the trust anchor details:

Start validity	End validity	Subject DN	Serial #	Retired	Actions
08.10.19	08.10.22	C=CH,O=libC,OU=CloudPKI,CN=Test	00DAFC281D6761DA0E06F8B62A5A59287A		 

Below the table, it says 'Showing 1 to 1 of 1 entries'. There are 'Previous' and 'Next' navigation buttons. A large light blue box contains the text 'Add a CMP Trust Anchor (PEM, DER, PKCS#7):' and a drop zone with the text 'Drop files here to upload'. A 'Back' button is located at the bottom right of this box.

### 12.2.3.8 ACME Tokens

List the *PENDING* ACME Tokens Challenges to install on the DNS for the requested domains.

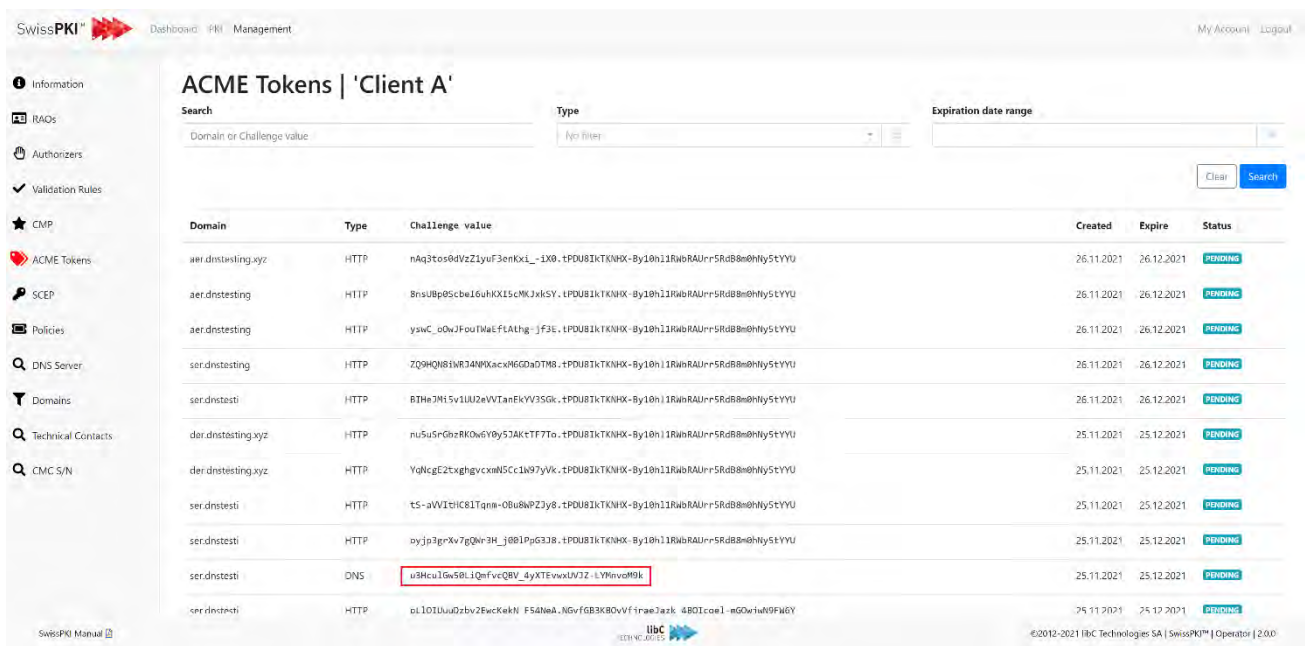
When a Client issues certificates via ACME a challenge token is issued for the Client to install on the DNS server. The requested ACME protocol can occur for http-01 and/or dns-01.

*Example (snipped) of a client ACME request:*

Please deploy a DNS TXT record under the name  
\_acme-challenge.help.libc.ch with the following value:

**u3HculGw50LiQmfvcQBV\_4yXTEvwxUVJZ-LYMnvoM9k**

Before continuing, verify the record is deployed.



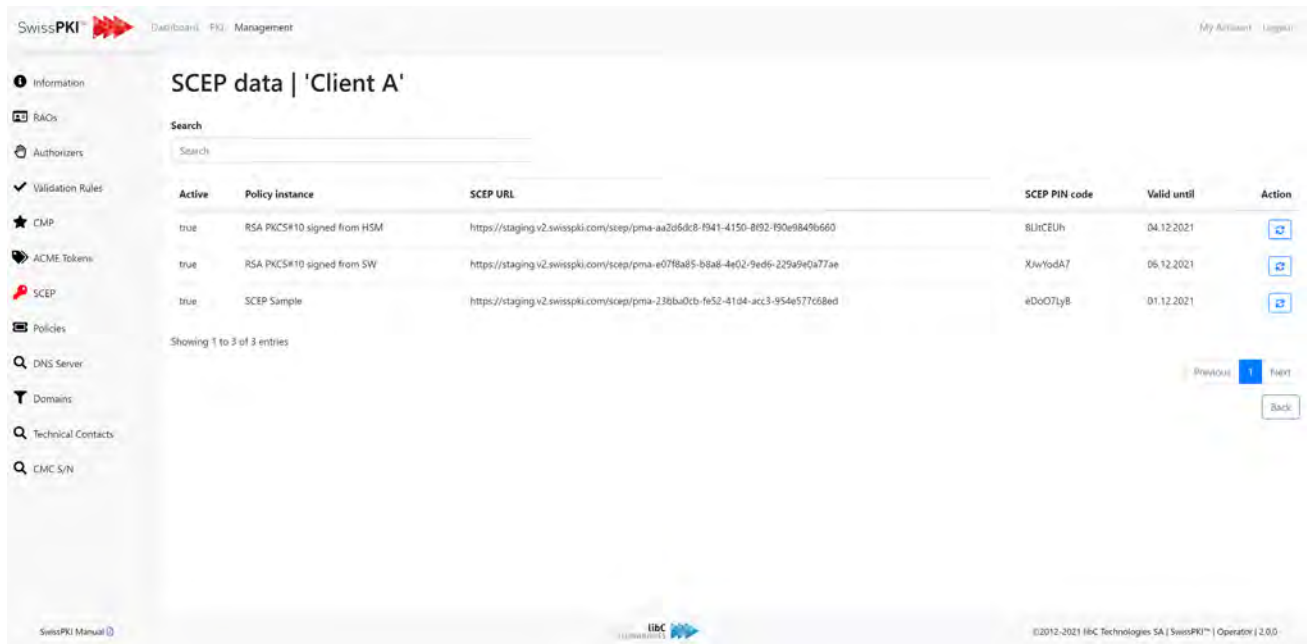
Domain	Type	Challenge value	Created	Expire	Status
aer.dnstesting.xyz	HTTP	nAq3tos8VzZ1yu3enKxL_1x0..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	26.11.2021	26.12.2021	PENDING
aer.dnstesting	HTTP	8nsU8p0Scbe16uHXI5cMKJxkSY..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	26.11.2021	26.12.2021	PENDING
aer.dnstesting	HTTP	yswC_uOwJFouTMeEFLAhg-jf3E..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	26.11.2021	26.12.2021	PENDING
ser.dnstesting	HTTP	ZQ9HQ8fWR34W9KacxN6GDaDT8..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	26.11.2021	26.12.2021	PENDING
ser.dnstesti	HTTP	BThe3H15v1Uu2eVVTaEKyV3Gk..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	26.11.2021	26.12.2021	PENDING
der.dnstesting.xyz	HTTP	nu5uSrGbzRK0vGyBy57AkTf7To..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	25.11.2021	25.12.2021	PENDING
der.dnstesting.xyz	HTTP	YqNcgE2txghgvcmN5Cc1N97yK..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	25.11.2021	25.12.2021	PENDING
ser.dnstesti	HTTP	tS-aVVICi8E1Tqne-0bu8NP23y8..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	25.11.2021	25.12.2021	PENDING
ser.dnstesti	HTTP	oyjpn3rXv7gQw3H_j@1PpG3J8..tPDU81kTKNHX-By10h11RwBRAUrr5Rd88mHly5tYyU	25.11.2021	25.12.2021	PENDING
ser.dnstesti	DNS	<b>u3HculGw50LiQmfvcQBV_4yXTEvwxUVJZ-LYMnvoM9k</b>	25.11.2021	25.12.2021	PENDING
ser.dnstesti	HTTP	6L101Uu0zbv2EecKekN_F54MeA..NGvfGB3K80vVfiraae7axk_4B01coe1-mGDeiaW9FM0Y	25.11.2021	25.12.2021	PENDING

ACME registration URLs are linked to a Client Policy Mapping (see 12.3.1.1.1.2.3 Policy instance mappings).

Note that only PENDING tokens are displayed

### 12.2.3.9 SCEP

When a Client has SCEP certificates policies associated to it, then you list its published SCEP client URLs used for SCEP device registration. The SCEP registration PINs displayed for each URL are valid for a 7 day period before being automatically renewed by the Scheduler.



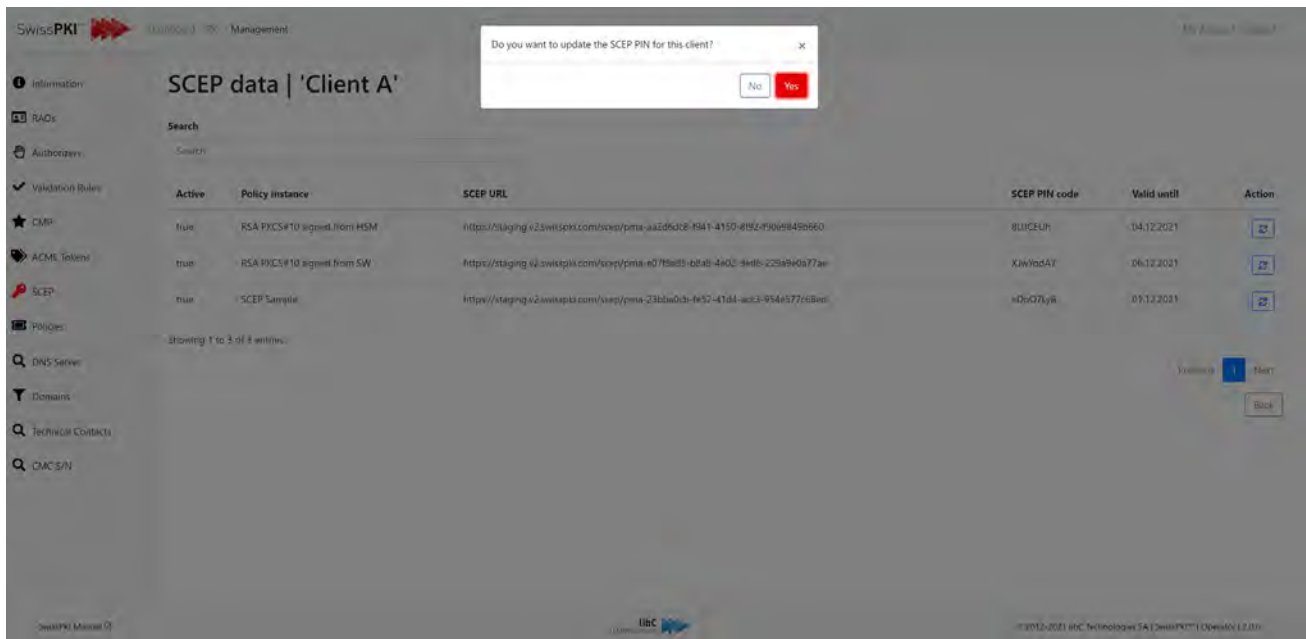
The screenshot shows the 'SCEP data | 'Client A'' page in the SwissPKI management interface. The page includes a search bar and a table with the following data:

Active	Policy instance	SCEP URL	SCEP PIN code	Valid until	Action
true	RSA PKCS#10 signed from HSM	https://staging-v2.swisspki.com/scep/pma-aa2d6dc8-f941-4150-8f92-f50e9849b660	BLHCEU4	04.12.2021	
true	RSA PKCS#10 signed from SW	https://staging-v2.swisspki.com/scep/pma-e07f8a85-b8a8-4e02-9ed6-229a9e0a77ae	XJwYodA7	06.12.2021	
true	SCEP Sample	https://staging-v2.swisspki.com/scep/pma-238ba0cb-fe52-4104-arc3-954e577c68ed	#Dc07ly8	01.12.2021	

Showing 1 to 3 of 3 entries

Navigation: Previous 1 Next Back

As a CA Operator, you can force the renewal of a SCEP PIN by clicking on the 'edit' button.






SwisSPKI Management

Do you want to update the SCEP PIN for this client?

SCEP data | 'Client A'

Search

Active	Policy instance	SCEP URL	SCEP PIN code	Valid until	Action
true	RSA PKCS#10 signwit.from.HSM	https://staging.v2.swissspki.com/scep/pma-aa2d5d32-9341-4150-8f52-f9069459660	8UJCEUH	04.12.2021	
true	RSA PKCS#10 signwit.from.SW	https://staging.v2.swissspki.com/scep/pma-ed7f5a85-bb4b-4402-94b5-22549e0a77ae	XjWvodAT	06.12.2021	
true	SCEP Sample	https://staging.v2.swissspki.com/scep/pma-23bb0d3d-f657-41d4-ac83-954a577e6ba0	xDoGZjyR	03.12.2021	

Showing 1 to 3 of 4 entries.

libC Technologies SA | SwissPKI | OpenSSL 1.1.1g

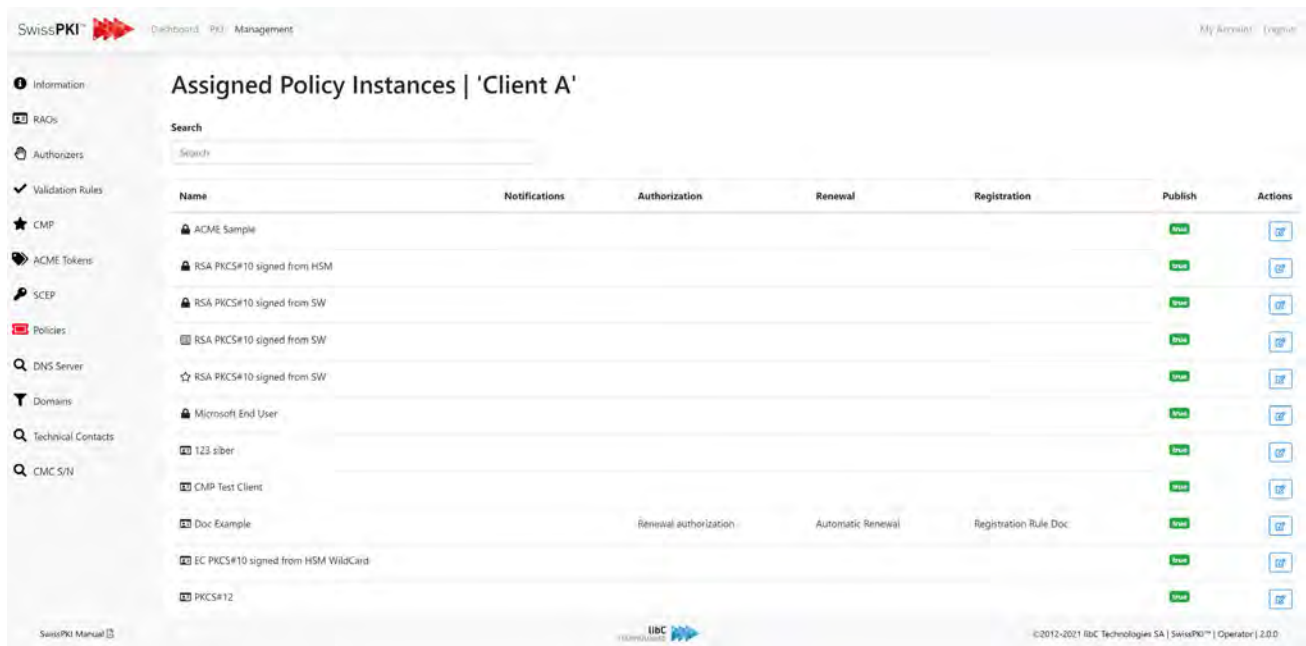
Manually renewing a SCEP PIN will reset the select PIN for a 7 day validity period.

Note: the RA Operator has the identical view for its associated Clients in the Registration UI.

### 12.2.3.10 Client Policies

Display the list of associated Policy Instance (technically named Policy Mappings) with the Client.

Field	Description
<b>Name</b>	Name of the certificate product displayed at the Registration UI
<b>Notifications</b>	List of notification templates associated with the certificate product
<b>Authorizations</b>	Name of the Authorization rule associated with the certificate product
<b>Renewal</b>	Name of the Renewal rule associated with the certificate product
<b>Registration</b>	Name of the Registration rule associated with the certificate product
<b>Publish</b>	Publish the issued certificates to the LDAP
<b>Action</b>	Link to the Policy Mapping settings

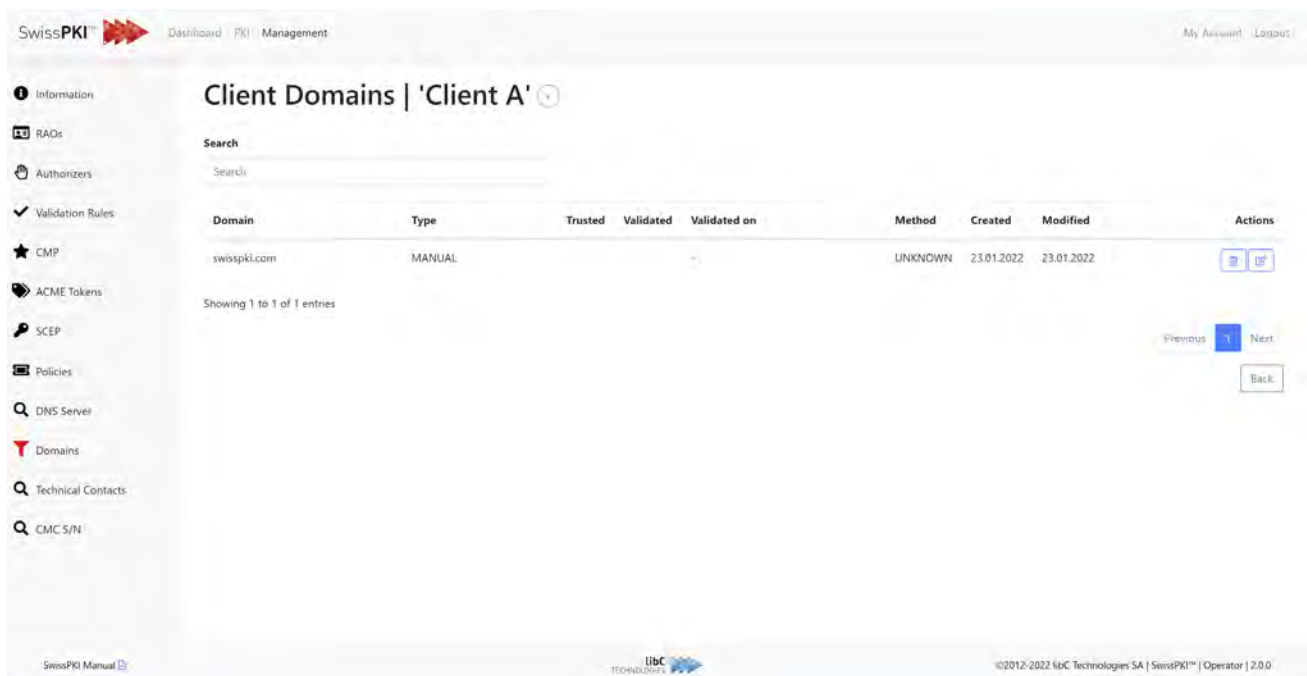


The screenshot shows the SWISSPKI Management interface. The main heading is "Assigned Policy Instances | 'Client A'". Below the heading is a search bar. A table lists the policy instances with columns for Name, Notifications, Authorization, Renewal, Registration, Publish, and Actions. The table contains several rows, including "ACME Sample", "RSA PKCS#10 signed from HSM", "RSA PKCS#10 signed from SW", "Microsoft End User", and "EC PKCS#10 signed from HSM WildCard". Each row has a "Publish" status (green "true") and an "Actions" button. The footer of the interface includes the libC Technologies logo and copyright information: "©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".

### 12.2.3.11 Client Domains

You can create pre-validated domain names such that each issued certificate for the Client is validated against the list of valid domains.

Field	Description
<b>Domain</b>	Acceptable domain name
<b>Type</b>	<p>MANUAL (dns-01 pre-validation only)</p> <p>When created via the Operator or RA UI, the validation is performed by an operator by registering the DNS challenge token with the DNS.</p> <p>If the type is AUTOMATIC, then a DNS challenge is generated and notified to the Client on the fly while issuing the certificate.</p>
<b>Trusted</b>	<p>Indicated if the domain requires CAB validation</p> <p>For private domain</p>
<b>Validated</b>	Date/time when the record was created
<b>Validated On</b>	Date/time when the challenge was validated by the PKI



The screenshot shows the SwissPKI management interface. The main heading is "Client Domains | 'Client A'". Below the heading is a search bar. A table displays the domain information:

Domain	Type	Trusted	Validated	Validated on	Method	Created	Modified	Actions
swisspki.com	MANUAL				UNKNOWN	23.01.2022	23.01.2022	[Edit] [Delete]

Below the table, it says "Showing 1 to 1 of 1 entries". There are navigation buttons for "Previous", "Next", and "Back". The footer includes "SwissPKI Manual", the libC TECHNOLOGIES logo, and copyright information: "©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".

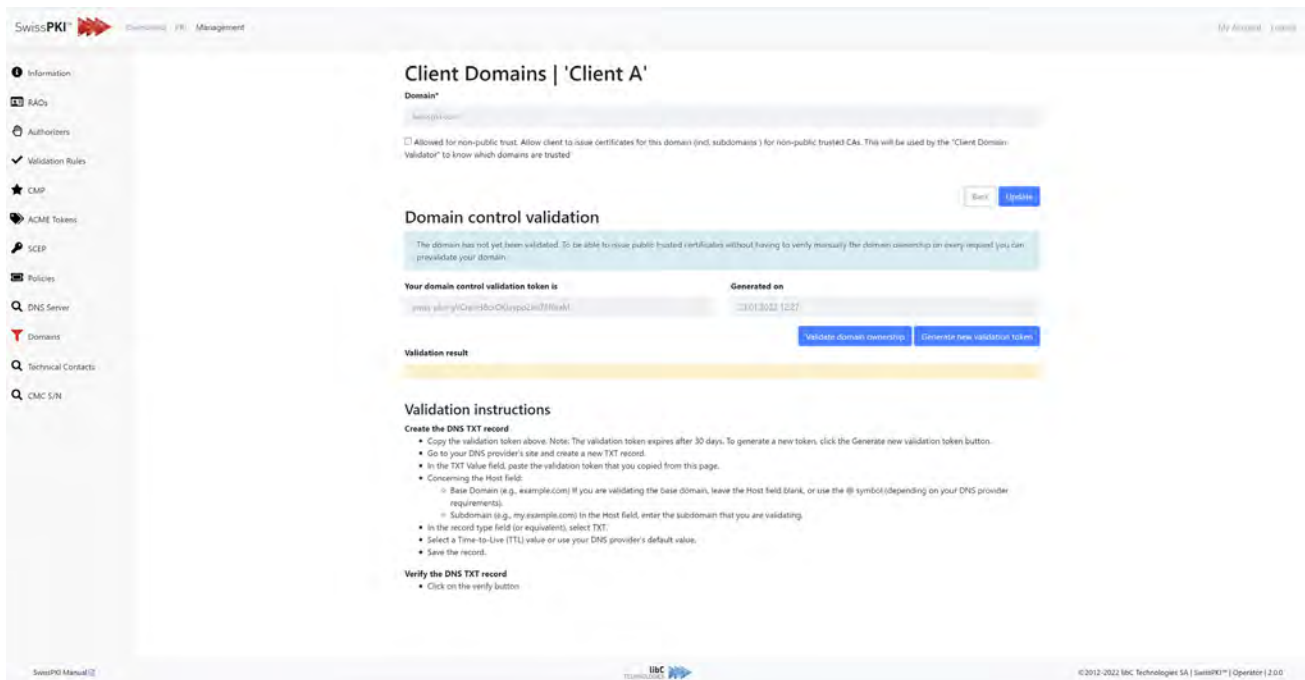
For public trust domain name owner check, set the issuing CA to *'This instance is a Public Trust Certification Authority'* when creating the CA instance. Additionally, define a DNS Owner check rules and map the created rule to the certificate policy template. Issuing CAs with the setting *'This instance is a Public Trust Certification Authority'* enabled will always perform domain name owner check for SSL certificate issuance.

### 12.2.3.11.1 Create Client Domain

Creating a new client domain is done by clicking on the add button located on the right of the client domains' page title. After clicking on the button, you are redirected to a form where you need to provide the following information:

Fields	Description
<b>Domain</b>	Domain name
<b>Allow for non-public trust</b>	<p>Enable/disable the check box.</p> <p>When enabled, the domain is configured as a “trusted domain” for the client. The list of “trusted domains” will be used in the policy validator of type “Client domain validator” (12.3.2.1.4.1.1.2.3)</p> <p>This setting is intended for private PKIs which would like to add a restriction on the domains which can be issued.</p> <p>Note: Even if the checkbox is enabled, you still need to validate the domain using a CAB DNS change to be able to use it as a pre-validated domain in a public trust context. ( Domain owner check is enabled in the policy template )</p>





For public trust DNS Owner check, copy the DNS challenge token to the DNS server as defined in the instructions displayed on the page. As an operator, you can optionally manually validate the DNS entry by clicking on 'Validate domain ownership.'

Click on 'Generate new validation token' to generate a new challenge. Follow the instructions displayed on the screen for the domain you created. The token is valid 30 days. After this period, a new token must be generated.

### 12.2.3.11.2 Client Domains Notifications

Distinct types of notifications are sent during the DNS validation process:

#### System Notifications

- A notification is sent to the RAO x days before certificate expiration.
- A notification is sent to the client's technical contact when the CAB is constructed.
- A notification is sent to the selected end user when the DNS is validated.

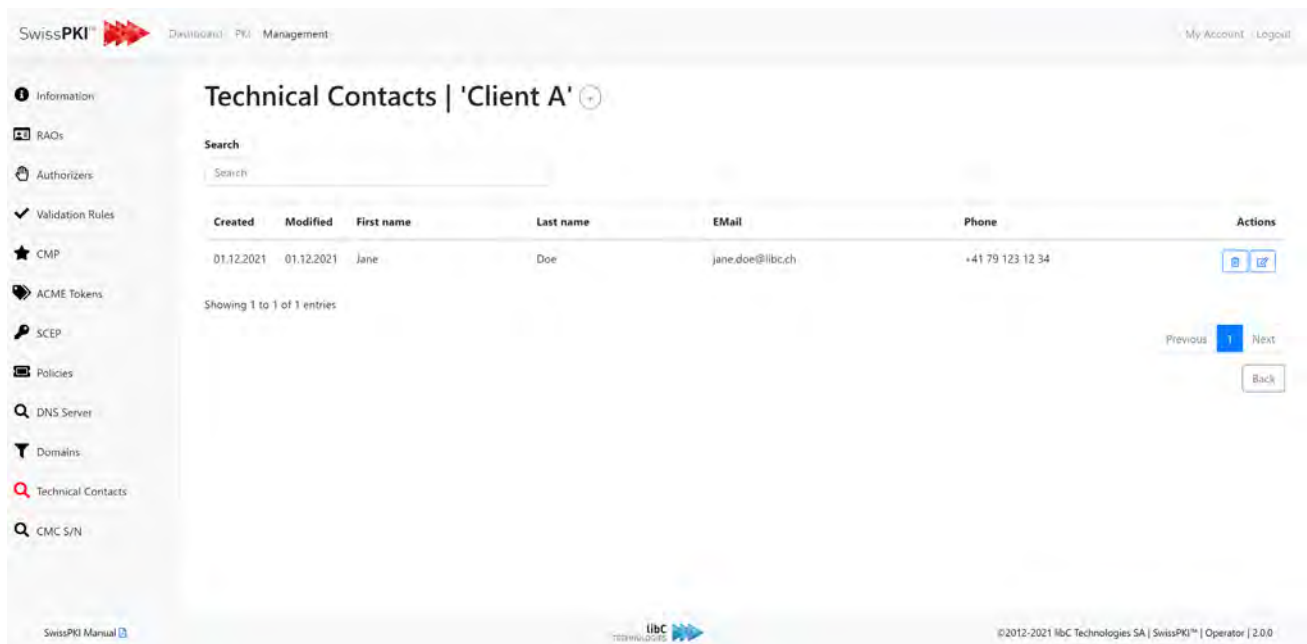
#### Custom notifications

- A custom notification can be assigned for the CAB DNS change.
- A custom notification can be assigned for the CAB Agreed-upon change to website v2.



More information on how to configure these notifications is found in the chapter covering DNS owner check rules. (12.2.4.5

### 12.2.3.12 Technical Contact

List all Client Technical Contacts. The technical contacts are also notified when DNS Owner Check tokens are sent to the constructed postmaster email addresses (refer to 8.2.2.3 *Constructed Email to Domain Contact* ). Additionally, Technical contacts also receive notifications about DNS expiration, renewal, and validation.



The screenshot shows the SwissPKI Management interface. The main content area is titled 'Technical Contacts | 'Client A'' and features a search bar. Below the search bar is a table with the following data:

Created	Modified	First name	Last name	E-Mail	Phone	Actions
01.12.2021	01.12.2021	Jane	Doe	jane.doe@libc.ch	+41 79 123 12 34	 

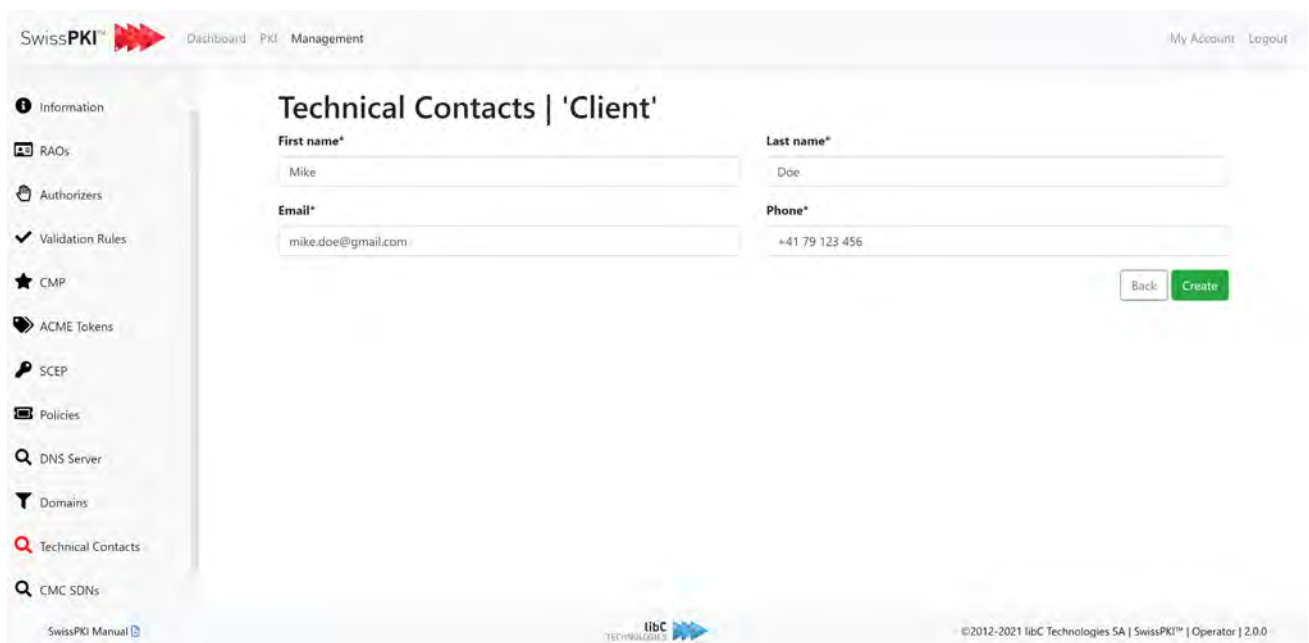
Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom right of the table area, there are navigation buttons: 'Previous', '1', 'Next', and a 'Back' button.

The footer of the interface includes the libC Technologies logo, the text '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0', and a link to the 'SwissPKI Manual'.

### 12.2.3.12.1 Create Technical Contact

To create a new technical contact, complete the following fields and click on the create button at the bottom of the page.

Fields	Description
<b>First Name</b>	The technical contact first name
<b>Last Name</b>	The technical contact's last name
<b>Email</b>	The technical contact email address
<b>Phone</b>	The technical contact phone number



The screenshot shows the 'Technical Contacts | 'Client'' form in the SwissPKI interface. The form includes the following fields:

- First name\***: Input field containing 'Mike'
- Last name\***: Input field containing 'Doe'
- Email\***: Input field containing 'mike.doe@gmail.com'
- Phone\***: Input field containing '+41 79 123 456'

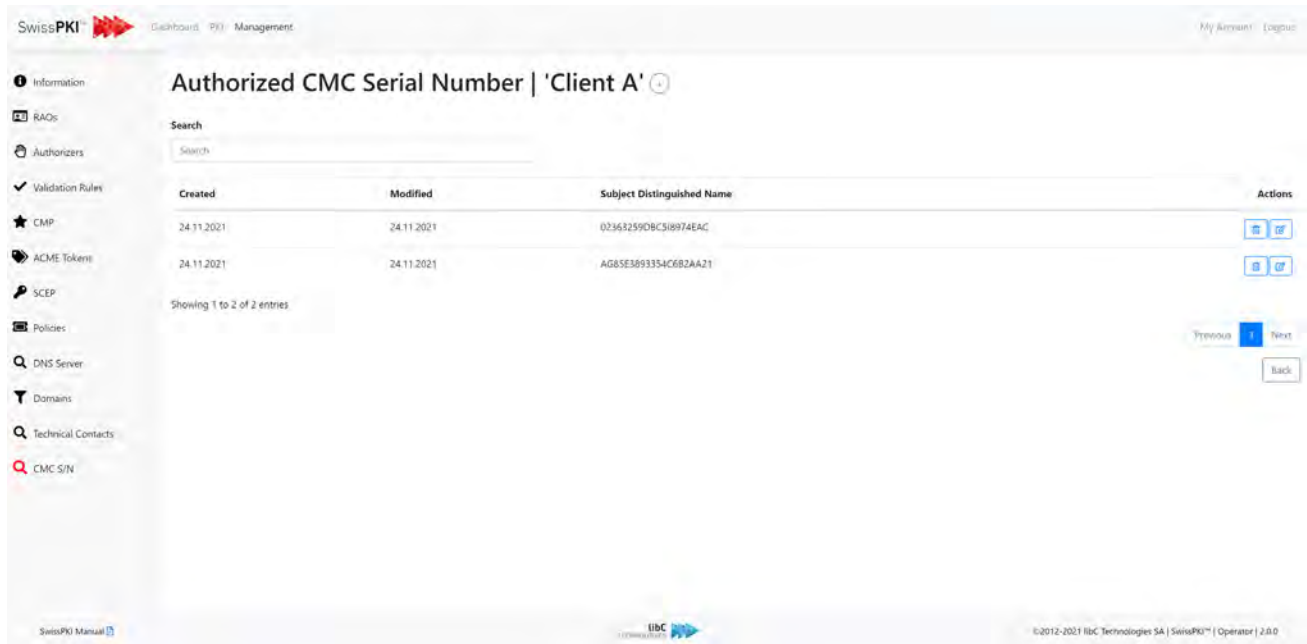
At the bottom right of the form, there are two buttons: 'Back' and 'Create'.

The interface also features a sidebar menu on the left with options like Information, RAOs, Authorizers, Validation Rules, CMP, ACME Tokens, SCEP, Policies, DNS Server, Domains, Technical Contacts, and CMC SDNs. The top navigation bar includes 'SwissPKI', 'Dashboard', 'PKI Management', 'My Account', and 'Logout'.

### 12.2.3.13 Client CMC Serial Number

When CMC is enabled, lists all Client certificate serial numbers authorized to issue, revoke, and search certificates (of type policy type CMC) via CMC Client.

Note that the Client CMC Account name **MUST** also be present.



The screenshot shows the SWISSPKI management interface. The main content area is titled "Authorized CMC Serial Number | 'Client A'". Below the title is a search bar. A table displays the following data:

Created	Modified	Subject Distinguished Name	Actions
24.11.2021	24.11.2021	02365259DBC518974EAC	[Icons]
24.11.2021	24.11.2021	AG85E3893354C682AA21	[Icons]

Below the table, it says "Showing 1 to 2 of 2 entries". Navigation buttons for "Previous", "Next", and "Back" are visible. The footer includes "SwissPKI Manual", the libC logo, and "©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".

## 12.2.4 Rules

Rules are workflow elements to regulate the flow of issuance, renewals, recoveries, and authorizations.

You define these rules independently. In principle, they are linked to predefined notifications. Once a rule is defined, it can be associated to any Client certificate policy mapping (certificate product). Assigning rules to a policy instance for a Client will get triggered during the processing of the rule.

Rules are separated in the following categories:

1. Registration Rules  
Enforce document registration (i.e., copy of ID) during certificate registration
2. Authorization Rules  
Enforce authorization when issuing, revoking, renewing, or recovering certificates
3. Renewal Rules  
Enforce automatic or manual certificate renewal and notifications
4. CAA Rules  
Enforce CAA check when issuing certificates
5. DNS Owner Check Rules  
Enforce DNS Owner Check when issuing certificates
6. CT Rules  
Enforce CT log publication when issuing certificates

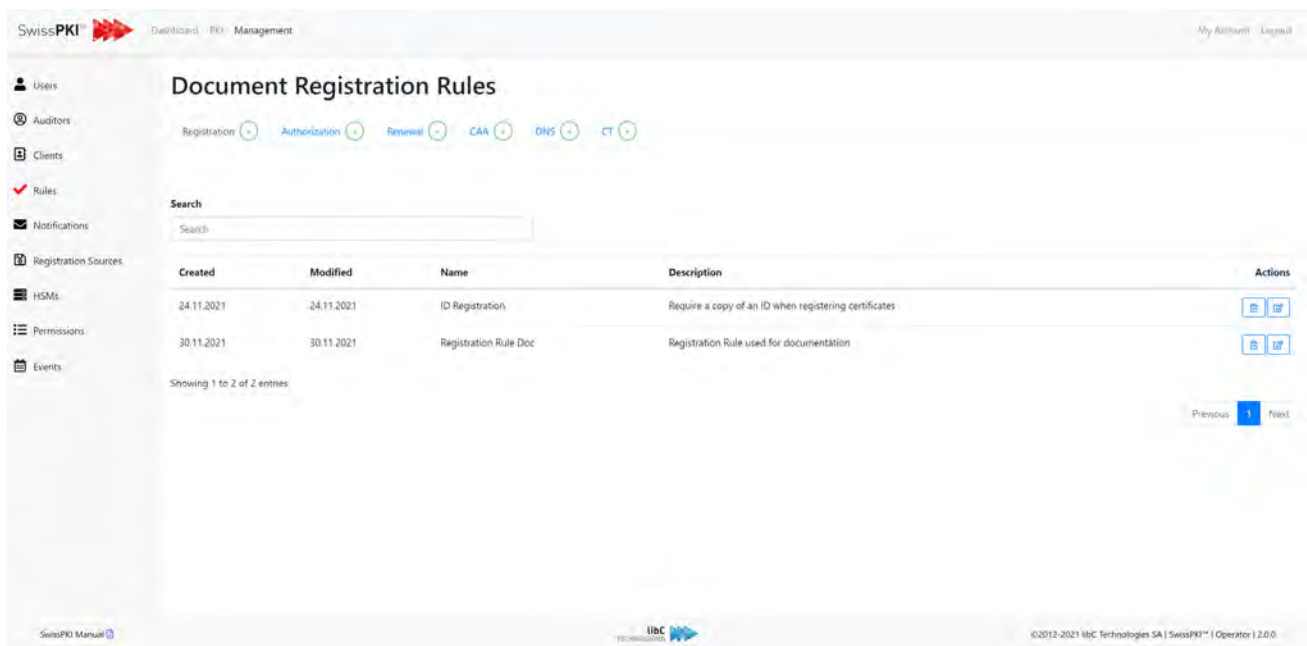
### 12.2.4.1 Registration Rules

Registration rules are rules that are applied during the workflow of issuing certificates with specific policy for a given realm. It forces the registration of documents related to the issuance.





These rules allow RAOs to collect information related to the issuance process in the form of PDF documents or images. These rules may also be enforced at the time of issuance when the RAO is required to enter or provide documents before the certificate is issued. Once these documents have been provided, they are associated with the certificate that has been issued and can be downloaded or corrected when searching for user-related or system-specific certificates.

In general, registration rules are documents that are collected and linked to a process that must be certified, such as the issuance of qualified certificates.

Registration Rules are linked to a Client Policy Mapping (see *12.3.1.1.1.2.3 Policy instance mappings*).



The screenshot shows the 'Document Registration Rules' page in the SwissPKI Management interface. The page includes a navigation sidebar on the left with options like Users, Auditors, Clients, Rules (selected), Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area features a breadcrumb trail: Registration > Authorization > Renewal > CAA > DNS > CT. Below this is a search bar and a table of rules.

Created	Modified	Name	Description	Actions
24.11.2021	24.11.2021	ID Registration	Require a copy of an ID when registering certificates	 
30.11.2021	30.11.2021	Registration Rule Doc	Registration Rule used for documentation	 

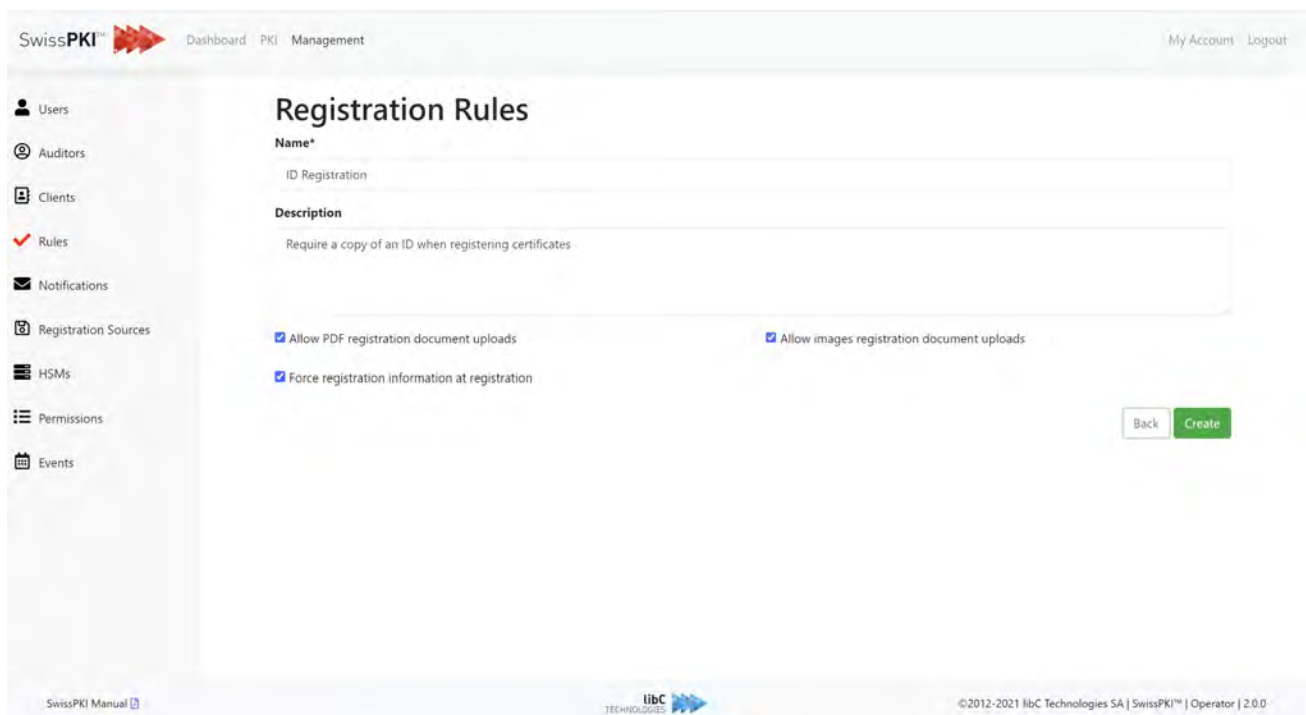
Showing 1 to 2 of 2 entries

Previous 1 Next

### 12.2.4.1.1 Create Registration Rule

Creating a first-time registration rule is done by clicking on the add button located on the right of the rules navigation's registration link. You are redirected to a form where you need to provide the following information:

Fields	Description
<b>Name</b>	The registration rule's name
<b>Description</b>	The registration rule's description
<b>Allow PDF registration document uploads</b>	If checked PDF registration document upload will be allowed
<b>Allow image registration document uploads</b>	If checked image registration document upload will be allowed
<b>Force registration information at registration</b>	If checked, registration information will be forced at registration



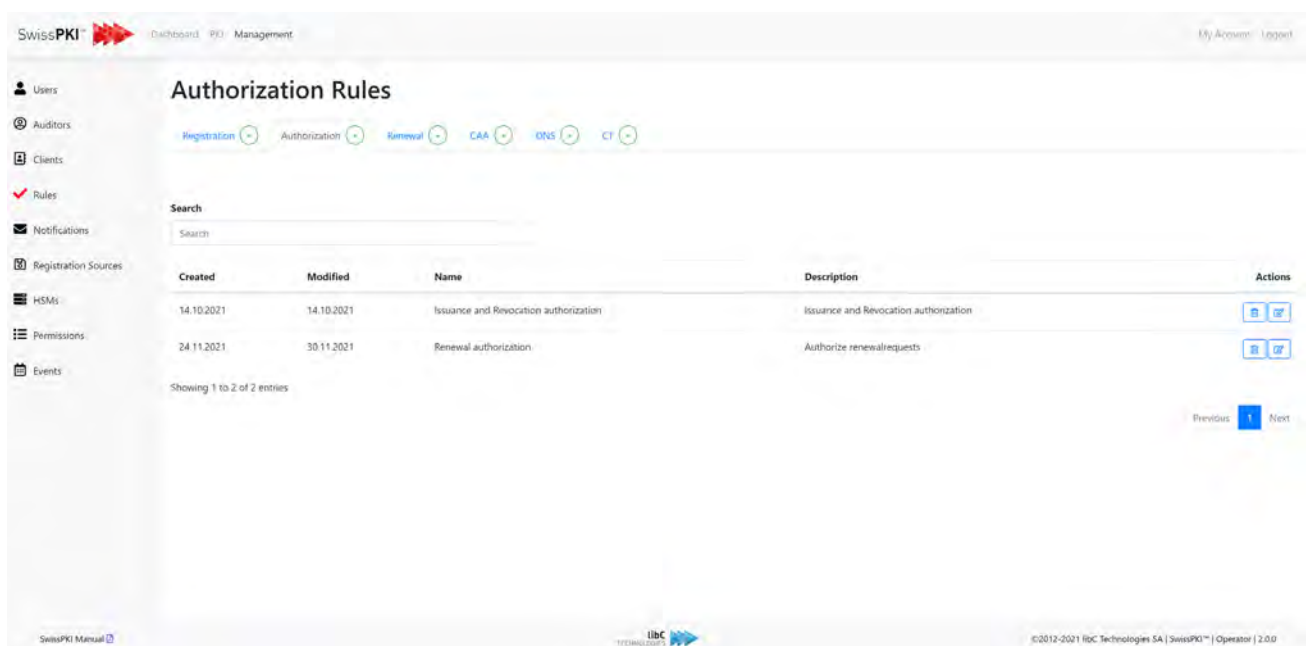
## 12.2.4.2 Authorization Rules

When issuing, revoking, renewing, or recovering certificates, these rules are applied during the workflow to allow an Authorizer to accept or reject any of these requests.

For each of these rules, you can select the different permissions to apply by selecting the check boxes. You optionally associate notifications with each authorization

If an authorization is applied to the issuance of certificates, the workflow creates an authorization element for the group of people who authorize the requests and optionally send messages to the separate roles associated to the notification. In general, a CAO issuing a certificate, for which an authorization is required during issuance, is notified by email if the request is accepted or rejected by the authorizers. The same applies to certificate renewal, key recovery, and certificate revocation.

Authorization Rules are linked to a Client Policy Mapping (see *12.3.1.1.1.2.3 Policy instance mappings*).



The screenshot shows the 'Authorization Rules' management interface. At the top, there are filters for 'Registration', 'Authorization', 'Renewal', 'CAA', 'ONS', and 'CT'. Below the filters is a search bar. The main table contains the following data:

Created	Modified	Name	Description	Actions
14.10.2021	14.10.2021	Issuance and Revocation authorization	Issuance and Revocation authorization	[Edit] [Delete]
24.11.2021	30.11.2021	Renewal authorization	Authorize renewalrequests	[Edit] [Delete]


At the bottom of the table, it says 'Showing 1 to 2 of 2 entries'. There are 'Previous' and 'Next' navigation buttons. The footer of the interface includes 'SwissPKI Manual', the libC Technologies logo, and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.












### 12.2.4.2.1 Create Authorization Rule

Creating a new authorization rule is done by clicking on the add button located in the rule's navigation. After clicking the button, you are redirected to a form where you need to provide the following information:

Fields	Description
<b>Name</b>	The authorization rule's name
<b>Description</b>	The authorization rule's description
<b>Enforce authorization on</b>	When selected, an authorization will be required. This can be activated for the following: <ul style="list-style-type: none"> <li>- Certificate issuance</li> <li>- Certificate renewal</li> <li>- Key recovery</li> <li>- Certificate revocation</li> </ul>
<b>Notification</b>	When the authorization is activated, a notification can be selected.
<b>Authorizer can edit certificate request</b> ( <i>issuance only</i> )	When selected, an authorizer is able to update values of the certificate request before approval
<b>Authorizer can edit SAN fields</b> ( <i>issuance only</i> )	When selected, an authorizer is able to edit SAN-fields (DNS, RFC822) and SDN-fields (CN)
<b>Number of required approvals</b> ( <i>issuance only</i> )	Defines, how many authorizers are required to give their approval before issuance (default: 1)

SwissPKI™  Dashboard PKI Management My Account Logout

-  Users
-  Auditors
-  Clients
-  Rules
-  Notifications
-  Registration Sources
-  HSMs
-  Permissions
-  Events

## Authorization Rules

**Name**  
Authorization Rule

**Description**  
Authorization Rule

Enforce authorization on certificate issuance


Enforce authorization on certificate renewal


Enforce authorization on key recovery










**Notification**  
Authorization Notification

Enforce authorization on certificate revocation

[Back](#) [Create](#)

SwissPKI Manual  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

SwissPKI™  Dashboard PKI Management My Account Logout

-  Users
-  Auditors
-  Clients
-  Rules
-  Notifications
-  Registration Sources
-  HSMs
-  Permissions
-  Events

## Authorization Rules

**Name\***  
multi-auth-rule

**Description**  
Rule for multiple Authorization on issuance

Enforce authorization on certificate issuance

**Notification**  
No selection

Authorizer can edit certificate request

Authorizer can edit SAN fields


**Number of required approvals**  
2

Enforce authorization on certificate renewal

Enforce authorization on key recovery

Enforce authorization on certificate revocation

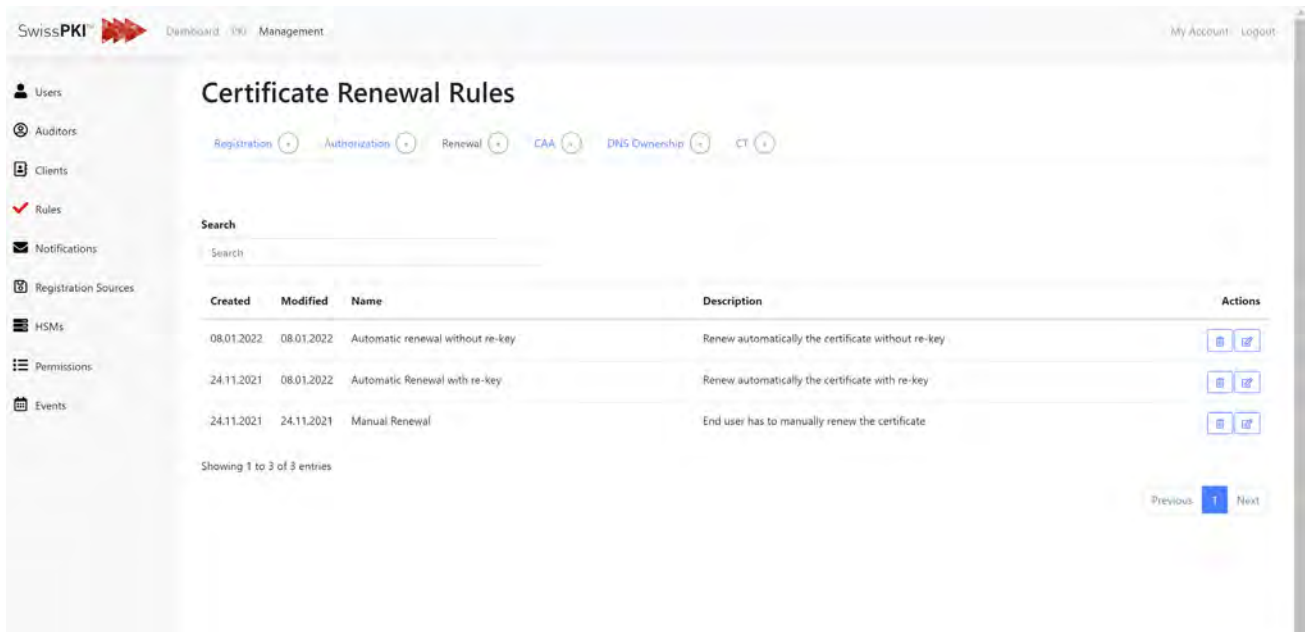
[Back](#) [Update](#)

User manual  ©2012-2023 libC Technologies SA | SwissPKI™ | Operator | 2.3.0







### 12.2.4.3 Renewal Rules

You create Renewal rules to issued certificates for which an automatic or manual renewal is wanted. Based on the certificates' expiration date and you have the possibility, the renewal rule will notify the recipients and optionally re-issue (when automatic renewal is enabled) a certificate or request a new CSR depending on the certificate policy key generation type.

Renewal Rules are linked to a Client Policy Mapping (see *12.3.1.1.1.2.3 Policy instance mappings*).



The screenshot displays the 'Certificate Renewal Rules' management page in the SwissPKI system. The page features a sidebar with navigation options: Users, Auditors, Clients, Rules (selected), Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area shows a table of renewal rules with columns for Created, Modified, Name, Description, and Actions. There are three entries in the table:

Created	Modified	Name	Description	Actions
08.01.2022	08.01.2022	Automatic renewal without re-key	Renew automatically the certificate without re-key	 
24.11.2021	08.01.2022	Automatic Renewal with re-key	Renew automatically the certificate with re-key	 
24.11.2021	24.11.2021	Manual Renewal	End user has to manually renew the certificate	 

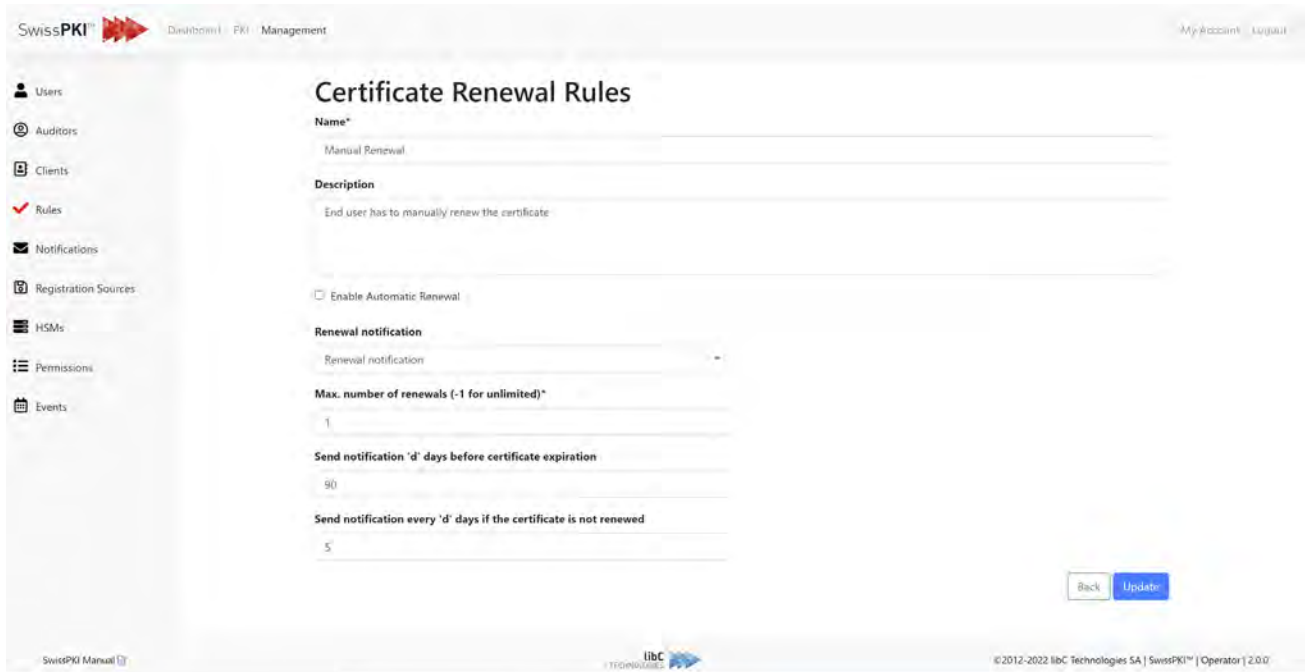
Below the table, it indicates 'Showing 1 to 3 of 3 entries' and includes pagination controls for 'Previous', '1', and 'Next'.


### 12.2.4.3.1 Create Renewal Rule

#### 12.2.4.3.1.1 Manual renewal

A manual renewal rule only sends out notifications to the recipients. The recipient may renew its certificate by requesting a new certificate using a new CSR or reusing the same private key. When reusing the same private key, the Issuing CA must have its setting *'unique public key check'* disabled.

Fields	Description
<b>Name</b>	The renewal rule's name
<b>Description</b>	The renewal rule's description
<b>Enable automatic renewal</b>	When selected, the certificate will be automatically renewed.
<b>Renewal notification</b>	Let us you define which notification is used for certificate renewal.
<b>Max renewal number</b>	The maximum number of authorized renewals for a certificate. Use -1 for unlimited renewal
<b>Send notification 'd' days before certificate expiration</b>	Only available when automatic renewal is disabled. Let us you define how many days before expiration the notification is sent.
<b>Send notification every 'd' days if the certificate is not renewed</b>	Only available when automatic renewal is not selected. Let us you define the interval at which notifications are sent when the certificate is not renewed



SwissPKI™  Dashboard | PKI | Management My Account Logout

**Certificate Renewal Rules**

**Name\***  
Manual Renewal

**Description**  
End user has to manually renew the certificate

Enable Automatic Renewal


**Renewal notification**  
Renewal notification

**Max. number of renewals (-1 for unlimited)\***  
1

**Send notification 'd' days before certificate expiration**  
90

**Send notification every 'd' days if the certificate is not renewed**  
5

[Back](#) [Update](#)

SwissPKI Manual  © 2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

#### 12.2.4.3.1.2 Automatic renewal

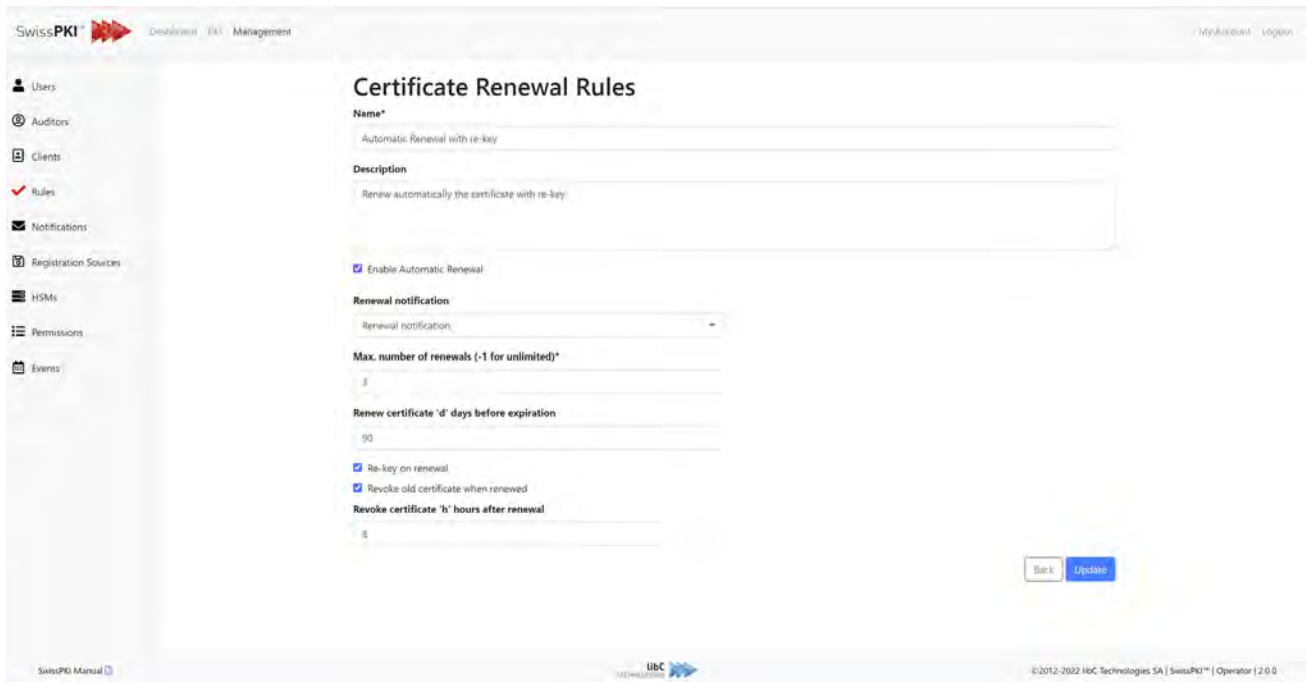
An automatic renewal rule renews the certificate and mail it to the recipient. Depending on the certificate policy key generation type and the Issuing CA *'unique public key constraint'* setting, the renewal behavior will be different.

Key gen.	Public key constraint	Re-key	Behavior
PKCS#12	enabled	yes	Certificate and key automatically issued, and recipients notified
PKCS#12	enabled	no	Renewal will fail during key validation
PKCS#12	disabled	yes	Certificate and key automatically issued, and recipients notified
PKCS#12	disabled	no	Certificate issued using previous key pair and recipients notified.
HSM	enabled	yes	Certificate and key automatically issued, and recipients notified
HSM	enabled	no	Renewal will fail during key validation
HSM	disabled	yes	Certificate and key automatically issued, and recipients notified

<b>HSM</b>	disabled	no	Certificate issued using previous key pair and recipients notified.
<b>PKCS#10</b>	enabled	yes	CSR requested by PKI and notified to recipient. Recipient generates a new CSR and copy/paste's it to the generated Self-service URL in the notification link.
<b>PKCS#10</b>	enabled	no	Renewal will fail during key validation
<b>PKCS#10</b>	disabled	yes	CSR requested by PKI and notified to recipient. Recipient generates a new CSR and copy/paste's it to the generated Self-service URL in the notification link
<b>PKCS#10</b>	disabled	no	Certificate issued using previous key pair and recipients notified.

Fields	Description
<b>Name</b>	The renewal rule's name
<b>Description</b>	The renewal rule's description
<b>Enable automatic renewal</b>	When selected, the certificate will be renewed automatically.
<b>Renewal notification</b>	Let us you define which notification is used for certificate renewal.
<b>Max renewal number</b>	The maximum number of renewals for a certificate. Use -1 for unlimited renewal
<b>Renew certificate 'd' days before expiration</b>	Automatically renew the certificate 'd' days before expiration.
<b>Rekey on renewal</b>	If selected, a new key will be created or requested when the certificate is renewed
<b>Revoke old certificate when renewed</b>	If selected, the old certificate will be revoked after the renewal
<b>Revoke certificate 'h' hours after renewal</b>	Revokes the certificate 'h' hours after successful renewal.

**Note:** if the value is set to '0' then the certificate is immediately revoked after successful renewal



The screenshot shows the 'Certificate Renewal Rules' configuration page in the SwissPKI management interface. The page includes a sidebar with navigation options like Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area is titled 'Certificate Renewal Rules' and contains the following fields and options:

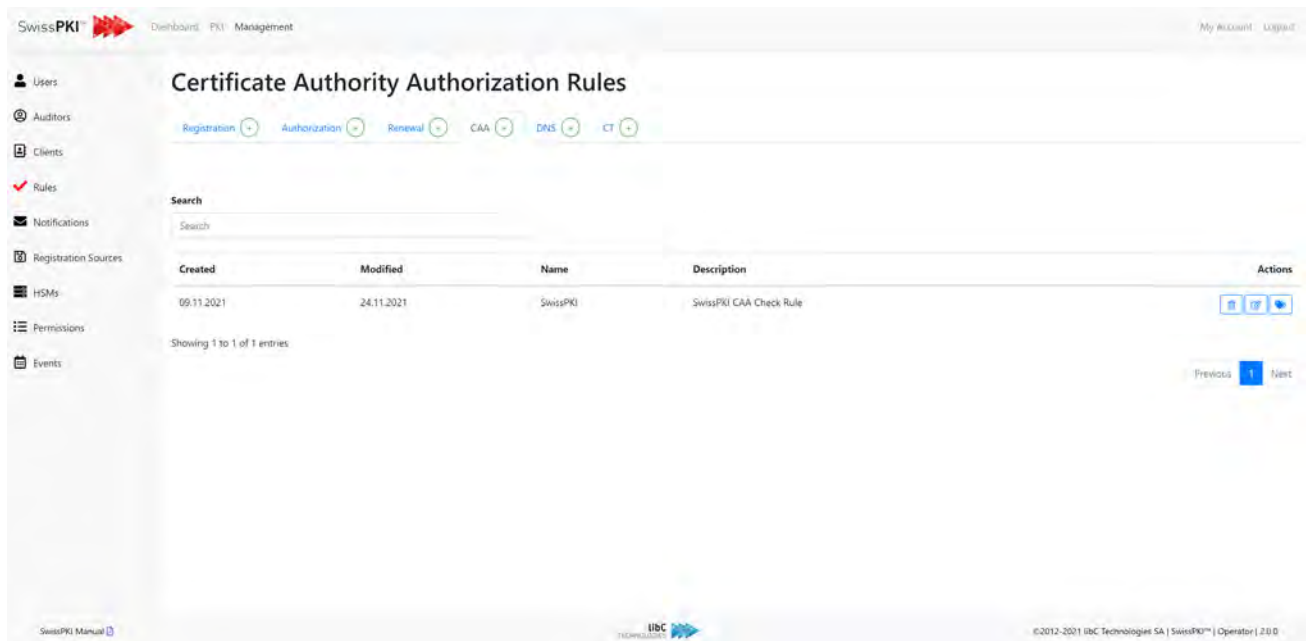
- Name\***: Automatic Renewal with re-key
- Description**: Renew automatically the certificate with re-key
- Enable Automatic Renewal**
- Renewal notification**: Renewal notification
- Max. number of renewals (-1 for unlimited)\***: 3
- Renew certificate 'd' days before expiration**: 90
- Re-key on renewal**
- Revoke old certificate when renewed**
- Revoke certificate 'h' hours after renewal**: 3

At the bottom right of the form, there are 'Back' and 'Update' buttons. The footer of the page includes 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.




#### 12.2.4.4 CAA Rules

CAA is a security measure that allows domain owners to specify in their Domain Name Servers (DNS) which CAs are authorized to issue certificates for that domain. If a CA receives an order for a certificate for a domain with a CAA record and that CA is not listed as an authorized issuer, they are prohibited from issuing the certificate to that domain or any subdomain. This supplements the Certificate Transparency to help domain owners identify mis-issued or frequently issued certificates for their domains after issuance, while CAA can help prevent unauthorized issuance before the fact. Together they build a better set of security than either one by themselves.

CAA Rules are linked to a Policy Template (see *12.3.2 Certificate Policy Templates*).



The screenshot shows the SwissPKI web interface for managing Certificate Authority Authorization Rules. The page title is "Certificate Authority Authorization Rules". There are navigation tabs for Registration, Authorization, Renewal, CAA, DNS, and CT. A search bar is present above a table of rules. The table has columns for Created, Modified, Name, Description, and Actions. One rule is listed with the following details:

Created	Modified	Name	Description	Actions
09.11.2021	24.11.2021	SwissPKI	SwissPKI CAA Check Rule	  

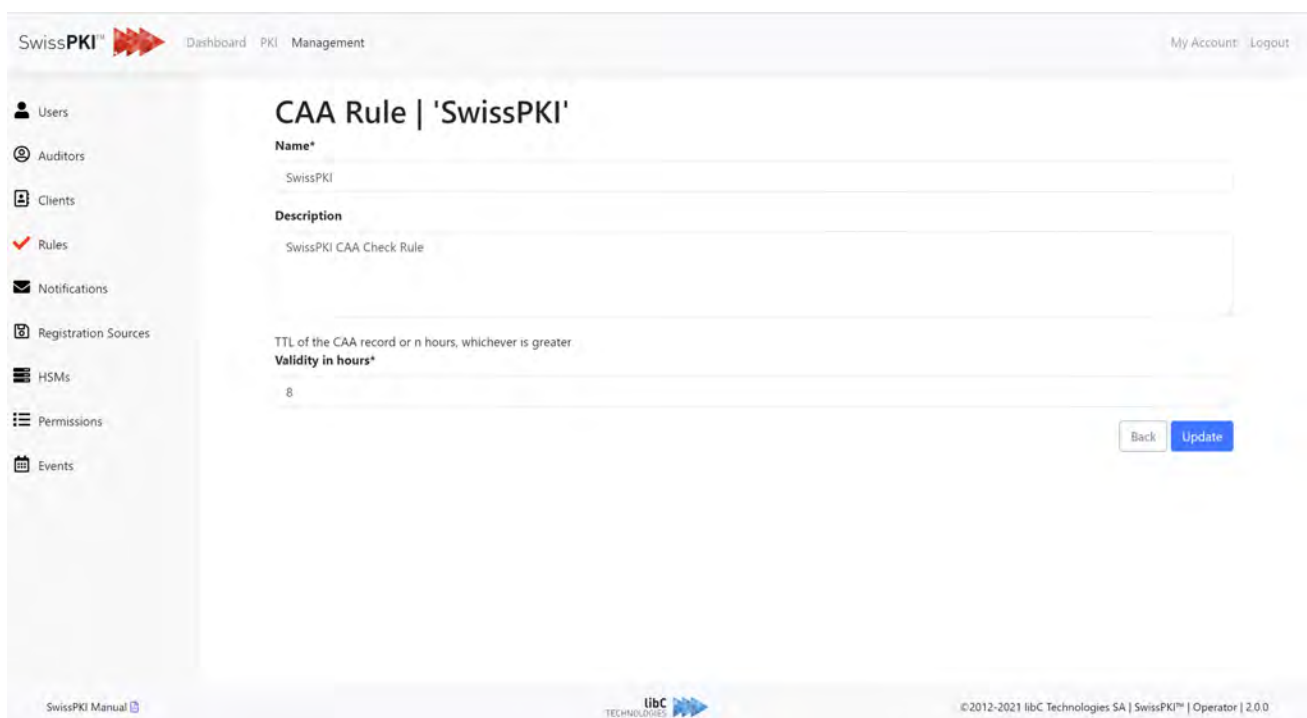
Below the table, it says "Showing 1 to 1 of 1 entries". There are "Previous" and "Next" navigation buttons. The footer of the interface includes "SwissPKI Manual", the libC Technologies logo, and copyright information: "©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".




### 12.2.4.4.1 Create CAA

Creating a new CAA rule is done by clicking on the button located in the rule's navigation. After clicking it, you are redirected to a form where you need to provide the following information:

Fields	Description
<b>Name</b>	The CAA rule's name
<b>Description</b>	The CAA rule's description
<b>Validity in hours</b>	The CAA rule's TTL validity in hours



SwissPKI  Dashboard PKI Management My Account Logout



**CAA Rule | 'SwissPKI'**

**Name\***  
SwissPKI


**Description**  
SwissPKI CAA Check Rule

TTL of the CAA record or n hours, whichever is greater  
**Validity in hours\***  
8

[Back](#) [Update](#)

SwissPKI Manual  libC TECHNOLOGIES  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

## Define the CAA Domain and notification recipient



The screenshot shows the 'SwissPKI' management interface. The left sidebar contains navigation options: Users, Auditors, Clients, Rules (selected), Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area is titled 'CAA Rule | 'SwissPKI'' and includes a search bar and a table with the following data:

Created	Modified	CAA Domain	Actions
24.11.2021	24.11.2021	swisspki.com	[Add] [Edit] [Delete]

Below the table, it indicates 'Showing 1 to 1 of 1 entries'. At the bottom right, there are 'Previous' and 'Next' buttons, and a 'Back' button. The footer contains 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

For each CAA Domain, you can optionally define IODEFs to notify mailboxes about rejected certificate issuance when CAA lookups fail



The screenshot shows the 'SwissPKI' management interface for IODEF configuration. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Domain 'swisspki.com' IODEFs for CAA Rule 'SwissPKI'' and includes a search bar and a table with the following data:

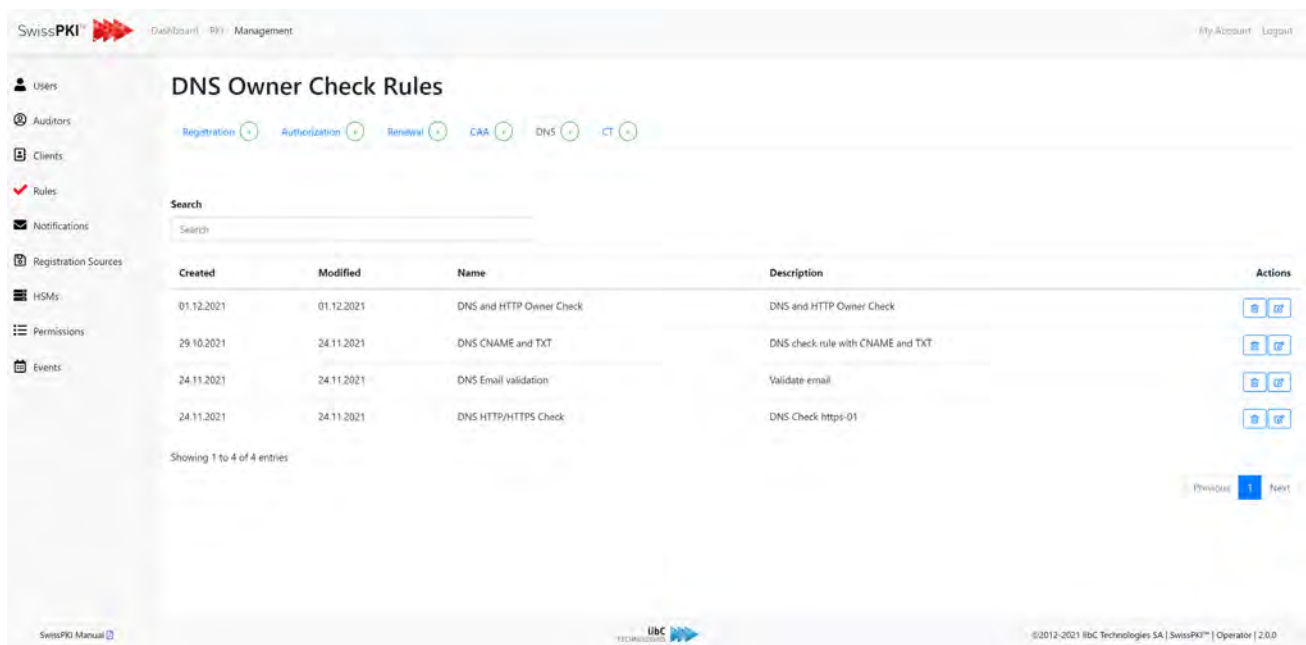
Created	Modified	IODEF Mail To	Actions
24.11.2021	24.11.2021	infra@swisspki.com	[Add] [Edit]
24.11.2021	24.11.2021	support@swisspki.com	[Add] [Edit]

Below the table, it indicates 'Showing 1 to 2 of 2 entries'. At the bottom right, there are 'Previous' and 'Next' buttons, and a 'Back' button. The footer contains 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.2.4.5 DNS Owner Check Rules

DNS Owner Check Rules are used to validate DNS entries based on *8.2.2 Challenge Tokens*.

DNS Owner Check Rules are linked to a Policy Template (see *12.3.2 Certificate Policy Templates*).



The screenshot shows the 'DNS Owner Check Rules' management page in the SwissPKI interface. The page includes a navigation menu on the left, a breadcrumb trail 'Dashboard > PKI > Management', and a search bar. Below the search bar is a table listing four DNS Owner Check Rules. Each rule entry includes its creation and modification dates, name, description, and a set of action buttons (edit, delete, etc.).

Created	Modified	Name	Description	Actions
01.12.2021	01.12.2021	DNS and HTTP Owner Check	DNS and HTTP Owner Check	[Edit] [Delete]
29.10.2021	24.11.2021	DNS CNAME and TXT	DNS check rule with CNAME and TXT	[Edit] [Delete]
24.11.2021	24.11.2021	DNS Email validation	Validate email	[Edit] [Delete]
24.11.2021	24.11.2021	DNS HTTP/HTTPS Check	DNS Check http:01	[Edit] [Delete]

Showing 1 to 4 of 4 entries

Navigation: Previous 1 Next

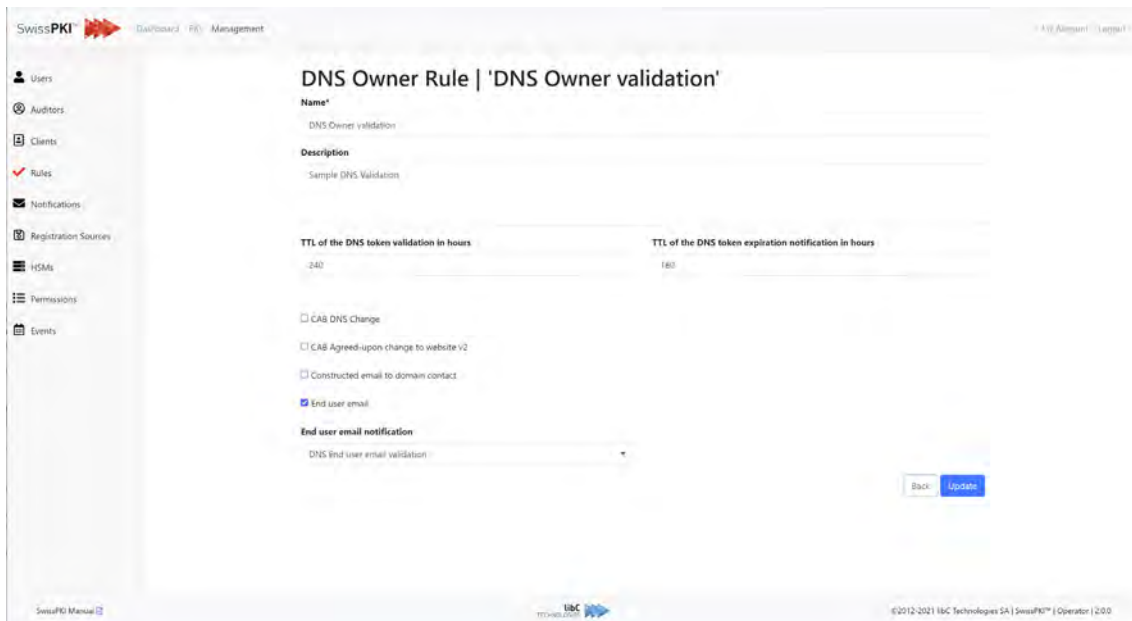
### 12.2.4.5.1 Create DNS Rule

Creating a new DNS rule is done by clicking on the add button located in the rule's navigation. After clicking it, you will be redirected to a form where you need to provide the following information:

Fields	Description
<b>Name</b>	The DNS rule's name
<b>Description</b>	The DNS rule's description
<b>Reuse previous domain validations</b>	<p>If enabled</p> <p>If a domain validation for a specific domain was performed in the past no longer than the domain revalidation interval ago, then no new domain validation must be performed when issuing another certificate for the same domain ( by the same client )</p> <p>If disabled</p> <p>The domain validation must be performed again for every certificate order.</p> <p>Note: This option considers validations performed by previous orders. If the client has pre-validated client domains, he does not have to revalidate the domain even if this option is disabled.</p>
<b>Domain revalidation interval ( in days )</b>	A domain validation can be used the specified number of days before it needs to be revalidated
<b>Send a notification x days before expiration</b>	For unsuccessful validations, send a notification x days before the validation expires. Usually, validation token expires after 30 days
<b>CAB DNS Change</b>	<p>Enable or not the DNS Record check as per CAB specifications.</p> <p>When enabled, let us you select a notification template</p>
<b>CAB Agreed-upon change to website v2</b>	<p>Enable or not the HTTP Record check as per CAB specifications</p> <p>When enabled, let us you select a notification template</p>
<b>HTTP well known path</b>	<p>Only appears when "HTTP check" is selected. Let us you override the well-known path.</p> <p>When enabled, let us you select a notification template</p>
<b>Constructed email to domain contact</b>	<p>Send email notification to constructed postmaster as per CAB specifications.</p> <p>When enabled, let us you select a notification template</p>
<b>End user email</b>	Enable or not the email validation

When enabled, let us you select a notification template

**Note:** Send an email with link to validate email address(es). The certificate issuance is put on hold. The email link is valid 30 days.



#### 12.2.4.6 CT Rules

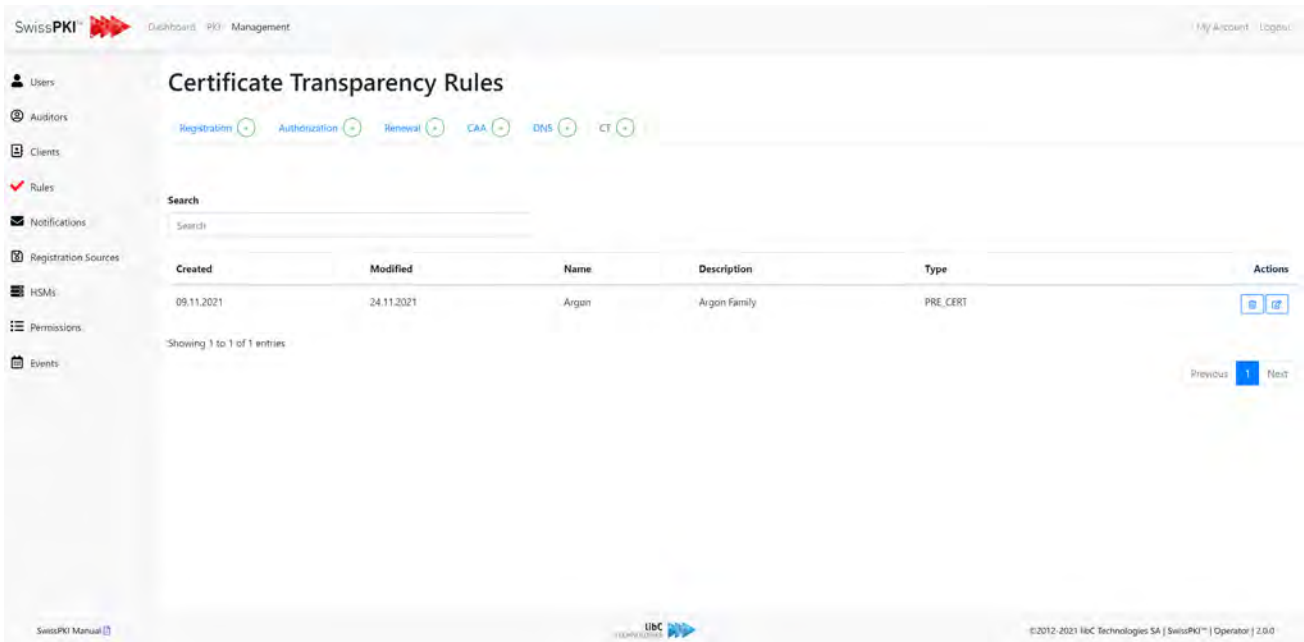
Certificate Transparency is an open framework for monitoring SSL Certificates. Domain owners may find it useful to monitor certificate issuance for their domain and use that to detect mis-issued certificates

With CT, all certificates are publicly disclosed, providing greater insight and transparency into the Web PKI ecosystem. The CT aims to achieve three goals:

1. To make it impossible (or at least difficult) for a CA to issue an SSL Certificate for a domain without the certificate being visible to the owner of that domain.
2. To provide an open auditing and monitoring system that lets any domain owner or CA determine whether certificates have been mistakenly or maliciously issued.
3. To protect users from being duped by certificates that were mistakenly or maliciously issued.

You create CT Rules based on the CT Log Families defined in Realm (please refer to *11.5.9 CT Log Families*)


CT Rules is linked to a Policy Template (see *12.3.2 Certificate Policy Templates*).



#### 12.2.4.6.1 Create CT Rule

Creating a CT rule is done by clicking on the add button located in the rule’s navigation. After clicking the button, you are redirected to a form where you will need to provide the following information:


Fields	Description
<b>Name</b>	The CT rule name
<b>Description</b>	The CT rule description
<b>Type</b>	The CT rule type (PRE Certificate or OCSP Stapling)
<b>CT Log Families</b>	Selection of log families defined at the Realm level

SwissPKI™  Dashboard PKI Management My Account Logout

**Certificate Transparency Rule | 'Argon'**

<b>Name*</b>	Argon	<b>Type</b>	Pre Certificate
<b>Description</b>	Argon Family		
<b>CT Log Families*</b>	<input type="text" value="Argon"/>		

**Argon** [Back](#) [Update](#)

[SwissPKI Manual](#)  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

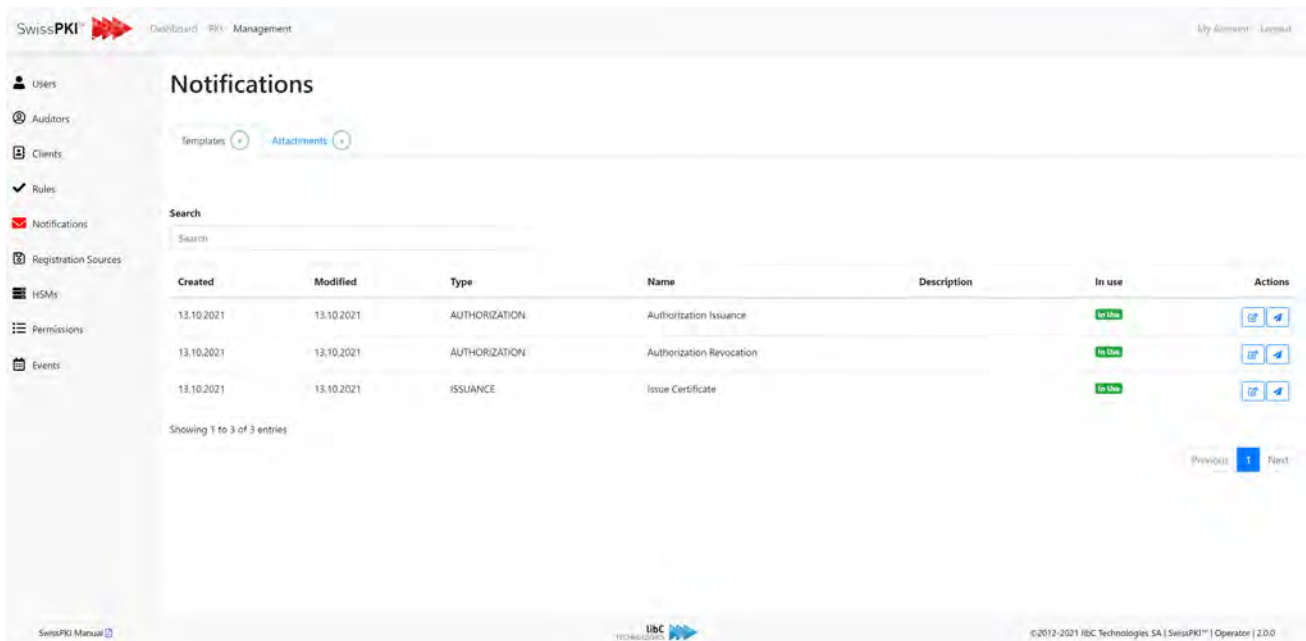
## 12.2.5 Notifications Templates

Notifications Templates are message templates with predefined content which are associated to Client Policy Mappings (see 12.3.1.1.1.2.3 Attachments *Policy instance mappings*) or workflow Rules (see 12.2.4 Rules)

Notification Templates are divided in two sections:

1. Templates  
Effective notification

Optional attachment linked to a notification template



SWISSPKI Dashboard PKI Management My Account Logout

**Notifications**

Templates + Attachments -


**Search**

Search

Created	Modified	Type	Name	Description	In use	Actions
13.10.2021	13.10.2021	AUTHORIZATION	Authorization Issuance		In Use	Off On
13.10.2021	13.10.2021	AUTHORIZATION	Authorization Revocation		In Use	Off On
13.10.2021	13.10.2021	ISSUANCE	Issue Certificate		In Use	Off On

Showing 1 to 3 of 3 entries

Previous 1 Next

SwissPKI Manual  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0



### 12.2.5.1 Notifications and Recipients

SwissPKI distinguished between different type of notifications:

- Templated notification where you can define your own multilingual message content and attribute value place holder
- User notifications which are non-editable
- Self-service ticket notifications

#### 12.2.5.1.1 Templated notifications

Templated notifications sent to selected (enabled/disabled) roles:

Notification type	Description	Recipient(s)
<b>Certificate issuance</b>	Templated notification upon certificate issuance events	Certificate owner (if present) RAO(s) CAO(s) Client Custom recipient(s) Additional recipient(s)
<b>Certificate renewal</b>	Templated notification upon certificate renewal events	Certificate owner (if present) RAO(s) CAO(s) Client Custom recipient(s) Additional recipient(s)
<b>Authorizations</b>	Templated notification upon authorization events (issuance, renewal, revocation, recovery, accepted and rejected authorization requests)	RAO(s) Authorizer(s) Custom recipient(s) Additional authorizers(s)
<b>Recovery</b>	Templated notification upon key recovery events (Microsoft CES/CEP)	Certificate owner (if present) RAO(s) CAO(s) Client

		Custom recipient Additional recipient(s)
<b>Revocation</b>	Templated notification upon certificate revocation events	Certificate owner (if present) RAO(s) CAO(s) Client Custom recipient Additional recipient(s)
<b>End user email validation</b>	Templated notification upon end user email validation events (email confirmation link)	Certificate owner (if present) RAO(s) CAO(s) Client Custom recipient(s)
<b>Constructed email to domain owner</b>	Templated notification upon DNS challenge registration events	Constructed email to domain owner (automatically sends to the five allowed domain owners) RAO(s) CAO(s) Client Custom recipient
<b>CAB DNS change/Agreed-upon change to website v2</b>	Templated notification upon DNS challenge registration events	RAO(s) CAO(s) Client Custom recipient Additional recipient(s)
<b>CAB DNS email link to domain contact</b>	Templated notification upon DNS challenge registration events	Selected email to domain owner Additional recipient(s)

**Note:** Users have the possibility, when assigned an RAO role, to mute notifications generated by other RA Operators associated with the same Client. Refer to section *12.2.5.2 Notifications*

### 12.2.5.1.2 User notifications

User and PKI Realm notifications:

Notification type	Description	Recipient(s)
<b>CMP Renewal</b>	Manual or automatic CMP certificate renewal triggered	CAO(s)
<b>DSS Renewal</b>	Manual or automatic DSS certificate renewal triggered	CAO(s)
<b>OCSP Renewal</b>	Manual or automatic OCSP certificate renewal triggered	CAO(s)
<b>TSA Renewal</b>	Manual or automatic TSA certificate renewal triggered	CAO(s)
<b>Login PIN confirmation</b>	Link to password reset page upon password change request	SwissPKI user when username/password with TOTP is enabled
<b>Login PIN changed</b>	Information that user login password was changed	SwissPKI user when username/password with TOTP is enabled
<b>Login PIN change request</b>	Information about password reset event	SwissPKI user when username/password with TOTP is enabled
<b>Login TOTP change</b>	Updated TOTP token initiated by a CA Operator or user account	SwissPKI user when username/password with TOTP is enabled
<b>Updated user role</b>	Information about role update (add/remove)	SwissPKI user when new role is associated to the user
<b>Expiration of DNS challenge</b>	DNS challenge for domain is expired	Client Contact Info (technical contact)
<b>Successful validation of DNS challenge</b>	DNS challenge for domain did get successfully validated	Client Contact Info (technical contact)

<b>Renewal of DNS challenge</b>	DNS challenge for domain should be renewed within 'd' days	Client Contact Info (technical contact)
---------------------------------	--	---

### 12.2.5.1.3 Self-service notifications

Self-service notifications<sup>16</sup> are sent to certificate recipients and certificate key reminder recipients when certificate renewal rules are enabled for certificate managed by RA Operators.

Notification type	Description
<b>Automatic certificate renewal</b>	<p>Certificate owners receive an automated notification (self-service ticket) when:</p> <ul style="list-style-type: none"> <li>The certificate policy template key generation is of type PKCS#10 and the certificate renewal rules requires 're-keying', then the recipient(s) are emailed a link for copy/pasting a new CSR for the automatic renewal request. If no end user recipient is available in the certificate, then the RAO, CAO, Custom recipient, Additional recipients are notified if enabled on the notification template.</li> <li>The certificate policy template key generation is of type PKCS#12 or HSM and the certificate renewal rules requires 're-keying', then the recipient(s) are emailed a link for to download the new PKCS#12 file. In case of HSM key generation, a notification is sent to the recipient(s) including the new certificate and information relative to the HSM partition. The HSM partition PIN must be obtained out of band by the recipient(s). If no end user recipient is available in the certificate, then the RAO, CAO, Custom recipient, Additional recipients are notified if enabled on the notification template.</li> </ul>
<b>PKCS#12 key generation with User PIN</b>	Issuing certificates with PKCS12 with User provided PIN key generation type, sends a self-service notification ticket to the certificate's recipient for setting the PKCS#12 PIN. If no end user recipient is available in the certificate, then the

<sup>16</sup> Templated notifications packaged with the SwissPKI deployment. Those notifications are not editable through the OperatorUI but can be white labelled (see support FAQ white labelling)

	<p>RAO, CAO, Custom recipient, Additional recipients are notified if enabled on the notification template.</p> <p>The end user provided PKCS#12 PIN cannot be recovered from the PKI. If the PIN is lost, the end user must request a new certificate.</p>
<p><b>PKCS#12 key generation with CA PIN</b></p>	<p>Issuing certificates with PKCS12 with CA generated PIN key generation type, sends a self-service notification ticket to the certificate's recipient for recovering the CA generated PKCS#12 PIN.</p> <p>This page is accessible only once.</p> <p>If no end user recipient is available in the certificate, then the RAO, CAO, Custom recipient, Additional recipients are notified if enabled on the notification template.</p> <p>The end user provided PKCS#12 PIN cannot be recovered from the PKI. If the PIN is lost, the end user must request a new certificate.</p>
<p><b>PKCS#12 key generation</b></p>	<p>Issuing certificates with PKCS12 key generation type, sends a self-service notification ticket to the certificate's recipient for setting the CA generated PKCS#12 PIN.</p> <p>Additional key recovery recipients can be added when requesting the certificate or at a later time added to the certificate key reminder recipient list.</p> <p>The end user provided PKCS#12 PIN can be recovered from the PKI. If the PIN is lost, the RA Officer can resend the PIN information to the recipients.</p>

#### 12.2.5.1.4 Additional Recipients

Additional recipients are optionally added per certificate order request and editable by RA Officers<sup>17</sup> in the certificate detail view once the certificate is issued.

Recipient type	Description
<b>DNS_OWNER_CHECK_EMAIL_LINK</b>	When a DNS CAB email link domain owner check is enabled on the certificate policy template, an additional recipient for the delivery of the validation link must be added to the request. The recipient email must match the domain owner.  <b>Example:</b> requesting a certificate for domain <code>www.swisspki.com</code> requires a domain owner check email of type <code>admin</code>   <code>administrator</code>   <code>webmaster</code>   <code>hostmaster</code>   <code>postmaster @swisspki.com</code>
<b>DNS_CAB</b>	When a DNS CAB (dns-01 and/or http-01) check is enabled on the certificate policy template, an additional recipient is notified with the DNS token
<b>ISSUANCE</b>	Notifies certificate issuance to the additional recipient
<b>REVOCACTION</b>	Notifies certificate revocation to the additional recipient
<b>RENEWAL</b>	Notifies certificate renewal to the additional recipient
<b>RECOVERY</b>	Notifies key recovery to the additional recipient
<b>AUTHORIZATION</b>	Notifies authorization request to the additional recipient
<b>AUTHORIZATION_ACCEPTED</b>	Notifies accepted authorization to the additional recipient
<b>AUTHORIZATION_REJECTED</b>	Notifies rejected authorization to the additional recipient

<sup>17</sup> Additional recipient management is also available in the Registration API

### 12.2.5.2 Notifications

Notification templates are created for:

1. Certificate renewal
2. Certificate issuance
3. Certificate revocation
4. Authorizations for certificate issuance, revocation, renewal, and recovery
5. Key recovery
6. End user Email validation
7. Constructed Email to domain contact (CAB DNS)
8. CAB DNS change/CAB Agreed-upon change to website v2
9. CAB DNS email link to domain contact

Creating a new notification template is done by clicking on the add button located inside of the Templates tab

Additionally, the following actions are accessible through the table's action tab:

- Deleting a notification template
- Editing a notification template
- Creating notifications for a template

A template can refer to notifications in multiple languages. When the recipient's language is known such as CAO or RAO the existing notification in the user's language is selected. If no notification in the user's language is not found, then the neutral notification is used.

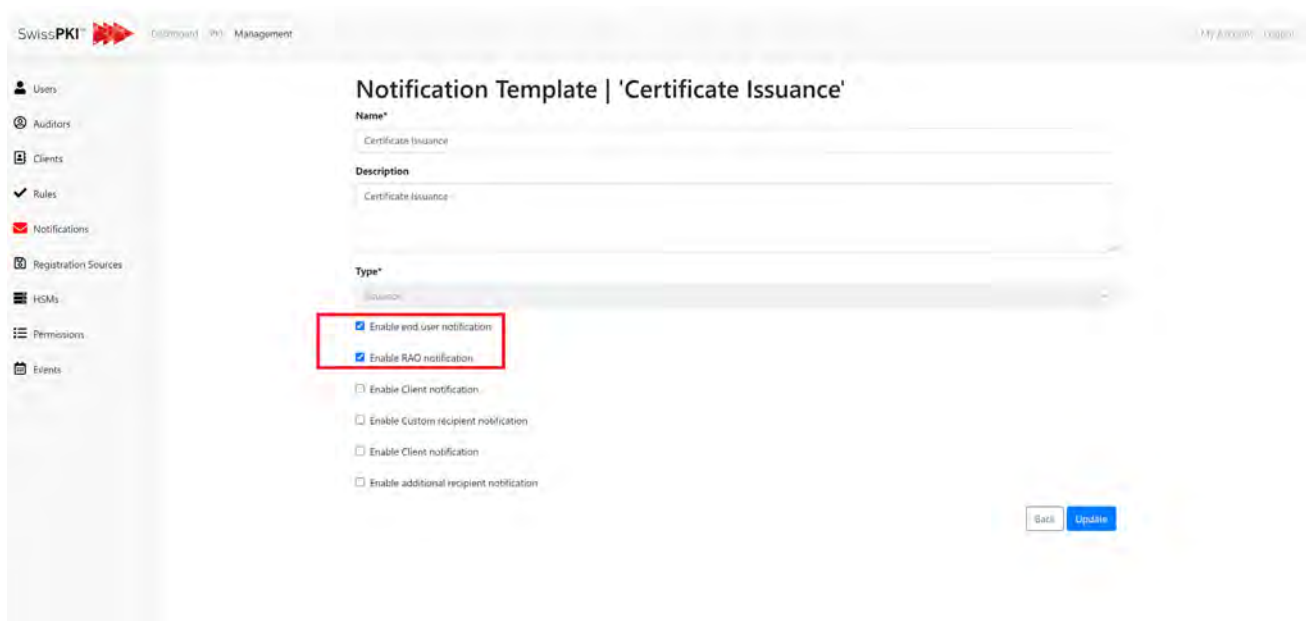
Supported languages are: Neutral, English, French and German

**Note:** a neutral language corresponds to an email notification written in multiple languages.

### 12.2.5.2.1 Create Notification Template

To create a new notification template, you will have to provide the following information:

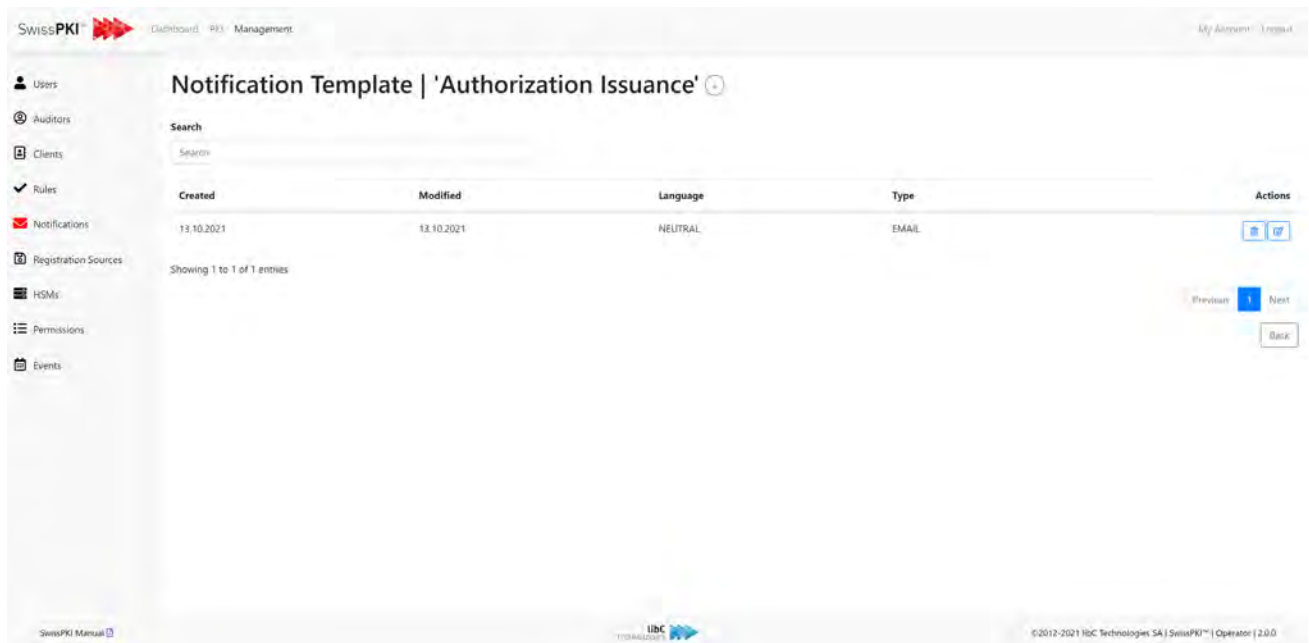
Fields	Description
<b>Name</b>	The notification template's name
<b>Description</b>	The notification template's description
<b>Type</b>	The notification template's type. Five types are available: <ul style="list-style-type: none"> <li>• Renewal</li> <li>• Recovery</li> <li>• Issuance</li> <li>• Revocation</li> <li>• Authorization</li> <li>• End user email validation</li> <li>• Constructed email to domain contact</li> <li>• CAB DNS change/CAB Agreed-upon change to website v2</li> <li>• CAB DNS email link to domain contact</li> </ul>
<b>Enable Notification</b>	By ticking this box, you will activate the notification for the selected roles.







### 12.2.5.2 Notification

Notifications are created for a template. The list of notifications is accessed by clicking on the template's notifications button.



The screenshot shows the SWISSPKI Management interface. The main content area is titled "Notification Template | 'Authorization Issuance'". Below the title is a search bar. A table lists the notification details:

Created	Modified	Language	Type	Actions
13.10.2021	13.10.2021	NEUTRAL	EMAIL	 

Below the table, it says "Showing 1 to 1 of 1 entries". Navigation buttons for "Previous", "Next", and "Back" are visible. The footer includes "libC Technologies" and "©2012-2021 libC Technologies SA | SWISSPKI™ | Operator | 2.0.0".

### 12.2.5.2.2.1 Create Notification

To create a new notification, you need to provide the following information:

Fields	Description
<b>Language</b>	The notification's language. <ul style="list-style-type: none"> <li>• Neutral</li> <li>• English</li> <li>• French</li> <li>• German</li> </ul>
<b>Notification Type</b>	The notification's type: <ul style="list-style-type: none"> <li>• EMAIL</li> </ul>
<b>The Notification Recipient</b>	The available recipient differs depending on the notification type. A list of all recipients is available in the notification chapter.

## Notification Message | 'CAB DNS Additional recipient'

**Language** Neutral **Notification type** EMAIL

Additional Recipient

**CC**

CC








**BCC**

BCC

**Subject**

You certificate request with Id \$(CertificateOrderData.UUID) in status \$(CertificateOrderData.certificateOrderStatus) requires DNS validation.

**Message**

Tag       **A** **B** *I* U     Segoe UI 14    **T**  

Dear customer,

You certificate request with Id \$(CertificateOrderData.UUID) in status \$(CertificateOrderData.certificateOrderStatus) requires DNS validation.

In order to validate your domain \$(DNSValidationTokenData.domain) to meet the requirements for public certificates of the CA/Browser Forum, please create a DNS TXT record for \$(DNSValidationTokenData.domain) with value \$(DNSValidationTokenData.token).

We will issue your certificate once the DNS TXT record is created or you can manually force validation of your domain via the Web RA UI (add URL here).

Please note that your Random Value token is valid until \$(DNSValidationTokenData.validUntil).

This is an automatically generated email.

Best regards,  
Your SwissPKI team

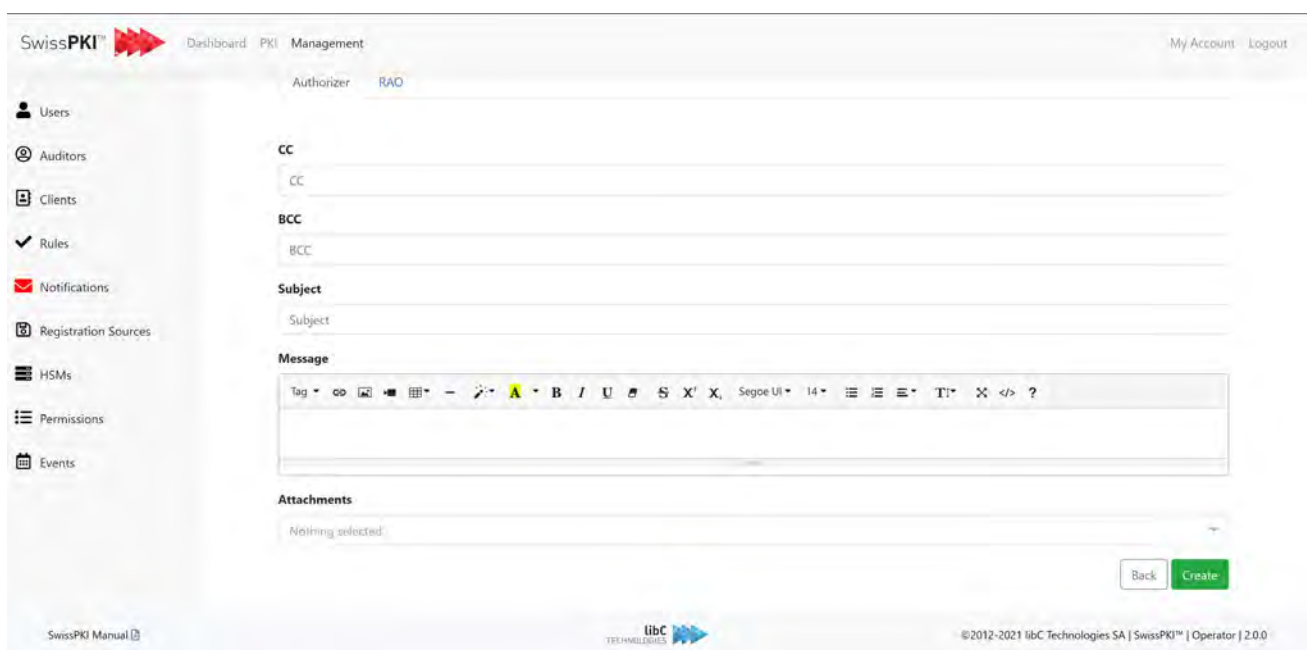
**Attachments**

Nothing selected

[Back](#) [Update](#)

Once the notification is enabled for a recipient, you can configure the following fields:

Fields	Description
<b>CC</b>	The people in copy of the notification
<b>BCC</b>	The people in hidden copy of the notification
<b>Subject</b>	The notification's subject
<b>Message</b>	The notification message content. To help you include useful information about the certificate, tags are available.
<b>Attachments</b>	The files attached to this notification



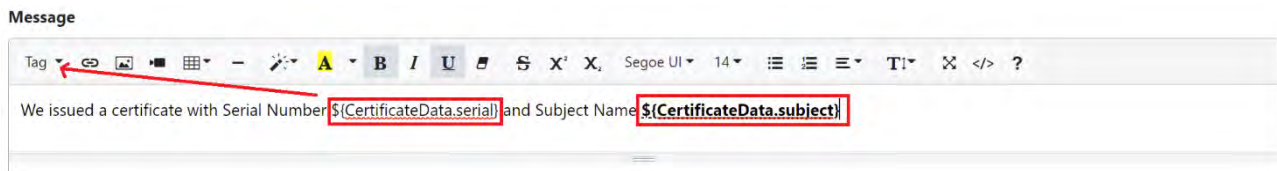
The screenshot shows the SwissPKI Management interface. The breadcrumb trail is Dashboard > PKI > Management. The current page is 'Authorizer' for 'RAO'. The left sidebar contains navigation items: Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area has the following fields:

- CC:** A text input field.
- BCC:** A text input field.
- Subject:** A text input field.
- Message:** A rich text editor with a toolbar (bold, italic, underline, link, unlink, list, list, link, unlink, code, help) and a text area.
- Attachments:** A file selection area showing 'Nothing selected'.

At the bottom right, there are 'Back' and 'Create' buttons. The footer contains 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '© 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.2.5.2.3 Notification attributes

Based on the notification type, you can compose messages including runtime values by inserting predefined tags. From the Message textbox -> Tags menu, you select runtime variables which will be replaced by the content value when processing the notification.



#### Note:

- Unknown parameters will cause the template processing engine to fail generating the notification. No notification is sent.
- Undefined or NULL parameter values will not be processed by the template engine. A notification with a missing placeholder value is sent.
- Optionally, you can test for runtime parameter values and print a message.

If `CertificateOrderData.revocationCodeLink` is present, then process the enclosed block of text

```
[if CertificateOrderData.revocationCodeLink?? if]
```

Certificate Revocation

Use this link whenever you need to revoke your certificate.  
Revocation link: `${CertificateOrderData.revocationCodeLink}`

```
[fi]
```

### 12.2.5.2.4 Custom recipients

Custom recipients email addresses support semi-colon separated email addresses.

## Notification Message | 'Issuance sample'

Language: Neutral | Notification type: EMAIL

End User | CAO | Client | Custom Recipient



**To**  
custom.one@swisspki.ch;custom.two@swisspki.com;custom.three@swisspki.com

**CC**  
CC

**BCC**  
BCC

**Subject**  
Subject

**Message**

Tag |  Segoe UI | 14 | 

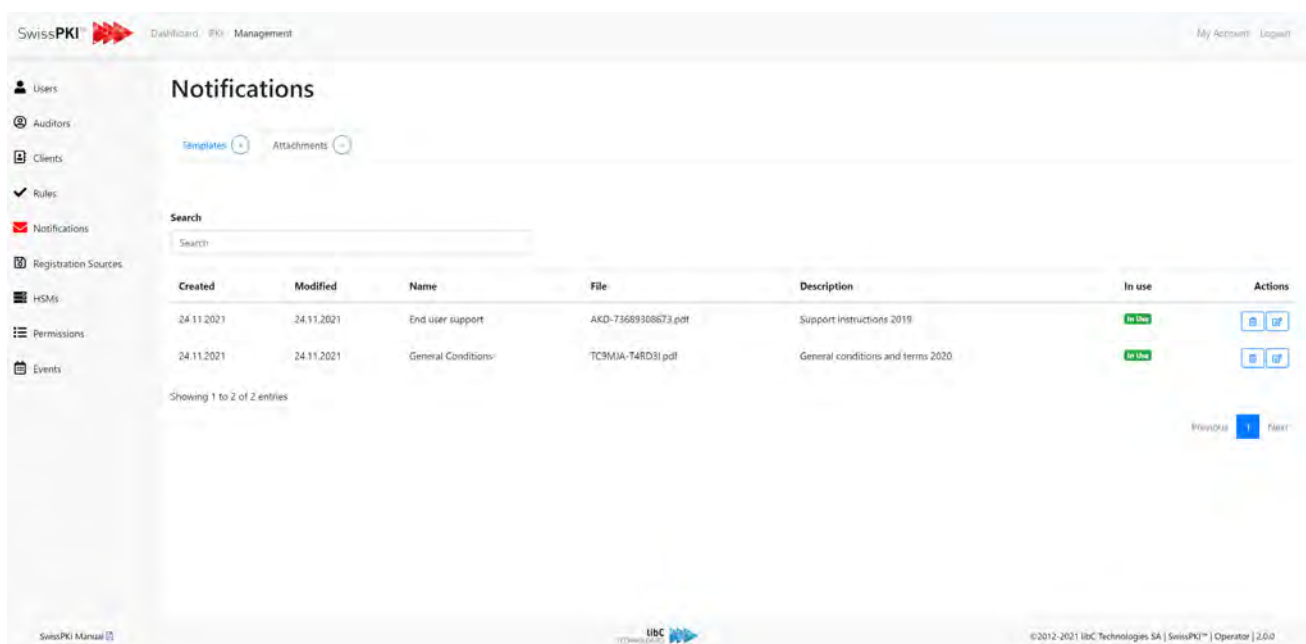
**Attachments**  
Nothing selected

Back Update





### 12.2.5.3 Notification Attachments

Attachments are PDF files that are unloadable in the application and used for notifications. The PDF files are processed in a transparent manner: the documents are sent as attachments without modifications.



A list of all available attachments is found on the attachments tab. You can add new one by clicking on the add button in the tab. Additionally, you can delete or edit existing attachments by using the buttons in the action's column of the table.



The screenshot shows the 'Notifications' management interface in SwissPKI. The 'Attachments' tab is active, displaying a table of notification attachments. The table has the following data:

Created	Modified	Name	File	Description	In use	Actions
24.11.2021	24.11.2021	End user support	AKD-T3689308673.pdf	Support instructions 2019	In Use	 
24.11.2021	24.11.2021	General Conditions	TC9MJA-T4RD31.pdf	General conditions and terms 2020	In Use	 

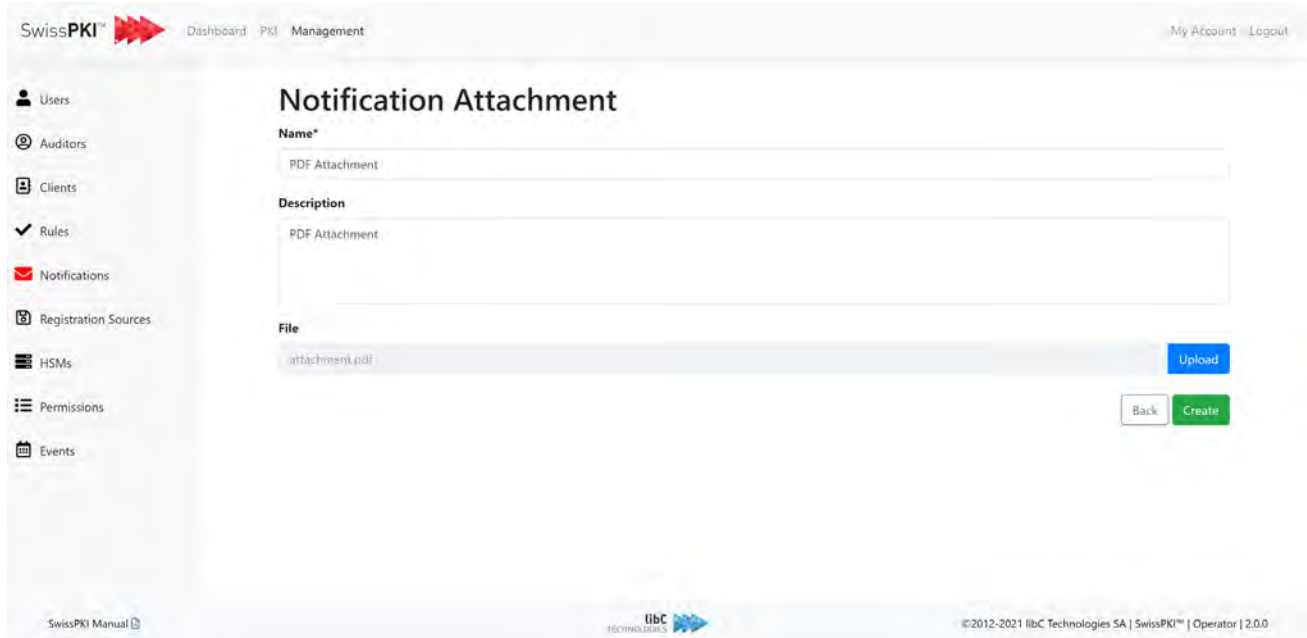
Showing 1 to 2 of 2 entries

Navigation:  **1** 

### 12.2.5.3.1 Create Notification Attachment

To add a file as a notification attachment, you need to provide the following information:

Fields	Description
<b>Name</b>	The notification attachment's name
<b>Description</b>	The notification attachment's description
<b>File</b>	The file you wish to add as an attachment



The screenshot shows the 'Notification Attachment' form in the SwissPKI web interface. The form has three main sections: 'Name\*', 'Description', and 'File'. The 'Name\*' field contains the text 'PDF Attachment'. The 'Description' field also contains 'PDF Attachment'. The 'File' field shows a file named 'attachment.pdf' with an 'Upload' button next to it. At the bottom right of the form, there are 'Back' and 'Create' buttons. The interface includes a sidebar with navigation options like Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMS, Permissions, and Events. The footer contains the libC Technologies logo and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.



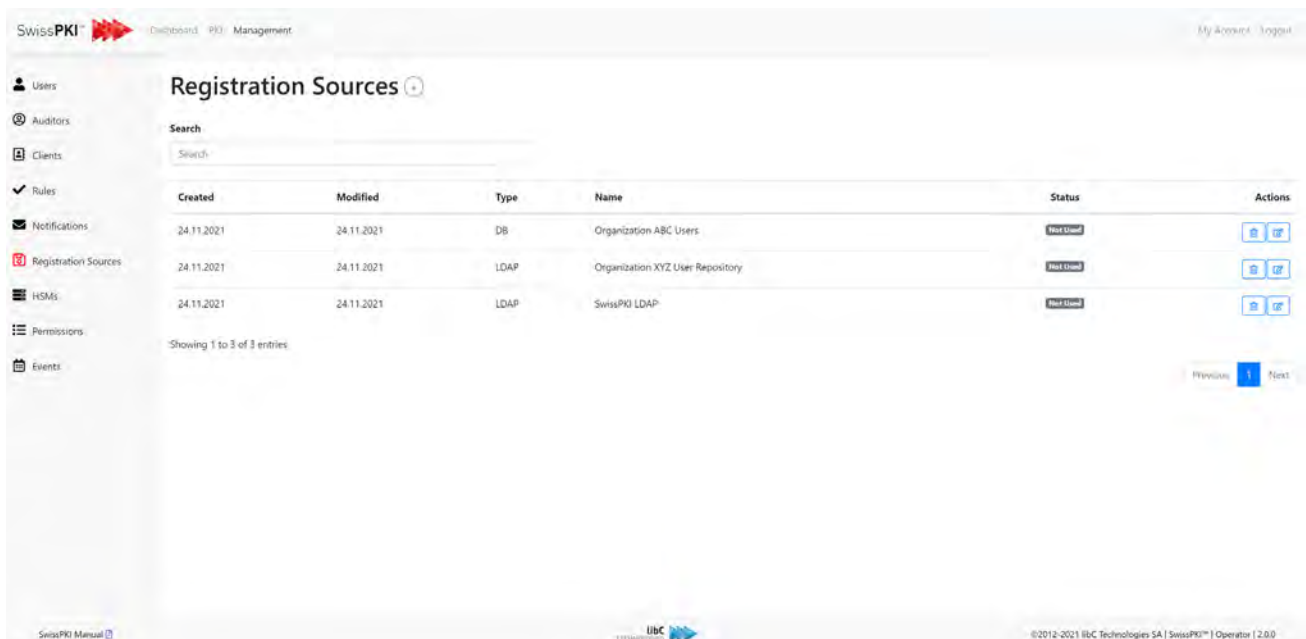
## 12.2.6 Registration Sources

Registration sources are identified data sources which enable you to restrict issuance of certificates limited to the records defined within the boundary of those sources. When associated to a Policy Mapping (see 12.3.1.1.2.3 Policy instance mappings), RA Operators can only issue certificates with the data available from the selected registration sources.

There are two distinct types of data sources:

1. LDAP Data Sources
2. DB Data Sources

Registration sources are limited to information identifying persons (*inetOrgPerson* for LDAP and *t\_registered\_users* for DB)



SwissPKI Management Dashboard

### Registration Sources

Search

Created	Modified	Type	Name	Status	Actions
24.11.2021	24.11.2021	DB	Organization ABC Users	Not Configured	<a href="#">Edit</a> <a href="#">Delete</a>
24.11.2021	24.11.2021	LDAP	Organization XYZ User Repository	Not Configured	<a href="#">Edit</a> <a href="#">Delete</a>
24.11.2021	24.11.2021	LDAP	SwissPKI LDAP	Not Configured	<a href="#">Edit</a> <a href="#">Delete</a>

Showing 1 to 3 of 3 entries

SwissPKI Manual


libC TECHNOLOGIES










©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.6.1 LDAP Data Source

A LDAP Data Source identifies *inetOrgPerson* object classes as registration candidates for certificate issuance. You configure a LDAP data source providing the following information

Attribute	Description
<b>Name</b>	Logical name of the LDAP Data Source
<b>Description</b>	Description of the LDAP Data Source
<b>Comment</b>	Comment of the LDAP Data Source
<b>Host</b>	LDAP host
<b>Port</b>	LDAP port, usually 636 or 389
<b>Base DN</b>	The LDAP DIT base DN where to search for <i>inetOrgPerson</i> object classes
<b>Bind DN</b>	LDAP user name
<b>Bind Password</b>	LDAP user password
<b>Search Filter</b>	Optional LDAP search filter

SwissPKI  Dashboard PKI Management My Account Logout

-  Users
-  Auditors
-  Clients
-  Rules
-  Notifications
-  Registration Sources
-  HSMs
-  Permissions
-  Events

## LDAP Registration source | 'Organization XYZ User Repository'

**Name\***  
Organization XYZ User Repository

**Description**  
Organization XYZ User Repository


**Comment**

**Host\*** ldap.swisspki.com **Port\*** 636

**Base DN\*** ou=USers,dc=swisspki,dc=com **Bind DN\*** cn=admin,dc=swisspki,dc=com

**Bind Password\*** ..... **Search filter** (&(objectclass=\*)(ou=\*))

[Back](#) [Update](#)

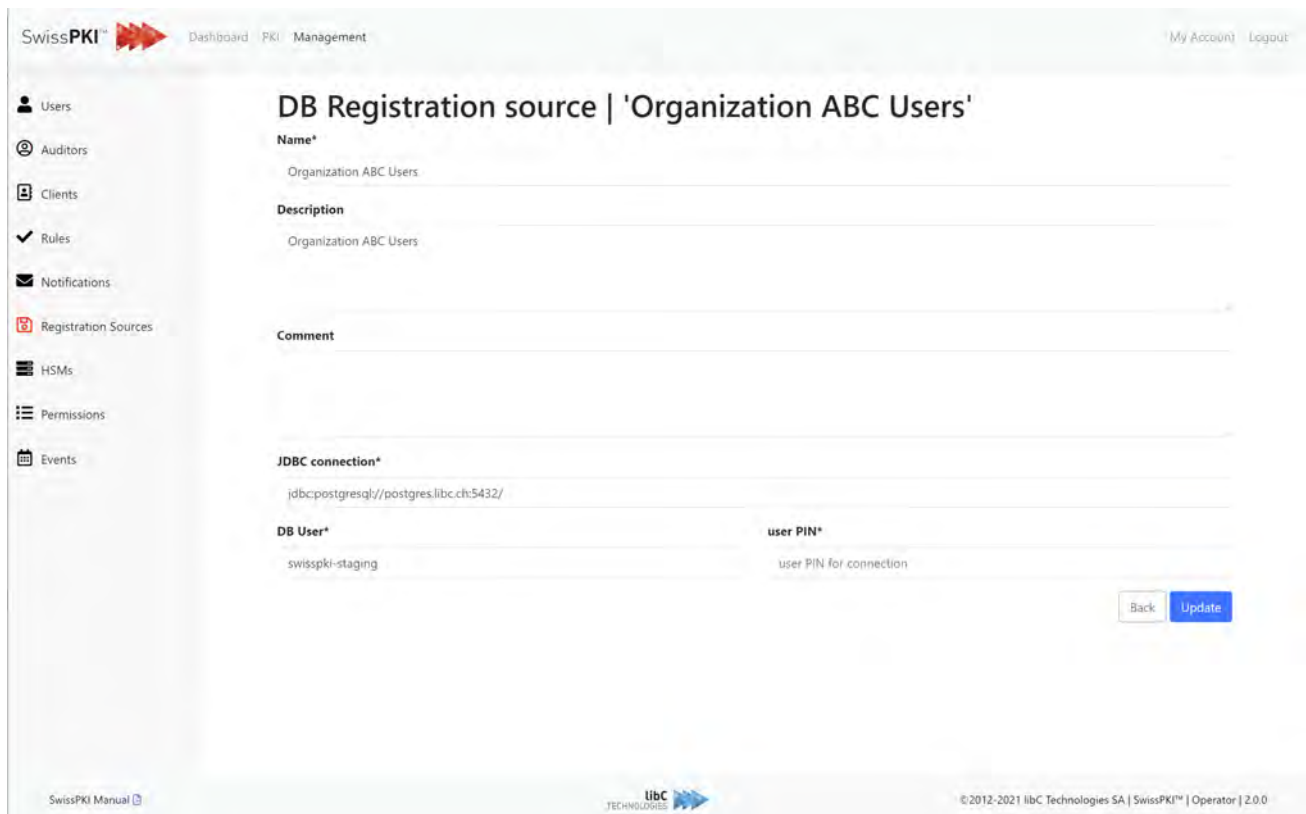
SwissPKI Manual [↗](#)  ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

There is no limitation to the number of LDAP data sources and number of LDAP data sources associated to a specific Client Policy Mapping. When mapping multiple LDAP (or DB) data sources to a Client Policy Mapping, searches occur across all LDAP sources.

### 12.2.6.2 DB Data Source

A DB Data Source identifies *t\_registered\_users* as registration candidates for certificate issuance. You configure a DB data source providing the following information

Attribute	Description
<b>Name</b>	Logical name of the DB Data Source
<b>Description</b>	Description of the DB Data Source
<b>Comment</b>	Comment of the DB Data Source
<b>JDBC Connection</b>	JDBC connection string in the form of <i>jdbc:server://host:port/... parameters ...</i>
<b>DB User</b>	DB user to use for the connection
<b>User PIN</b>	DB user PIN



The screenshot shows the 'SwissPKI' management interface. The main content area is titled 'DB Registration source | 'Organization ABC Users''. The configuration fields are as follows:

- Name\***: Organization ABC Users
- Description**: Organization ABC Users
- Comment**: (empty)
- JDBC connection\***: jdbc:postgresql://postgres.libc.ch:5432/
- DB User\***: swisspki-staging
- user PIN\***: user PIN for connection

At the bottom right of the form, there are 'Back' and 'Update' buttons. The footer of the interface includes 'SwissPKI Manual', the libC TECHNOLOGIES logo, and the copyright notice '© 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

There is no limitation to the number of LDAP data sources and number of LDAP data sources associated to a specific Client Policy Mapping. When mapping multiple LDAP (or DB) data sources to a Client Policy Mapping, searches occur across all LDAP sources.

#### 12.2.6.2.1 DB table requirement

The DB schema used for the registered users must contain the following table definition:

```
create table t_registered_users(  
    ID int default 0 not null primary key,  
    CN varchar(255) not null,  
    FIRST_NAME varchar(255) null,  
    LAST_NAME varchar(255) null,  
    ORG_UNIT varchar(255) null,  
    ORGANIZATION varchar(255) null,  
    MAIL varchar(255) not null,  
    constraint t_registered_users_mail_uindex unique (MAIL)).
```

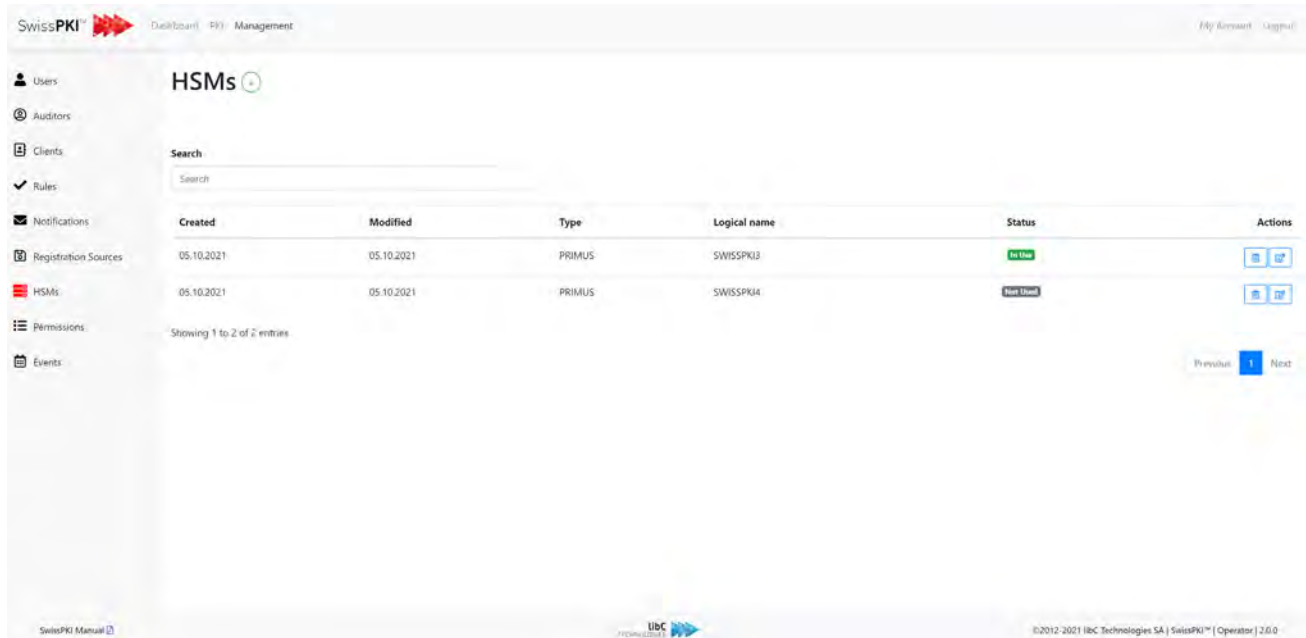
## 12.2.7 HSMs





SwissPKI integrates with Primus HSMs from Securosys SA which can either be used on premises or via a CloudsHSM service where all operational tasks related to maintenance and support of the devices are taken care of.

Additionally, SwissPKI integrates with Thales LunaSA 7.x and CySec's ARCA HSMs

Partitions are used to store or generate cryptographic keys. The PKI is organized according to the number of keys and CA you plan to take into production. The HSM partitions are automatically replicated for backups and failover. The cloud service is already included, and the keys of PKI entities can use these partitions to conserve the keys and perform signature operations.

It is also possible to define user partitions to issue keys on HSMs for user groups or certificate recipients as opposed to keys that are generated through software as PKCS12 and are transmitted to PKI for certification. You can create as many partitions as needed. The hardware keys are distributed over different HSM partitions according to your CPS. As a typical use, Root Certification Authorities keys are generated on dedicated partitions. Other PKI entities such as Issuing CAs, DSS, OCSP or TSA may have their keys stored on a shared or dedicated partition.

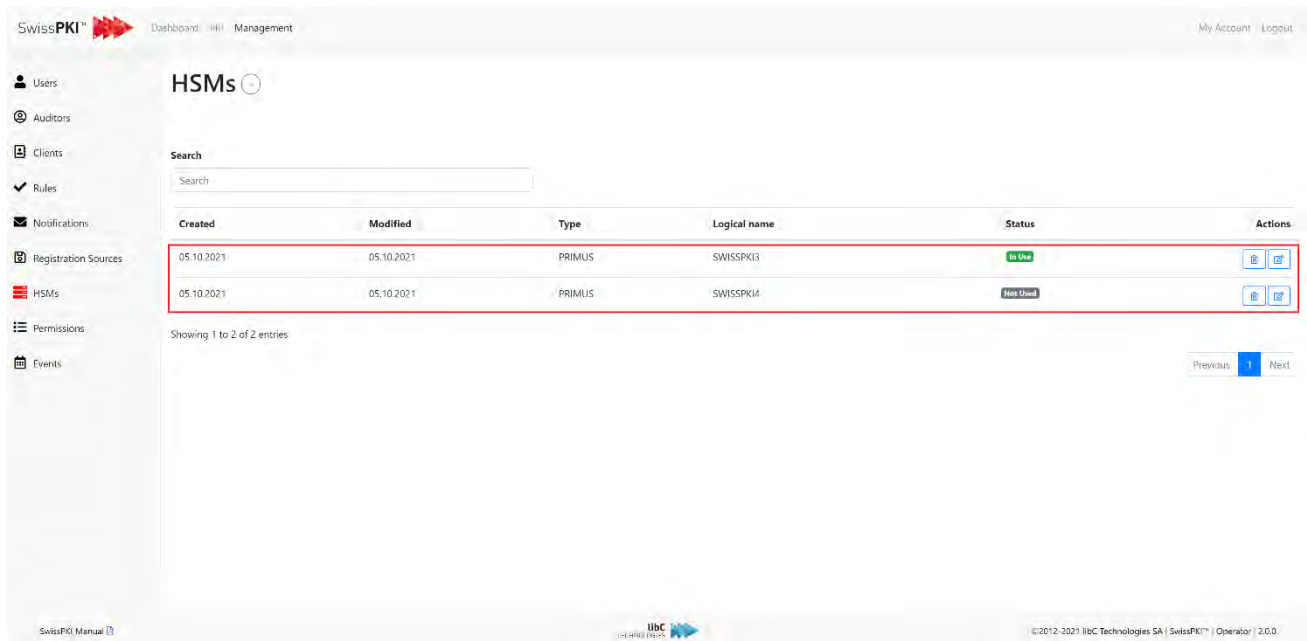


Created	Modified	Type	Logical name	Status	Actions
05.10.2021	05.10.2021	PRIMUS	SWISSPKI3	In Use	 
05.10.2021	05.10.2021	PRIMUS	SWISSPKI4	Not Used	 

### 12.2.7.1 Primus Partitions

You can create new HSM partitions by clicking on the add button located on the right of the page title. Additionally, you can do the following by using the buttons in the action's column of the table:

- Delete an HSM partition. Deleting an HSM partition will remove it if is not in use and mark it as retired if in use. A retired partition cannot be access in the key generation parameter setting of a Certificate Template.
- Edit an HSM partition
- Access the hosts page for an HSM partition



SwissPKI™ Dashboard HSM Management My Account Logout

HSMs

Search

Created	Modified	Type	Logical name	Status	Actions
05.10.2021	05.10.2021	PRIMUS	SWISSPKI3	In Use	[Edit] [Delete]
05.10.2021	05.10.2021	PRIMUS	SWISSPKI4	Not Used	[Edit] [Delete]

Showing 1 to 2 of 2 entries

Previous 1 Next

SwissPKI Manual

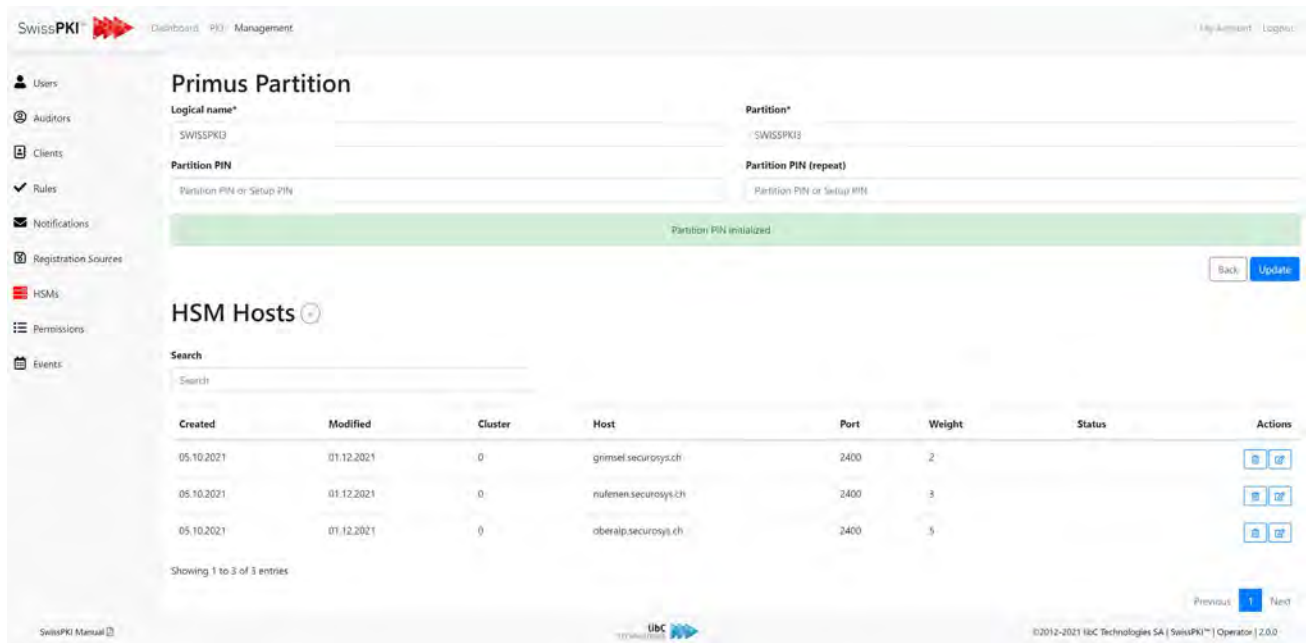
libC TECHNOLOGIES

©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.7.1.1 Create HSM Partition

To add a new HSM partition, you need to provide the following information:

Fields	Description
<b>Logical Name</b>	The HSM partition's logical name
<b>Partition</b>	The HSM partition's name
<b>Partition Pin</b>	The HSM partition's pin (Setup or Permanent PIN) Setup PIN: 29 characters (e.g., Xe8CV-CaP0m-4WLCA-FA8ej-CSG1S) Permanent PIN: 184 hex characters (ciphered PIN hex coded from DB export)



The screenshot shows the SWISSPKI management interface. The top navigation bar includes 'SWISSPKI', 'Dashboard', 'PKI Management', and 'Logout'. A sidebar on the left lists various management options: Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area is titled 'Primus Partition' and contains several input fields: 'Logical name\*' (filled with 'SWISSPKI3'), 'Partition\*' (filled with 'SWISSPKI3'), 'Partition PIN' (with a sub-label 'Partition PIN or Setup PIN'), and 'Partition PIN (repeat)' (with a sub-label 'Partition PIN or Setup PIN'). A green success message states 'Partition PIN initialized'. Below this, there are 'Back' and 'Update' buttons. The 'HSM Hosts' section features a search bar and a table with the following data:

Created	Modified	Cluster	Host	Port	Weight	Status	Actions
05.10.2021	01.12.2021	0	grimmel.secursys.ch	2400	2		[Refresh] [Delete]
05.10.2021	01.12.2021	0	nufemen.secursys.ch	2400	3		[Refresh] [Delete]
05.10.2021	01.12.2021	0	oberalp.secursys.ch	2400	5		[Refresh] [Delete]

At the bottom of the table, it says 'Showing 1 to 3 of 3 entries'. The footer includes 'libC TECHNOLOGIES' and '©2012-2021 libC Technologies SA | SWISSPKI™ | Operator | 2.0.0'.

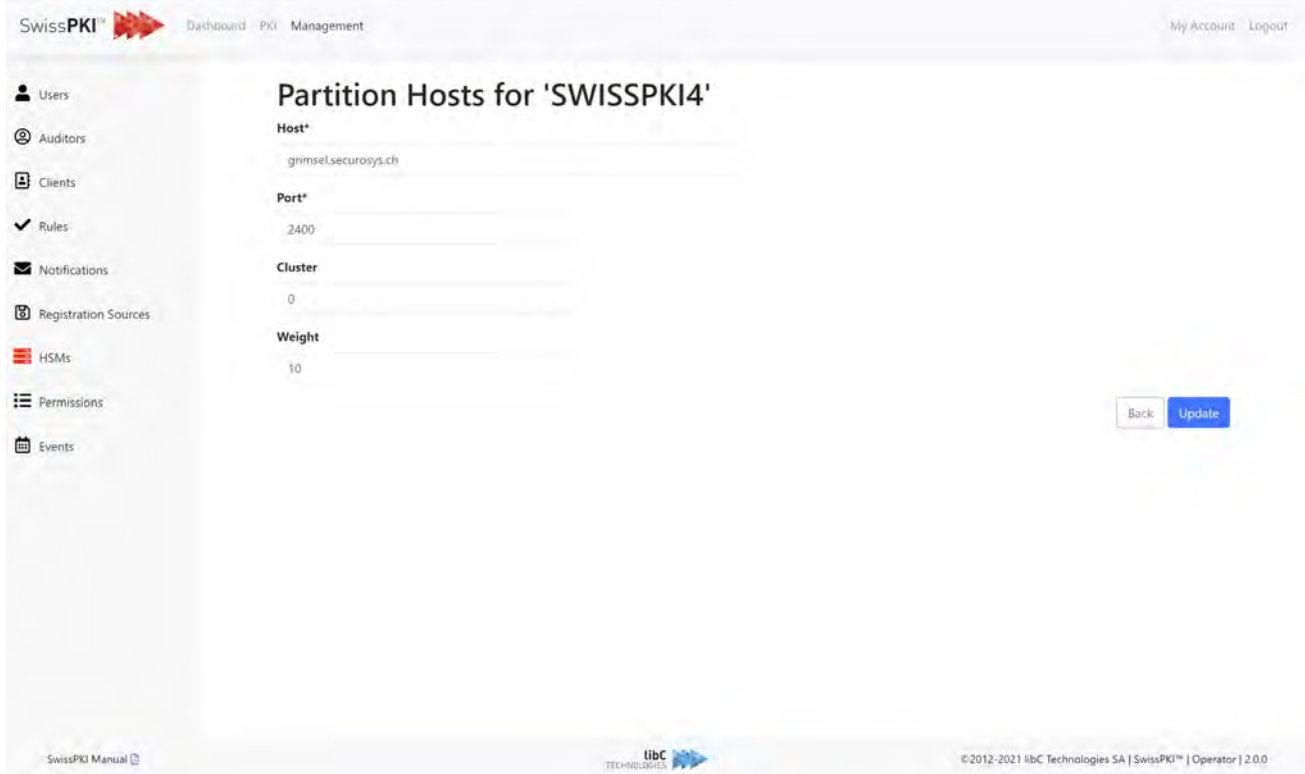


### 12.2.7.1.2 HSM Partition Hosts

Multiple hosts can be defined for each HSM partition. The list of hosts for a partition can be accessed by clicking on the host button of this partition.

Adding a new host is done by clicking on the add button located on the right of the page title.

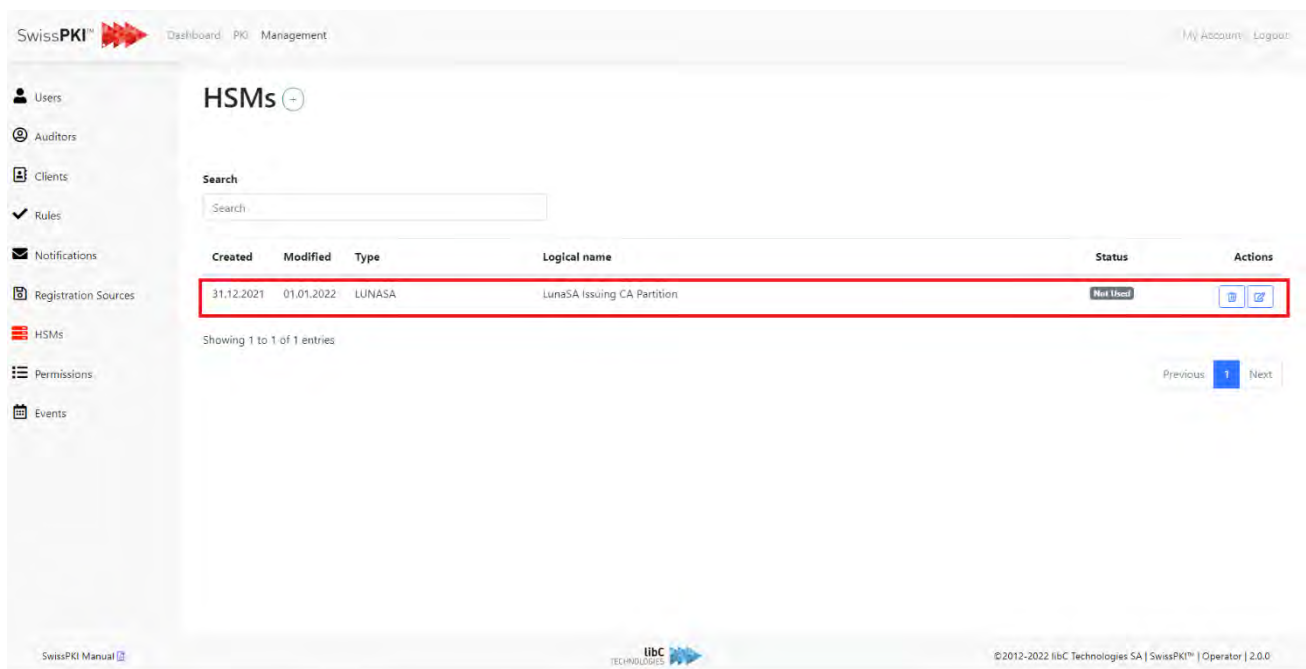
Fields	Description
<b>Host</b>	Partition host
<b>Port</b>	Port
<b>Cluster</b>	Value 0-n, multiple partitions can be clustered together by using the same index value
<b>Weight</b>	Weight 1-10: clustered partitions can be weighted together e.g.: <pre>partition_a:2400 cluster 0 weight 4 partition_b:2400 cluster 0 weight 3 partition_c:2400 cluster 0 weight 3 partition_d:2400 cluster 1 weight 10</pre>




### 12.2.7.2 LunaSA Partitions

You can create new HSM partitions by clicking on the add button located on the right of the page title. Additionally, you can do the following by using the buttons in the action's column of the table:




- Delete an HSM partition. Deleting an HSM partition will remove it if is not in use and mark it as retired if in use. A retired partition cannot be access in the key generation parameter setting of a Certificate Template.
- Edit an HSM partition




SwissPKI™  Dashboard PKI Management My Account Logout

**HSMs** -

Search

Created	Modified	Type	Logical name	Status	Actions
31.12.2021	01.01.2022	LUNASA	LunaSA Issuing CA Partition	Not Used	  

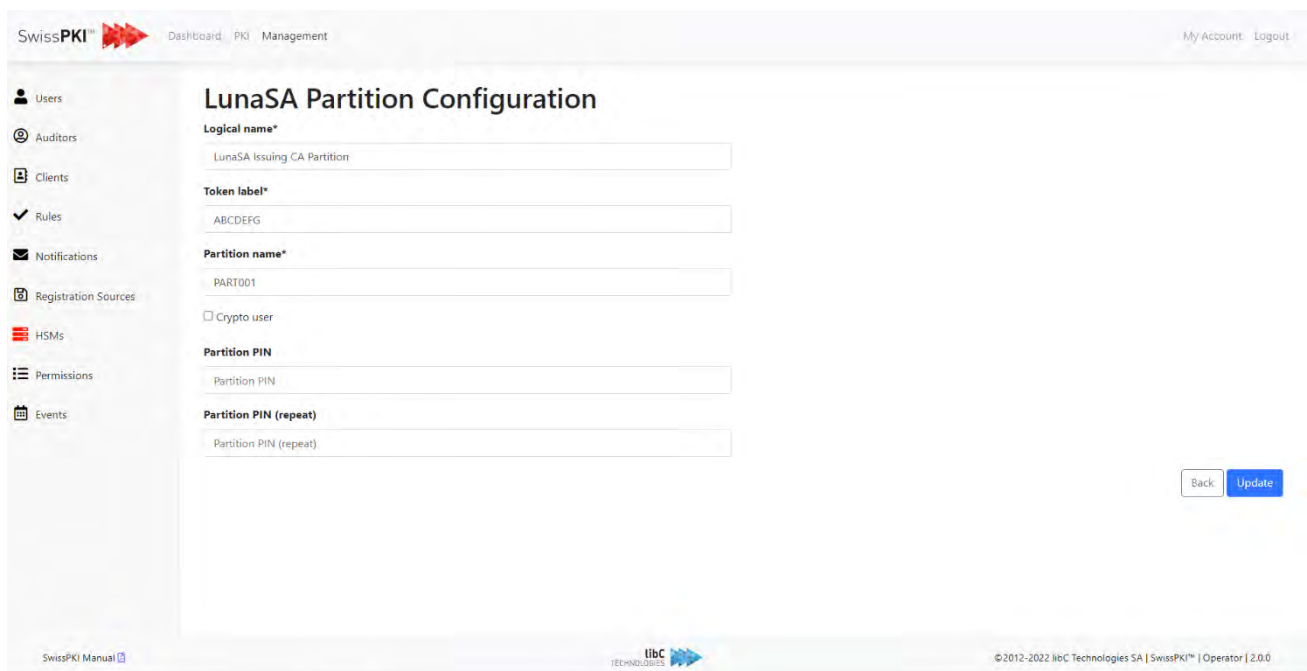
Showing 1 to 1 of 1 entries Previous **1** Next

SwissPKI Manual  ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.2.7.2.1 Create HSM Partition

To add a new HSM partition, you need to provide the following information:

Fields	Description
<b>Logical Name</b>	Partition logical name
<b>Token Label</b>	LunaSA Token label of the partition
<b>Partition name</b>	LunaSA partition name
<b>Crypto user</b>	When enabled, logs in with <code>CKU_CRYPT0_USER</code>  This prevents the application from creating or destroying keys, but the Crypto User can use the keys for crypto operations. The user is only available when using a PED and the "partition createUser" has already been executed using lush.
<b>Partition PIN</b>	Crypto service backend Hardware/Software
<b>Partition PIN (repeat)</b>	Vault service URL



The screenshot shows the 'LunaSA Partition Configuration' page in the SwissPKI management console. The page includes a sidebar with navigation options like Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area contains the following configuration fields:

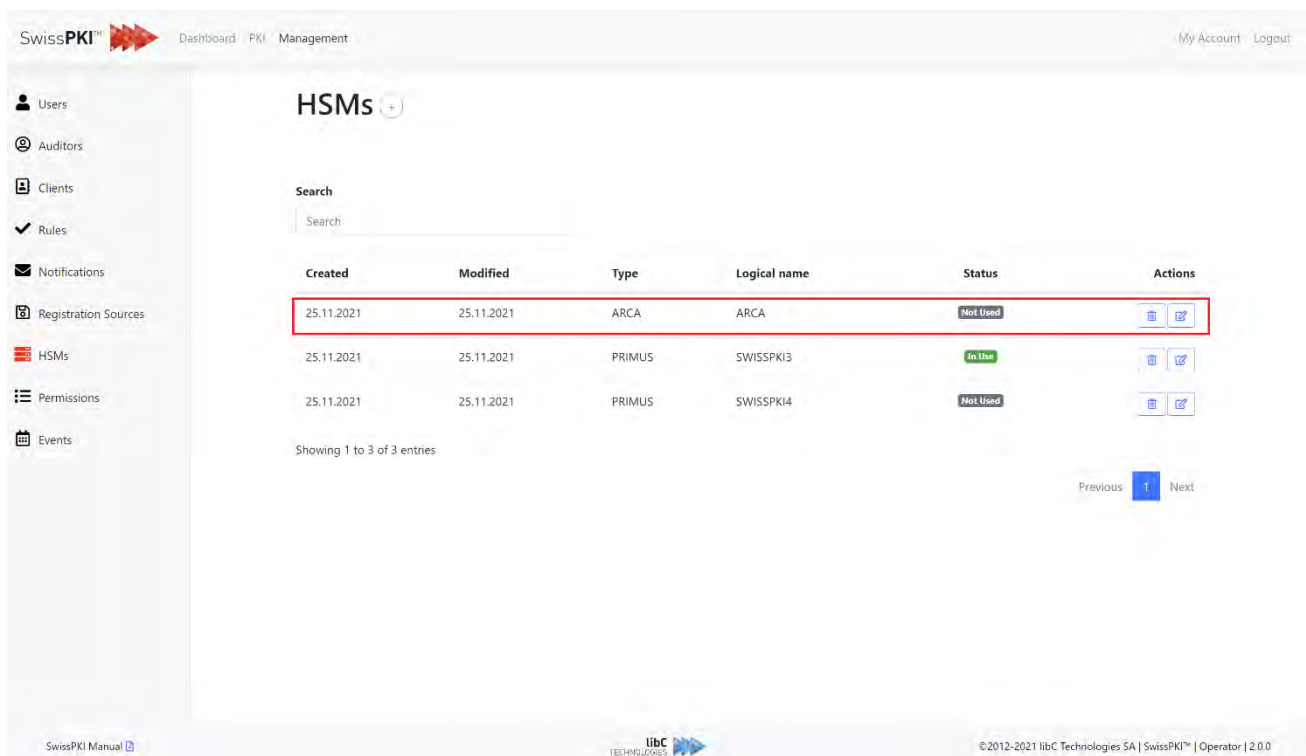
- Logical name\***: LunaSA Issuing CA Partition
- Token label\***: ABCDEFG
- Partition name\***: PART001
- Crypto user**
- Partition PIN**: Partition PIN
- Partition PIN (repeat)**: Partition PIN (repeat)

At the bottom right of the configuration area, there are 'Back' and 'Update' buttons. The footer of the page includes the libC Technologies logo, copyright information (© 2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0), and a link to the SwissPKI Manual.

### 12.2.7.3 ARCA Partitions

You can create new HSM partitions by clicking on the add button located on the right of the page title. Additionally, you can do the following by using the buttons in the action's column of the table:

- Delete an HSM partition. Deleting an HSM partition will remove it if is not in use and mark it as retired if in use. A retired partition cannot be access in the key generation parameter setting of a Certificate Template.
- Edit an HSM partition



The screenshot shows the 'HSMs' management page in the SwissPKI interface. The table contains the following data:

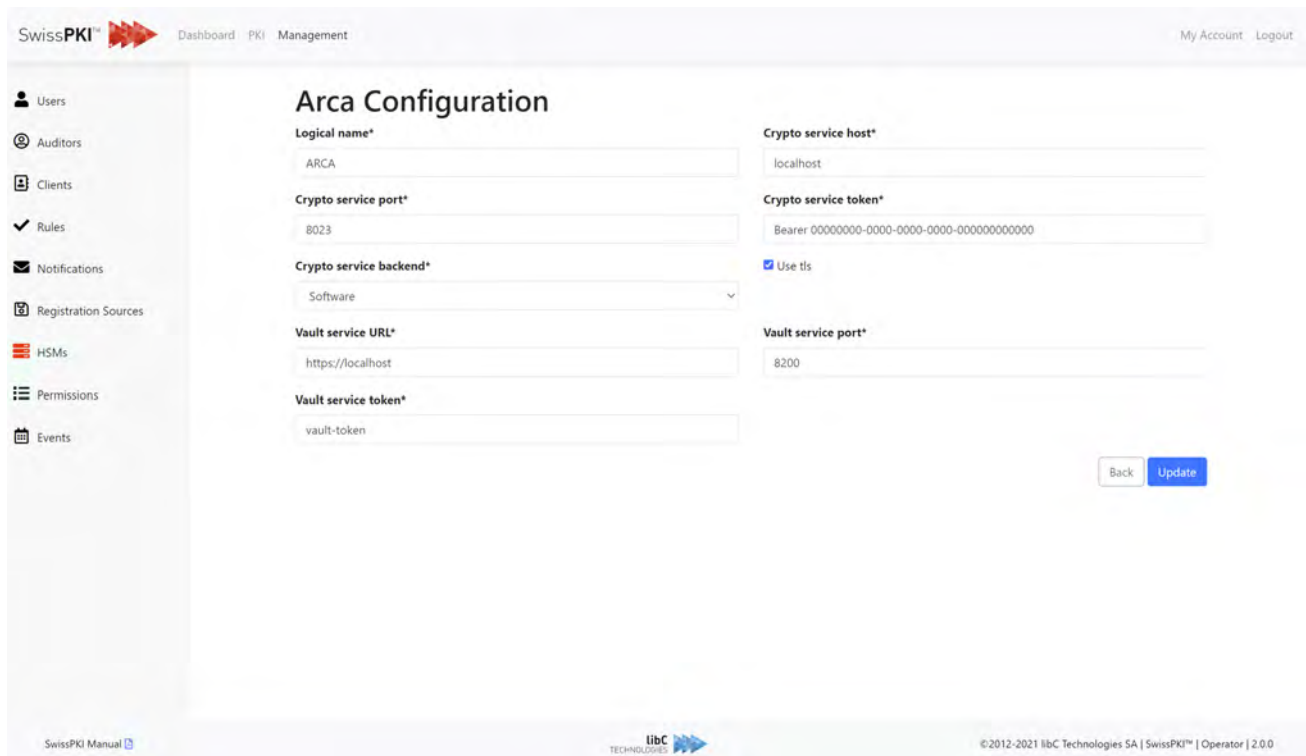
Created	Modified	Type	Logical name	Status	Actions
25.11.2021	25.11.2021	ARCA	ARCA	Not Used	[Delete] [Edit]
25.11.2021	25.11.2021	PRIMUS	SWISSPKI3	In Use	[Delete] [Edit]
25.11.2021	25.11.2021	PRIMUS	SWISSPKI4	Not Used	[Delete] [Edit]

The first row (ARCA partition) is highlighted with a red border. The interface also includes a search bar, a sidebar with navigation options, and pagination controls at the bottom.

### 12.2.7.3.1 Create HSM Partition

To add a new HSM partition, you need to provide the following information:

Fields	Description
<b>Logical Name</b>	Partition logical name
<b>Crypto service host</b>	Crypto service host/IP
<b>Crypto service port</b>	Crypto service port
<b>Crypto service token</b>	Crypto service bearer token
<b>Crypto service backend</b>	Crypto service backend Hardware/Software
<b>Vault service URL</b>	Vault service URL
<b>Vault service port</b>	Vault service port
<b>Vault service token</b>	Vault service token



The screenshot shows the 'Arca Configuration' page in the SwissPKI interface. The page includes a sidebar with navigation options like Users, Auditors, Clients, Rules, Notifications, Registration Sources, HSMs, Permissions, and Events. The main content area contains the following configuration fields:

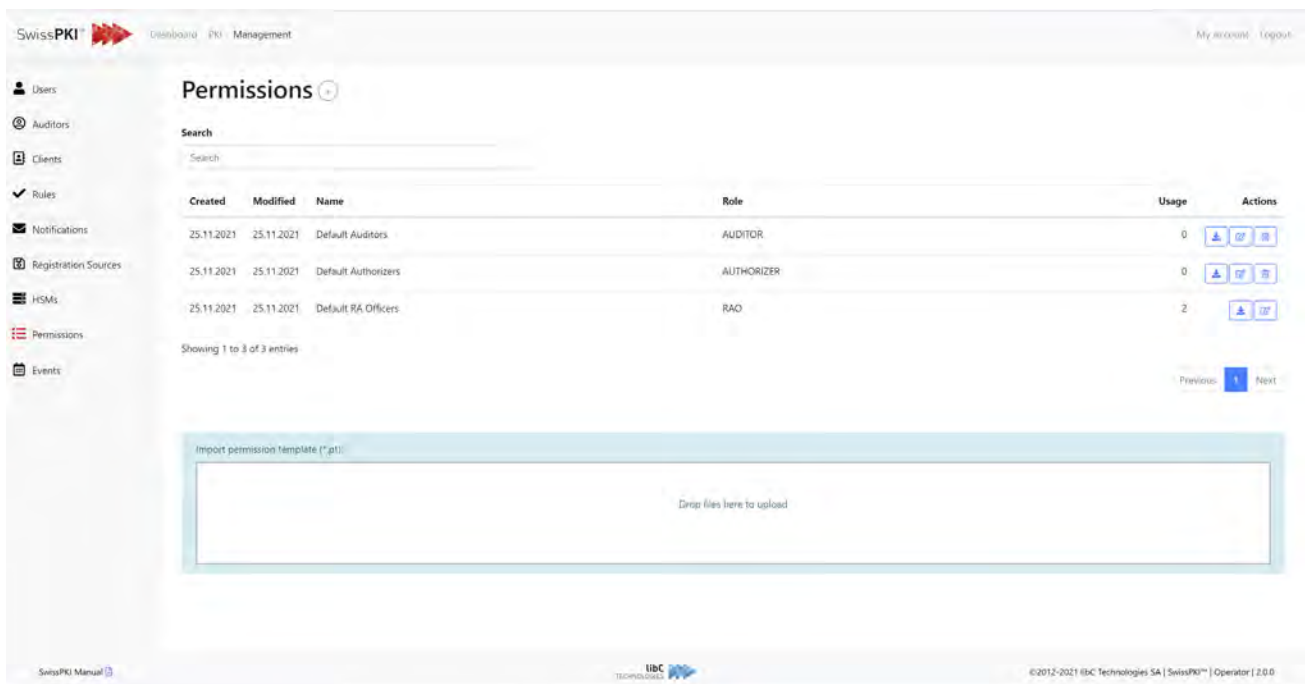
- Logical name\***: ARCA
- Crypto service port\***: 8023
- Crypto service backend\***: Software
- Vault service URL\***: https://localhost
- Vault service token\***: vault-token
- Crypto service host\***: localhost
- Crypto service token\***: Bearer 00000000-0000-0000-0000-000000000000
- Use tls
- Vault service port\***: 8200

At the bottom right of the configuration area, there are 'Back' and 'Update' buttons. The footer of the page includes 'SwissPKI Manual', the libC Technologies logo, and the copyright notice: '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

## 12.2.8 Permissions

On realm creation, three permission templates ‘**Default Auditors,**’ ‘**Default Authorizers**’ and ‘**Default RA officers**’ are generated with all permissions assigned to each template.

1. The add button located on the right of the page title is used to create new permission templates.
2. The export button is used to download a permission template.
3. The edit button is used to edit an existing permission template.
4. The delete button is used to delete permission templates. Note that the deletion of a permission template occurs only if the template is not in use (See usage column of the permissions table).
5. To import a permission template, drag & drop an exported permission template file to the upload area. You can only import permission templates of roles CAO, Authorizer or Auditor



The screenshot shows the 'Permissions' management interface in SwissPKI. The page title is 'Permissions' and it includes a search bar. A table lists three default permission templates:

Created	Modified	Name	Role	Usage	Actions
25.11.2021	25.11.2021	Default Auditors	AUDITOR	0	[Add] [Edit] [Delete]
25.11.2021	25.11.2021	Default Authorizers	AUTHORIZER	0	[Add] [Edit] [Delete]
25.11.2021	25.11.2021	Default RA Officers	RAO	2	[Add] [Edit]

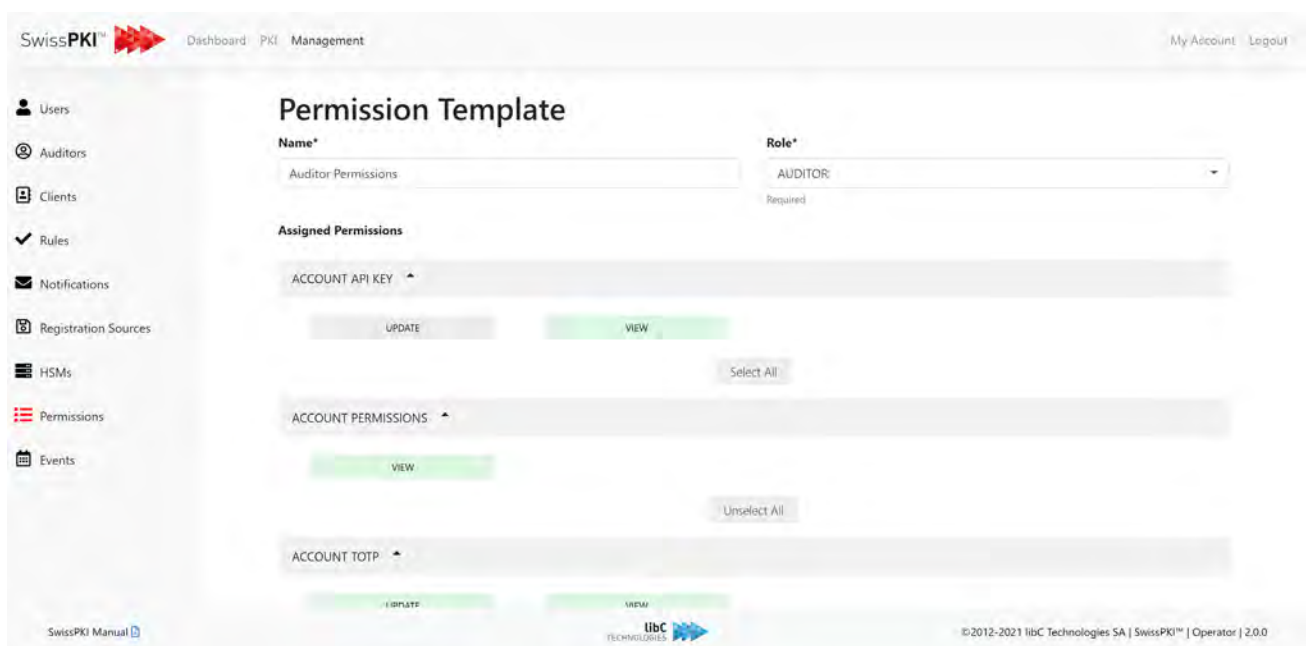
Below the table, there is an 'Import permission template (.ptt)' section with a large empty box and the text 'Drop files here to upload'.

**Note:** Please carefully read section 8.3 *End User Login Options* if you plan to rename the default permission templates.

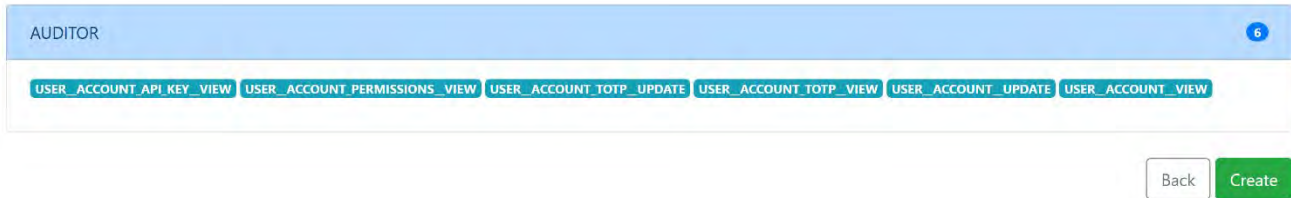
### 12.2.8.1 Create Permission Template

As a CA Operator, you have the possibility to create new permission templates for Auditor, Authorizer and RAO roles.

Fields	Description
<b>Permission Template Name</b>	The name of the permission template
<b>Permission Template Role</b>	The role associated to the permission template <ul style="list-style-type: none"> <li>• Auditor</li> <li>• Authorizer</li> <li>• RAO</li> </ul>
<b>Permission Template Assigned Permissions</b>	The selected permissions are granted to this permission template.



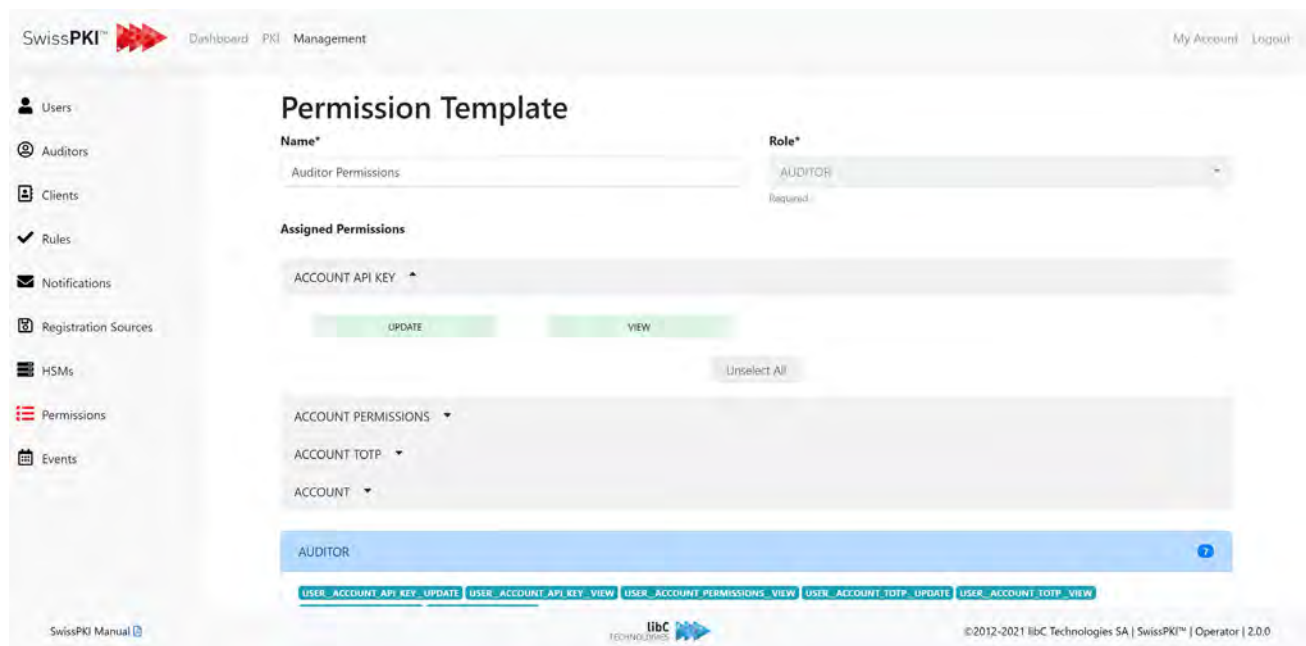
A list of all the selected permissions is available at the bottom of the page.



Click on the 'create' button to save your permission template.

### 12.2.8.2 Edit Permission Template

Click on the edit button of the desired permission template. Select/unselect the permissions you want to add/delete to the template and click on the 'update' button.



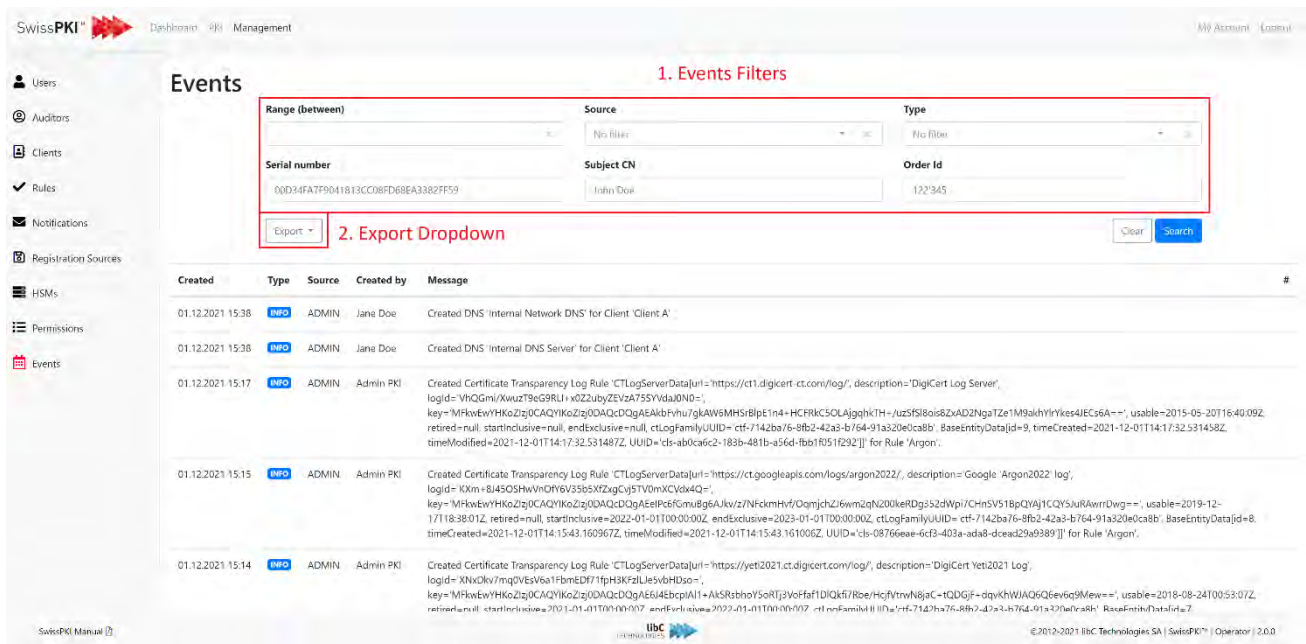


## 12.2.9 Events

The events page contains a list of all events that occurred for your realm.

Events logged to the database are also logged to the process log. Process logging is configurable as specified in section 4.3 *Logging*.

1. **Events Filters:** Events filters allow you to define criteria that will narrow down the events list. Applying the selected filters is done by clicking on the 'search' button. The clear button will reset all filters.
2. **Export Dropdown:** The Export dropdown allows to download the event list in either .csv or .xlsx file based on the search result but not limited to the maximum record returned in the view



The screenshot shows the SwissPKI Events management interface. At the top, there are navigation tabs for 'Dashboard', 'PKI', and 'Management'. A sidebar on the left contains various menu items like 'Users', 'Auditors', 'Clients', 'Rules', etc. The main content area is titled 'Events' and features a section for '1. Events Filters' with input fields for 'Range (between)', 'Source', 'Type', 'Serial number', 'Subject CN', and 'Order Id'. Below the filters is an 'Export' dropdown menu labeled '2. Export Dropdown' and 'Search' and 'Clear' buttons. Below the filters is a table of events with columns: 'Created', 'Type', 'Source', 'Created by', and 'Message'. The table contains four rows of event data, including details about DNS creation and Certificate Transparency Log Rule creation.

Created	Type	Source	Created by	Message
01.12.2021 15:38	INFO	ADMIN	Jane Doe	Created DNS 'Internal Network DNS' for Client 'Client A'
01.12.2021 15:38	INFO	ADMIN	Jane Doe	Created DNS 'Internal DNS Server' for Client 'Client A'
01.12.2021 15:17	INFO	ADMIN	Admin PKI	Created Certificate Transparency Log Rule 'CTLogServerData[url=https://ct1.digicert.ct.com/log/, description='DigiCert Log Server', logid='VhQGmi/Xwuz79eG9RLU+ x0ZzubyZEVzA755VdaJ0N0='], key='MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEe1Pc6fsmuBgeAjkvz/NFokmHv7/OgmjchZ/6wm2qN200keRDg552dWp/Chn5V518pQYAJ1CQY5uikAwrrDwg=-', usable=2015-05-20T16:40:09Z, retired=null, startInclusive=null, endExclusive=null, ctLogFamilyUID=ctf-7142ba/6-8fb2-42a3-b/64-91a320e0ca8b', BaseEntryData[id=9, timeCreated=2021-12-01T14:17:32.531458Z, timeModified=2021-12-01T14:17:32.531487Z, UID='c5-ab0cac62-183b-481b-a56d-fbb1f051f292']' for Rule 'Argon'.
01.12.2021 15:15	INFO	ADMIN	Admin PKI	Created Certificate Transparency Log Rule 'CTLogServerData[url=https://googleleaps.com/logs/argon2022/, description='Google 'Argon2022' log', logid='KXm+8I45OSHwVnDY6v35b5XZgCj5TV0mXCv4Q='], key='MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEe1Pc6fsmuBgeAjkvz/NFokmHv7/OgmjchZ/6wm2qN200keRDg552dWp/Chn5V518pQYAJ1CQY5uikAwrrDwg=-', usable=2019-12-17T18:38:01Z, retired=null, startInclusive=2022-01-01T00:00:00Z, endExclusive=2023-01-01T00:00:00Z, ctLogFamilyUID=ctf-7142ba/6-8fb2-42a3-b/64-91a320e0ca8b', BaseEntryData[id=8, timeCreated=2021-12-01T14:15:43.160967Z, timeModified=2021-12-01T14:15:43.161006Z, UID='c8-08766eee-6cf3-403a-ada8-dcead29a9389']' for Rule 'Argon'.
01.12.2021 15:14	INFO	ADMIN	Admin PKI	Created Certificate Transparency Log Rule 'CTLogServerData[url=https://yet2021.ct.digicert.com/log/, description='DigiCert Yet2021 Log', logid='XlvClkv7mqdVEsv8a1f8m5DF71fpH3kFzUJesv6H0se-', key='MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE64E8cpAl1+AKSRbhoY5oRTj3VoFaf1DlQk7/Roe/Hc/fvtnW8jpc+TQDGf+dqkKWIAQ6Q6evsq9Mew=-', usable=2018-08-24T00:53:07Z, retired=null, startInclusive=2021-01-01T00:00:00Z, endExclusive=2022-01-01T00:00:00Z, ctLogFamilyUID=ctf-7142ba/6-8fb2-42a3-b/64-91a320e0ca8b', BaseEntryData[id=7, timeCreated=2021-12-01T14:15:43.160967Z, timeModified=2021-12-01T14:15:43.161006Z, UID='c8-08766eee-6cf3-403a-ada8-dcead29a9389']' for Rule 'Argon'.

### 12.2.9.1 Events Filters

Fields	Description
<b>Range (between)</b>	The range filters allow you to define a range of dates. All events that happened during this range are displayed in the list.
<b>Source</b>	<ul style="list-style-type: none"> <li>• Air Gaped: Offline CA related events</li> <li>• ACME: ACME DNS validation and order creation</li> <li>• Issuance: certificate issuance</li> <li>• TSA: TSA related events</li> <li>• DSS: DSS related events</li> <li>• OCSP: OCSP related events</li> <li>• CMP: CMP related events</li> <li>• CRL: CRL related events</li> <li>• Admin: Administration related events</li> <li>• Authorization: Authorization related events</li> <li>• Renewal: Renewal related events</li> <li>• Recovery: Recovery related events</li> <li>• Revocation: Revocation related events</li> <li>• Publisher: Publisher related events</li> <li>• Login: Login related events</li> <li>• Microsoft: Microsoft related events</li> <li>• SNOW: SNOW related events</li> <li>• HSM: HSM related events</li> <li>• SCEP: SCEP related events</li> <li>• EMail: Email related events</li> <li>• Certificate Order: Order related events</li> <li>• Cross Sign: Cross Sign related events</li> </ul>
<b>Type</b>	Defines the type of the event. <ul style="list-style-type: none"> <li>• Info</li> <li>• Warning</li> <li>• Error</li> </ul>
<b>Serial Number</b>	Filters events by certificate serial number
<b>Subject CN</b>	Filters events by certificate subject CN

### 12.2.9.2 Events Fields

Fields	Description
<b>Created</b>	Date/Time of the event
<b>Type</b>	Event information level: Information Warning Error
<b>Source</b>	see Event Sources table below
<b>Created By</b>	User who created the event
<b>Message</b>	Event message (EN/FR/DE)

### 12.2.9.3 Events Sources

Source	Description
<b>Air Gapped</b>	<p>Issuance and imports of Air Gapped CA to Offline CA events for CA , CSR, CRL and ICA Issuance.</p> <p>Certificate Enrollment Service processing jobs.</p> <ul style="list-style-type: none"> <li> <b><i>AirGappedOfflineCACRLJob</i></b>  <i>Generation of an Air Gapped CRL request</i>            'Reference UUID' processed Air Gapped Offline CA CRL Request &lt;message&gt;            'Reference UUID' aborted Air Gapped Offline CA CRL Request &lt;message&gt;            'Reference UUID' handling Air Gapped Offline CA CRL Request &lt;message&gt;            'Reference UUID' error handling Air Gapped Offline CA CRL Request Job &lt;message&gt;         </li> <li> <b><i>AirGappedOfflineCACertificateIssuanceJob</i></b>  <i>Generation of an Air Gapped CA certificate request</i>            'Reference UUID' processed Air Gapped Offline CA Certificate Issuance &lt;message&gt;            'Reference UUID' aborted Air Gapped Offline CA Certificate Issuance &lt;message&gt;            'Reference UUID' handling Air Gapped Offline CA Certificate Issuance &lt;message&gt;            'Reference UUID' error handling Air Gapped Offline CA Certificate Issuance Job &lt;message&gt;         </li> </ul>

	<ul style="list-style-type: none"> <li>• <b><i>AirGapedOfflineCASubCAIssuanceJob</i></b> <i>Generation of an Air Gapped Sub CA signature request</i> 'Reference UUID' processed Air Gapped Offline CA Issue ICA &lt;message&gt; 'Reference UUID' aborted Air Gapped Offline CA Issue ICA &lt;message&gt; 'Reference UUID' handling Air Gapped Offline CA Issue ICA &lt;message&gt; 'Reference UUID' error handling Air Gapped Offline CA Issue ICA Job &lt;message&gt;</li> <li>• <b><i>AirGapedOfflineCALastCRLJob</i></b> <i>Generation of an Air Gapped Last CRL request</i> 'Reference UUID' processed Air Gapped Offline CA CRL Request &lt;message&gt; 'Reference UUID' aborted Air Gapped Offline CA CRL Request &lt;message&gt; 'Reference UUID' handling Air Gapped Offline CA CRL Request &lt;message&gt; 'Reference UUID' error handling Air Gapped Offline CA CRL Request Job &lt;message&gt;</li> <li>• <b><i>AirGapedOfflineCAXSignJob</i></b> <i>Generation of an Air Gapped Cross Signed request</i> 'Reference UUID' processed Air Gapped Offline CA Certificate Issuance &lt;message&gt; 'Reference UUID' aborted Air Gapped Offline CA Certificate Issuance &lt;message&gt; 'Reference UUID' handling Air Gapped Offline CA Certificate Issuance &lt;message&gt; 'Reference UUID' error handling Air Gapped Offline CA Certificate Issuance Job &lt;message&gt;</li> </ul>
<b>Issuance</b>	Issuance of a certificate. Message contains certificate order and certificate information such as Subject DN, Serial number and validity as well as Issuing CA and who issued the certificate.
<b>TSA</b>	<p>TSA automatic certificate renewal event</p> <p>Below are described Time Stamp Authority (TSA) Jobs:</p> <ul style="list-style-type: none"> <li>• <b><i>TSARenewalJob</i></b> <i>Automatic TSA certificate renewal task</i> 'Reference UUID' error handling Automatic TSA Renewal Job &lt;message&gt; 'Reference UUID' handling Automatic TSA Renewal &lt;message&gt; 'Reference UUID' aborted Automatic TSA Renewal &lt;message&gt; 'Reference UUID' processed Automatic TSA Renewal &lt;message&gt;</li> </ul>
<b>DSS</b>	<p>DSS automatic certificate renewal event</p> <p>Below are described (DSS) Jobs:</p> <ul style="list-style-type: none"> <li>• <b><i>DSSRenewalJob</i></b> <i>Automatic DSS certificate renewal task</i> 'Reference UUID' error handling Automatic DSS Renewal Job &lt;message&gt; 'Reference UUID' handling Automatic DSS Renewal &lt;message&gt; 'Reference UUID' aborted Automatic DSS Renewal &lt;message&gt;</li> </ul>

	'Reference UUID' processed Automatic DSS Renewal <message>
<b>OCSP</b>	<p>OCSP automatic certificate renewal event</p> <p>Below are described OCSP Jobs:</p> <ul style="list-style-type: none"> <li> <b>OCSPRenewalJob</b>  <i>Automatic OCSP certificate renewal task</i>            'Reference UUID' error handling Automatic OCSP Renewal Job &lt;message&gt;            'Reference UUID' handling Automatic OCSP Renewal &lt;message&gt;            'Reference UUID' aborted Automatic OCSP Renewal &lt;message&gt;            'Reference UUID' processed Automatic OCSP Renewal &lt;message&gt;         </li> </ul>
<b>CMP</b>	<p>CMP automatic certificate renewal and certificate revocation events</p> <p>Below are described CMP Jobs:</p> <ul style="list-style-type: none"> <li> <b>CMPRenewalJob</b>  <i>Automatic CMP certificate renewal task</i>            'Reference UUID' error handling Automatic CMP Renewal Job &lt;message&gt;            'Reference UUID' handling Automatic CMP Renewal &lt;message&gt;            'Reference UUID' aborted Automatic CMP Renewal &lt;message&gt;            'Reference UUID' processed Automatic CMP Renewal &lt;message&gt;         </li> </ul>
<b>CRL</b>	<p>CRL generation (manual and from CRL publication rules, CRL schedule updates and creation and automatic CRL job processing.</p> <p>Includes information about the CA generating the CRL, it is serial number and validity dates.</p> <p>Below are described CRL Jobs:</p> <ul style="list-style-type: none"> <li> <b>GenerateLastCRLJob</b>  <i>Last CRL issuance task (manual)</i>            'Reference UUID' error handling Generate Last CRL Job &lt;message&gt;            'Reference UUID' handling Generate Last CRL &lt;message&gt;            'Reference UUID' aborted Generate Last CRL &lt;message&gt;            'Reference UUID' processed Generate Last CRL &lt;message&gt;         </li> <li> <b>GenerateCRLFromRuleJob</b>  <i>CRL issuance task produce by a CRL generation Rule</i>            'Reference UUID' error handling Generate CRL from Rule Job &lt;message&gt;            'Reference UUID' handling Generate CRL from Rule &lt;message&gt;            'Reference UUID' aborted Generate CRL from Rule &lt;message&gt;            'Reference UUID' processed Generate CRL from Rule &lt;message&gt;         </li> </ul>

	<ul style="list-style-type: none"> <li>• <b>GenerateCRLJob</b> <i>CRL issuance manual task</i> 'Reference UUID' error handling Generate CRL Job &lt;message&gt; 'Reference UUID' handling Generate CRL &lt;message&gt; 'Reference UUID' aborted Generate CRL &lt;message&gt; 'Reference UUID' processed Generate CRL &lt;message&gt;</li> <li>• <b>RegisterCRLRuleJob</b> <i>Modification of a CRL publication rule</i> 'Reference UUID' error fetching CRL Rule &lt;message&gt; 'Reference UUID' registering CRL Rule &lt;message&gt; 'Reference UUID' aborted CRL Rule register &lt;message&gt; 'Reference UUID' registered CRL Rule &lt;message&gt;</li> <li>• <b>UnregisterCRLForRuleJob</b> <i>Deletion of a CRL publication rule</i> 'Reference UUID' error fetching CRL Rule &lt;message&gt; 'Reference UUID' aborted CRL Rule update &lt;message&gt; 'Reference UUID' unregistered CRL Rule with Id &lt;message&gt; 'Reference UUID' unregistering CRL Rule with Id &lt;message&gt;</li> <li>• <b>UpdateCRLForRuleJob</b> <i>Update of a CRL publication rule</i> 'Reference UUID' error fetching CRL Rule &lt;message&gt; 'Reference UUID' updating CRL Rule &lt;message&gt; 'Reference UUID' aborted CRL Rule &lt;message&gt; 'Reference UUID' updated CRL Rule &lt;message&gt;</li> </ul>
<b>Admin</b>	All Create, Read, Update and Delete operations performed by PKI_ADMIN and/or CA Operator roles
<b>Authorization</b>	<p>All relevant authorization actions related to</p> <ul style="list-style-type: none"> <li>• Certificate Order, Renewal, Revocation authorization</li> <li>• Kerberos, OIDC, LDAP and TOTP events during login phase</li> <li>• All updates of Permission Templates performed by PKI_ADMIN or CA Operator</li> <li>• All User updates relative to API Key management, token confirmation and User onboarding operations</li> </ul>
<b>Renewal</b>	All operations relative to automatic certificate order renewal and reminder notifications
<b>Recovery</b>	All operations relative to key recovery (Microsoft only) requests
<b>Revocation</b>	<p>All operations related to certificate revocation</p> <p>Below are described Revocation Jobs:</p>

	<ul style="list-style-type: none"> <li>• <b>RAOAuthorizeCertificateRevokeJob</b> <i>RA revocation request necessitating an authorization</i> 'Reference UUID' error handling Certificate Revocation Authorization Job &lt;message&gt; 'Reference UUID' handling Certificate Revocation Authorization &lt;message&gt; 'Reference UUID' aborted Certificate Revocation Authorization &lt;message&gt; 'Reference UUID' processed Certificate Revocation Authorization &lt;message&gt;</li> </ul>
<b>Publisher</b>	<p>All operations related to Certificate and CRL publication/un-publication (LDAP, file system and SFTP)</p> <p>Below are described Publisher Jobs:</p> <ul style="list-style-type: none"> <li>• <b>PublishCRLJob</b> <i>Publication of a CRL to an LDAP/SFTP/File system</i> 'Reference UUID' error handling CRL Publish Job &lt;message&gt; 'Reference UUID' handling CRL Publish &lt;message&gt; 'Reference UUID' aborted CRL Publish &lt;message&gt; 'Reference UUID' processed CRL Publish &lt;message&gt;</li> <li>• <b>ManualPublishCertificateJob</b> <i>Manual Publication of a CRL to an LDAP/SFTP/File system</i> 'Reference UUID' error handling Certificate Publish Job &lt;message&gt; 'Reference UUID' handling Certificate Publish &lt;message&gt; 'Reference UUID' aborted Certificate Publish &lt;message&gt; 'Reference UUID' processed Certificate Publish &lt;message&gt;</li> <li>• <b>ManualUnPublishCertificateJob</b> <i>Manual un-publication of a CRL to an LDAP/SFTP/File system</i> 'Reference UUID' error handling Certificate un-publish Job &lt;message&gt; 'Reference UUID' handling Certificate un-publish &lt;message&gt; 'Reference UUID' aborted Certificate un-publish &lt;message&gt; 'Reference UUID' processed Certificate un-publish &lt;message&gt;</li> <li>• <b>PublishCertificateOrderJob</b> <i>Publication of a Certificate to an LDAP/SFTP/File system after issuance</i> 'Reference UUID' error handling Certificate Publish Job &lt;message&gt; 'Reference UUID' handling Certificate Publish &lt;message&gt; 'Reference UUID' aborted Certificate Publish &lt;message&gt; 'Reference UUID' processed Certificate Publish &lt;message&gt;</li> </ul>
<b>Login</b>	All user login and logout operations including failed logins and/or locked accounts
<b>Microsoft</b>	All Microsoft CEP operations Certificate Enrollment Service processing jobs.

	<ul style="list-style-type: none"> <li>• <b>MicrosoftEnrolmentPoliciesJob</b> <i>Microsoft AD request for the list of certificate policies assigned to the MSCA service</i> 'Reference UUID' error handling Microsoft CEP Job &lt;message&gt; 'Reference UUID' handling Automatic Microsoft CEP &lt;message&gt; 'Reference UUID' aborted Automatic Microsoft CEP &lt;message&gt; 'Reference UUID' processed Automatic Microsoft CEP &lt;message&gt;</li> </ul>
<p><b>SNOW</b></p>	<p>All related SNOW operations</p> <ul style="list-style-type: none"> <li>• <b>SNOWRevokeAllClientCertificateJob</b> <i>SNOW request for revoking all Client certificates (i.e., Client deleted via SNOW)</i> 'Reference UUID' processed Client Certificate Revocation &lt;message&gt; 'Reference UUID' aborted Client Certificate Revocation &lt;message&gt; 'Reference UUID' handling Client Certificate Revocation &lt;message&gt; 'Reference UUID' error handling Client Certificate Revocation Job &lt;message&gt;</li> <li>• <b>SNOWRevokePolicyMappingCertificateJob</b> <i>SNOW request for removing Client products (i.e., Client MPKI downgrade)</i> 'Reference UUID' processed Policy Mapping Certificate Revocation &lt;message&gt; 'Reference UUID' aborted Policy Mapping Certificate Revocation &lt;message&gt; 'Reference UUID' handling Client Certificate Revocation &lt;message&gt; 'Reference UUID' handling Policy Mapping Certificate Revocation &lt;message&gt; 'Reference UUID' error handling Policy Mapping Certificate Revocation Job &lt;message&gt;</li> </ul>
<p><b>HSM</b></p>	<p>All related HSM operations such as Partition creation and update. HSM Partition PIN reset.</p> <p>Jobs responsible for HSM operations.</p> <ul style="list-style-type: none"> <li>• <b>HSMPINResetJob</b> <i>HSM PIN Reset request</i> 'Reference UUID' error handling HSM PIN Reset Job &lt;message&gt; 'Reference UUID' handling HSM PIN Reset &lt;message&gt; 'Reference UUID' aborted HSM PIN Reset &lt;message&gt; 'Reference UUID' processed HSM PIN Reset &lt;message&gt;</li> <li>• <b>HSMPINResetSchedulerJob</b> <i>HSM PIN Reset request</i> 'Reference UUID' error handling HSM PIN Reset Job &lt;message&gt; 'Reference UUID' handling HSM PIN Reset &lt;message&gt; 'Reference UUID' aborted HSM PIN Reset &lt;message&gt; 'Reference UUID' processed HSM PIN Reset &lt;message&gt;</li> </ul>



<p><b>SCEP</b></p>	<p>All SCEP protocol operations</p> <p>Jobs responsible for Simple Certificate Enrollment Protocol (SCEP).</p> <ul style="list-style-type: none"> <li>• <b>SCEPPKIOperationJob</b> <i>SCEP request for certificate enrollment</i> 'Reference UUID' error handling SCEP PKI Operation Job &lt;message&gt; 'Reference UUID' handling SCEP PKI Operation &lt;message&gt; 'Reference UUID' aborted SCEP PKI Operation &lt;message&gt; 'Reference UUID' processed SCEP PKI Operation &lt;message&gt;</li> </ul>
<p><b>Email</b></p>	<p>All Email notification types related to notification rules and templates</p> <ul style="list-style-type: none"> <li>• <b>SendEmailJob</b> <i>Send mal task</i> 'Reference UUID' error handling SendMail Job &lt;message&gt; 'Reference UUID' handling SendMail &lt;message&gt; 'Reference UUID' aborted SendMail &lt;message&gt; 'Reference UUID' processed SendMail &lt;message&gt; 'Reference UUID' retrying SendMail &lt;message&gt;</li> </ul>
<p><b>Cross Sign</b></p>	<p>All cross signing operations on a CA</p> <ul style="list-style-type: none"> <li>• <b>CrossSignedCSRJob</b> <i>Cross sign request (produces a CSR for the CA)</i> 'Reference UUID' error handling Cross Sign CSR Job &lt;message&gt; 'Reference UUID' handling Cross Sign CSR &lt;message&gt; 'Reference UUID' aborted Cross Sign CSR &lt;message&gt; 'Reference UUID' processed Cross Sign CSR &lt;message&gt;</li> </ul>
<p><b>Certificate Order</b></p>	<p>All Certificate Order processing:</p> <p>Certificate order processing status information is recorded at each process step.</p> <ul style="list-style-type: none"> <li>• <b>FinalizeIssueCertificateParentJob</b> <i>Finalization of the certificate order issuance parent Job starting the child finalization tasks</i> 'Reference UUID' error handling Finalize Certificate Issuance Parent Job &lt;message&gt; 'Reference UUID' handling Finalize Certificate Issuance Parent &lt;message&gt; 'Reference UUID' aborted Finalize Certificate Issuance Parent &lt;message&gt; 'Reference UUID' processed Finalize Certificate Issuance Parent &lt;message&gt;</li> </ul>

- **CertificateRenewalValidationJob**  
*Validation of a renewal order. Checks whether an authorization is required and the type of renewal for process routing purpose.*  
 'Reference UUID' error handling Certificate Order Renewal Validation Job <message>  
 'Reference UUID' handling Certificate Order Renewal Validation <message>  
 'Reference UUID' aborted Certificate Order Renewal Validation <message>  
 'Reference UUID' processed Certificate Order Renewal Validation <message>
- **GenerateTBSJob**  
*Generate TBS structure for pre validation tasks*  
 'Reference UUID' error handling Generate TBS Job <message>  
 'Reference UUID' handling Generate TBS <message>  
 'Reference UUID' aborted Generate TBS <message>  
 'Reference UUID' processed Generate TBS <message>
- **IssueCertificateJob**  
*Effective certificate issuance*  
 'Reference UUID' error handling Issue Certificate Job <message>  
 'Reference UUID' handling Issue Certificate <message>  
 'Reference UUID' aborted Issue Certificate <message>  
 'Reference UUID' processed Issue Certificate <message>
- **KeyValidationJob**  
*Validation of the CSR and/or PKCS#12/HSM key pair.*  
*Validation of key size and algorithms*  
 'Reference UUID' error handling Key Validation Job <message>  
 'Reference UUID' handling Key Validation <message>  
 'Reference UUID' aborted Key Validation <message>  
 'Reference UUID' processed Key Validation <message>
- **PostIssueCertificateParentJob(seems to be equal Pre)**  
*Parent job starting all post issuance tasks*  
 'Reference UUID' error handling Pre Issue Certificate Parent Job <message>  
 'Reference UUID' handling Pre Issue Certificate Parent <message>  
 'Reference UUID' aborted Pre Issue Certificate Parent <message>  
 'Reference UUID' processed Pre Issue Certificate Parent <message>
- **PreIssueCertificateParentJob**  
*Parent job starting all pre issuance tasks*  
 'Reference UUID' error handling Pre Issue Certificate Parent Job <message>  
 'Reference UUID' handling Pre Issue Certificate Parent <message>  
 'Reference UUID' aborted Pre Issue Certificate Parent <message>  
 'Reference UUID' processed Pre Issue Certificate Parent <message>
- **NotifyIssuedCertificateJob**  
*Notify recipients of issued certificate*  
 'Reference UUID' error handling Notify Issued Certificate Order Job <message>  
 'Reference UUID' handling Notify Issued Certificate Order <message>  
 'Reference UUID' aborted Notify Issued Certificate Order <message>  
 'Reference UUID' processed Notify Issued Certificate Order <message>

- **NotifyRenewalCertificateJob**  
*Notify recipient of renewed certificate*  
 'Reference UUID' error handling Notify Renewed Order Job <message>  
 'Reference UUID' handling Notify Renewed Order <message>  
 'Reference UUID' aborted Notify Renewed Order <message>  
 'Reference UUID' processed Notify Renewed Order <message>
- **RevokeRenewedCertificateJob**  
*Revocation of renewed certificate*  
 'Reference UUID' error handling Revoke Renewed Order Job <message>  
 'Reference UUID' handling Revoke Renewed Order <message>  
 'Reference UUID' aborted Revoke Renewed Order <message>  
 'Reference UUID' processed Revoke Renewed Order <message>
- **SubmitCertificateOrderJob**  
*Certificate Order Creation (new empty certificate order)*  
 'Reference UUID' error handling Submit Certificate Order Job <message>  
 'Reference UUID' handling Submit Certificate Order <message>  
 'Reference UUID' aborted Submit Certificate Order <message>  
 'Reference UUID' processed Submit Certificate Order <message>
- **PreValidationParentJob**  
*Parent validation job starting child validation processes*  
 'Reference UUID' error handling Certificate Order Pre Validation Parent Job <message>  
 'Reference UUID' handling Certificate Order Pre Validation Parent <message>  
 'Reference UUID' aborted Certificate Order Pre Validation Parent <message>  
 'Reference UUID' processed Certificate Order Pre Validation Parent <message>
- **PolicyValidationJob**  
*Certificate policy validation (static content and runtime value validation against business rules)*  
 'Reference UUID' error handling Policy Validation Job <message>  
 'Reference UUID' handling Policy Validation <message>  
 'Reference UUID' aborted Policy Validation <message>  
 'Reference UUID' processed Policy Validation <message>
- **CAACheckValidationJob**  
*CAA validation check*  
 'Reference UUID' error handling CAA Validation Job <message>  
 'Reference UUID' handling CAA Validation <message>  
 'Reference UUID' aborted CAA Validation <message>  
 'Reference UUID' processed CAA Validation <message>
- **DomainOwnerCheckValidationJob**  
*DNS Owner check validation*  
 'Reference UUID' error handling DNS Owner Check Validation Job <message>  
 'Reference UUID' handling DNS Owner Check Validation <message>  
 'Reference UUID' aborted DNS Owner Check Validation <message>  
 'Reference UUID' processed DNS Owner Check Validation <message>

- **EndUserEmailValidationJob**  
*End user email validation check*  
'Reference UUID' error handling End User Email Validation Job <message>  
'Reference UUID' handling End User Email Validation <message>  
'Reference UUID' aborted End User Email Validation <message>  
'Reference UUID' processed End User Email Validation <message>
- **PreLintingCertificateJob**  
*Certificate content pre linting*  
'Reference UUID' error handling Certificate Pre Linting Job <message>  
'Reference UUID' handling Certificate Pre Linting <message>  
'Reference UUID' aborted Certificate Pre Linting <message>  
'Reference UUID' processed Certificate Pre Linting <message>
- **CTLogPrecertPublicationJob**  
*CT Log pre cert handling, collect CT token for CT Log publication*  
'Reference UUID' error handling CT Log Pre Cert Job <message>  
'Reference UUID' handling CT Log Pre Cert <message>  
'Reference UUID' aborted CT Log Pre Cert <message>  
'Reference UUID' processed CT Log Pre Cert <message>
- **PostLintingCertificateJob**  
*Post certificate content linting*  
'Reference UUID' error handling Set P12 Pin Job <message>  
'Reference UUID' handling Set P12 Pin <message>  
'Reference UUID' aborted Set P12 Pin <message>  
'Reference UUID' processed Set P12 Pin <message>
- **PostPublishCertificateJob**  
*Publication of issued certificate to LDAP, file system or SFTP*  
Sends notification to Publisher service
- **CTLogPublicationJob**  
*Publication to the CT Log of the issued certificate*  
'Reference UUID' error handling CT Log Publication Job <message>  
'Reference UUID' handling CT Log Publication <message>  
'Reference UUID' aborted CT Log Publication <message>  
'Reference UUID' processed CT Log Publication <message>
- **CertificateOrderAuthorizationJob**  
*Authorization task for certificate issuance*  
'Reference UUID' error handling Certificate Order Authorization Job <message>  
'Reference UUID' handling Certificate Order Authorization <message>  
'Reference UUID' aborted Certificate Order Authorization <message>  
'Reference UUID' processed Certificate Order Authorization <message>

- **UpdateRenewalCertificateJob**  
*Update of internal structures of renewed certificate*
- **NotifyRenewalP12CertificateJob**  
*Notify recipients of PKCS#12 download (renewed certificate or recovery of PKCS#12)*
- **NotifyRenewalHsmCertificateJob**  
*Notify recipients of HSM Partition with certificate and key alias (End user partition PIN is provided by PKI operator)*
- **SetP12PinJob**  
*Request a pin from an end user for securing the PKCS#12 file*  
'Reference UUID' error handling Set P12 Pin Job <message>  
'Reference UUID' handling Set P12 Pin <message>  
'Reference UUID' aborted Set P12 Pin <message>  
'Reference UUID' processed Set P12 Pin <message>
- **RevokeCertificateJob**  
*Effective certificate revocation task*  
'Reference UUID' error handling Revoke Certificate Job <message>  
'Reference UUID' handling Revoke Certificate <message>  
'Reference UUID' aborted Revoke Certificate <message>  
'Reference UUID' processed Revoke Certificate <message>
- **MicrosoftCesKETJob**  
*Microsoft request for key encryption transport (escrow)*  
'Reference UUID' error handling Microsoft CES Job <message>  
'Reference UUID' handling Microsoft CES <message>  
'Reference UUID' aborted Microsoft CES <message>  
'Reference UUID' processed Microsoft CES <message>
- **MicrosoftCesQueryStatusJob**  
*Microsoft request for certificate issuance status*  
'Reference UUID' error handling Microsoft CES Job <message>  
'Reference UUID' handling Microsoft CES <message>  
'Reference UUID' aborted Microsoft CES <message>  
'Reference UUID' processed Microsoft CES <message>
- **MicrosoftCesRequestJob**  
*Microsoft request for certificate issuance*  
'Reference UUID' error handling Microsoft CES Job <message>  
'Reference UUID' handling Microsoft CES <message>  
'Reference UUID' aborted Microsoft CES <message>  
'Reference UUID' processed Microsoft CES <message>
- **MicrosoftCesStatusJob**  
*Microsoft request for order status*  
'Reference UUID' error handling Microsoft CES Job <message>  
'Reference UUID' handling Microsoft CES <message>  
'Reference UUID' aborted Microsoft CES <message>

'Reference UUID' processed Microsoft CES <message>

- **MicrosoftCesUnknownJob**

*Unknown Microsoft request type*

'Reference UUID' error handling Microsoft CES Job <message>

'Reference UUID' handling Microsoft CES <message>

'Reference UUID' aborted Microsoft CES <message>

'Reference UUID' processed Microsoft CES <message>

## 12.3 PKI

You access PKI entities and certificate policy templates via the PKI main menu. This is where you manage and create the PKI within your Realm.

1. PKI Entities

PKI entities are all PKI elements composing your PKI environment

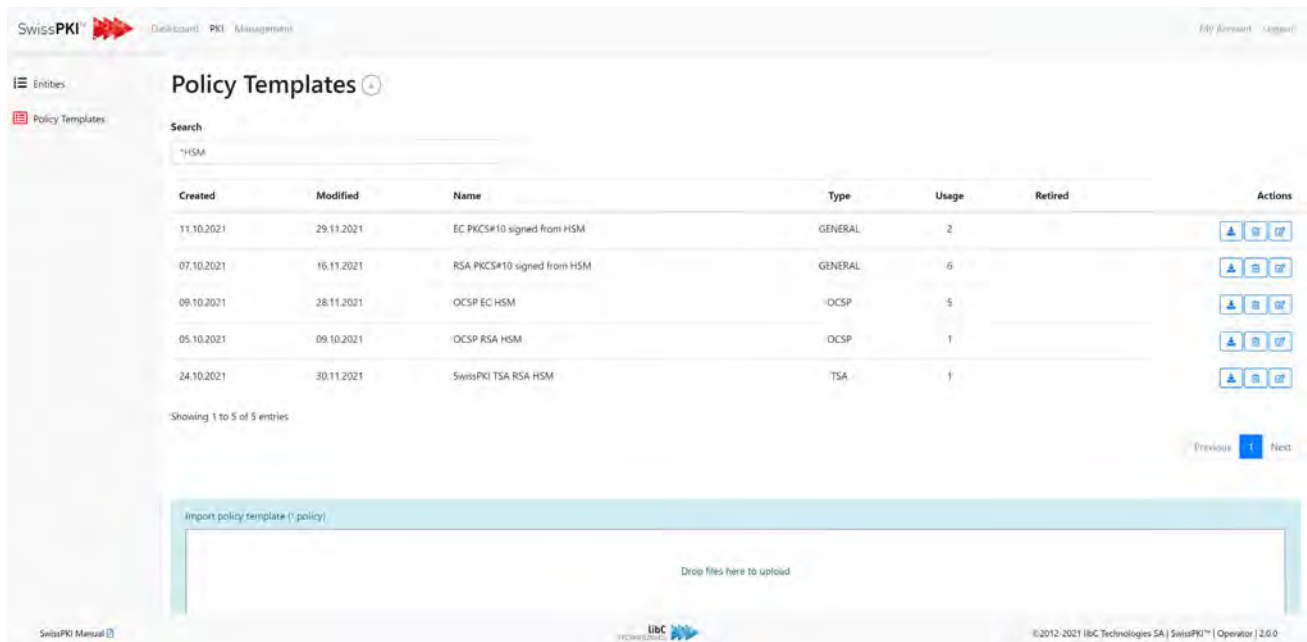
2. Certificate Policy Templates

Certificate Policy Templates are the definition of the certificate formats used by the PKI entities.

### 12.3.1 Certificate Policy Templates

Certificate Policy Templates define the certificates' static content. When associated to an Issuing CA, Certificate Policy Templates become Policy Instances.

The Certificate Policy Editor is a template editor which lets you edit and manage the certificate policy templates. SwissPKI distinguishes different template types and usages.



Created	Modified	Name	Type	Usage	Retired	Actions
11.10.2021	29.11.2021	EC PKCS#10 signed from HSM	GENERAL	2		[Icons]
07.10.2021	16.11.2021	RSA PKCS#10 signed from HSM	GENERAL	6		[Icons]
09.10.2021	28.11.2021	OCSP EC HSM	OCSP	5		[Icons]
05.10.2021	09.10.2021	OCSP RSA HSM	OCSP	1		[Icons]
24.10.2021	30.11.2021	SwissPKI TSA RSA HSM	TSA	1		[Icons]

Showing 1 to 5 of 5 entries

Import policy template (1 policy)

Drop files here to upload

Certificate Policy Templates are separated into two usages:

- **PKI only**  
Certificate policy templates marked as 'PKI only' in the table below are predefined templates used with the SwissPKI PKI entities.
- **End User**  
Certificate policy templates marked as 'End User' are templates which are used by the end users or systems to issue certificates.

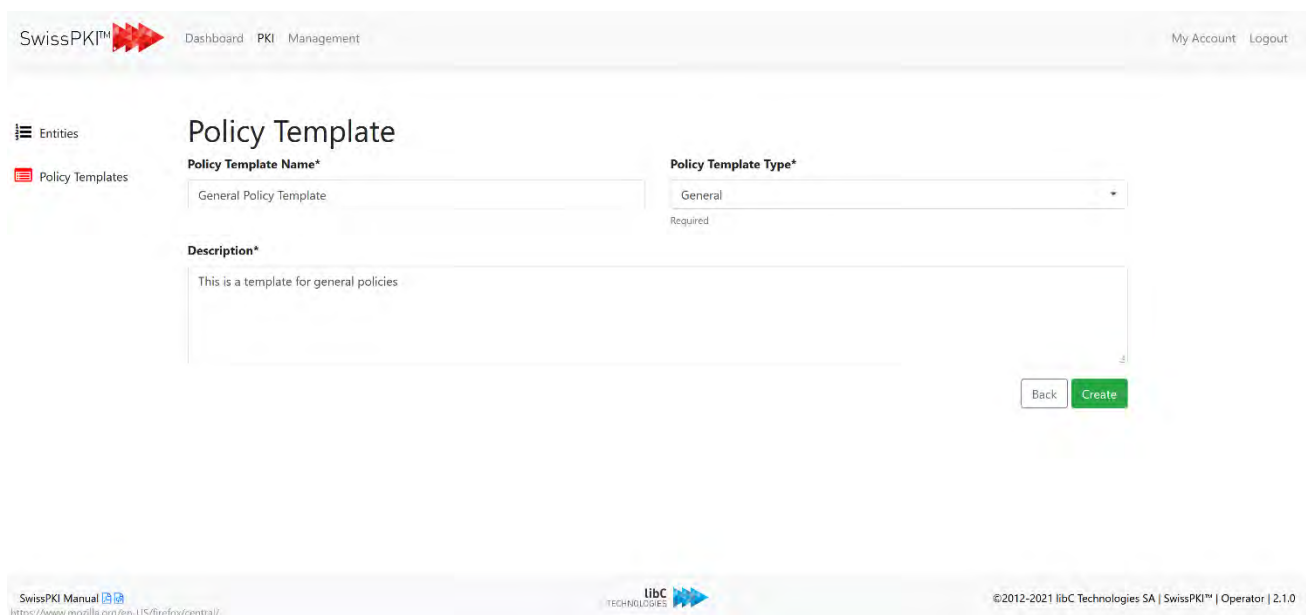


Types	Description	Usage
<b>Certificate Management Protocol (cipher)</b>	Used by the CMP service to issue cipher certificates and key pairs. The CMP service receives ciphered client requests with the encryption key wrapped with a certificate issued using this template	PKI only
<b>Certificate Management Protocol (signer)</b>	Used by the CMP service to issue signing certificates and key pairs. The CMP service sends client replies signed with a certificate issued using this template	PKI only
<b>Certification Authority</b>	Used to issue Root and Issuing Certification Authorities	PKI only
<b>Document Signer</b>	Used by the DSS service to issue signing certificates and key pairs. The DSS service signs requests using this template	PKI only
<b>External</b>	Used in conjunction with Certification Authorities of type EXTERNAL.	PKI only
<b>General</b>	Used to create your custom templates. This is the template to use for your end users and systems unless you plan to issue certificates for Microsoft using the SwissPKI autoenrollment service.	End User
<b>Microsoft</b>	Identical to the 'General' template but used in conjunction with the SwissPKI MSCA service.	End User
<b>Microsoft Public Trust (SwissSign)</b>	Used to issue Public Trust certificates for the SwissPKI MSCA service. Requires a Certification Authority of type 'SWISSIGN'	End User
<b>Online Certificate Status Protocol</b>	Used to issue OCSP server certificate and associated key pairs.	PKI only
<b>SCION Adapter</b>	Used to issue SCION certificates in conjunction with the SCION Protocol adapter.	PKI only
<b>SwissSign Public Trust</b>	Used to issue Public Trust certificates. Requires a Certification Authority of type 'SWISSIGN'	End User
<b>Time Stamp Authority</b>	Used to issue Time Stamp Authority certificate and associated key pairs.	PKI only

### 12.3.1.1 Create Policy Template

Creating a new policy template is done by clicking on the add button located on the right of the page title. After clicking, you are redirected to a form. There, you need to provide the following information:

Fields	Description
<b>Policy Template Name</b>	The policy template's name
<b>Policy Template Type</b>	The policy template's type. A list of all type is available on the previous page of this documentation
<b>Description</b>	The policy template's description




The screenshot shows the SwissPKI web interface. At the top, there is a navigation bar with 'SwissPKI™' and 'Dashboard PKI Management' on the left, and 'My Account Logout' on the right. Below the navigation bar, there is a sidebar with 'Entities' and 'Policy Templates'. The main content area is titled 'Policy Template' and contains a form with the following fields:

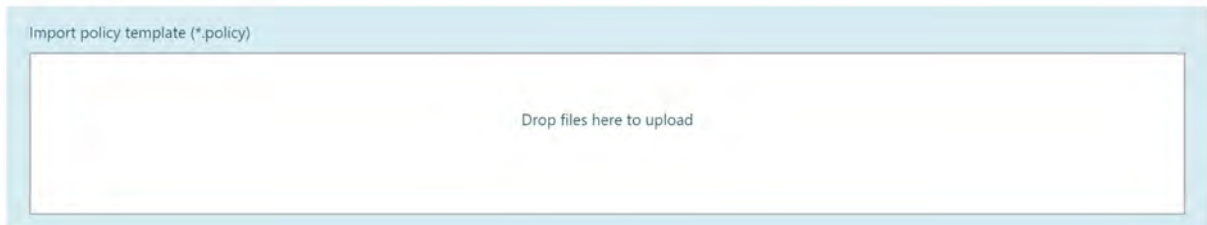
- Policy Template Name\***: A text input field containing 'General Policy Template'.
- Policy Template Type\***: A dropdown menu with 'General' selected. Below the dropdown, it says 'Required'.
- Description\***: A text area containing 'This is a template for general policies'.

At the bottom right of the form, there are two buttons: 'Back' and 'Create'.

At the bottom of the page, there is a footer with the following information:

- SwissPKI Manual <https://www.mozilla.org/en-US/firefox/central/>
- libC TECHNOLOGIES 
- ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.1.0







Optionally, you can import templates by drag and dropping the file into the upload area.



## 12.3.1.2 Certificate Policy Template Widgets

### 12.3.1.2.1 Widget settings

When configuring templates, you define each widget to be visible, mandatory and/or editable. The widget will behave based on the on/off state of each setting when used by a CA Operator or RA Officer.

	Fields	Description
	Mandatory	The mandatory setting forces users to provide a value when filling the policy.
	Visible	The visible setting displays or hides the field to the users when filling the policy.
	Editable	The editable setting allows the end user to provide a value when filling in the policy.
	Overwrite	The overwrite setting is only available to Subject DN attributes and allows to set fields as non-editable and non-visible. This option is used in conjunction with the SDN Overwrite Validator which can be set at policy instance and client policy mapping level to overwrite and prefill specific Subject DN attribute values
	www prefix	When SAN extension is DNS, then display check box at RA UI with option to add www prefix to DNS names
	Base domain	When SAN extension is DNS and wildcard DNS is enabled, then display check box at RA UI with option to add base domain to wildcard DNS.

### 12.3.1.2.2 General Information

Each certificate policy template has an information section:

^ General Information

<b>Policy type</b>	<b>Name</b>
GENERAL	RSA PKCS#10 signed from HSM
<b>Description</b>	
RSA PKCS#10 signed from HSM	

Fields	Description
<b>Policy Type</b>	The certificate policy template's type
<b>Name</b>	The certificate policy template's name
<b>Description</b>	The certificate policy template's description

### 12.3.1.2.3 Key Generation Parameters

Key generation parameters define the requirements of the keys used or produce for certificates. These settings cannot be modified by the end user.

^ Key Generation parameters

Key Gen: HSM | Key Type: RSA 2048 | Signature Algorithm: sha256 | Key Form: PKCS | HSM Partition: Nothing selected |  Exportable

Use existing key alias

---

^ Key Generation parameters

Key Gen: PKCS10 | Key Type: RSA 2048 | Signature Algorithm: sha256

---

^ Key Generation parameters

Key Gen: PKCS12 | Key Type: RSA 2048 | Signature Algorithm: sha256

---

^ Key Generation parameters

Key Gen: PKCS12 (with PIN) | Key Type: RSA 2048 | Signature Algorithm: sha256

Fields	Description
Key Gen	<p>The key generation forms:</p> <ul style="list-style-type: none"> <li>• PKCS10 User must provide a PKCS#10 request</li> <li>• PKCS12 The CA generates a software key pair, and the private key is escrowed</li> <li>• PKCS12 with PIN The CA generates a software key pair and protects it with the end user provided PIN. The key pair is not escrowed.</li> <li>• PKCS12 with CA PIN The CA generates a software key pair and protects it with a CA generated PIN. The PIN is sent via email self-service to the recipient. The key pair is not escrowed.</li> </ul>

	<ul style="list-style-type: none"> <li>• PKCS10 or PKCS12 (with PIN) When issuing a certificate, the RAO is required to select either PKCS10 or PKCS12 (with PIN)</li> <li>• PKCS10 or PKCS12 (with CA PIN) When issuing a certificate, the RAO is required to select either PKCS10 or PKCS12 (with CA PIN)</li> <li>• HSM The key pair is generated on an HSM partition</li> </ul>																				
<b>Key Type</b>	<p>The minimum key pair sizes</p> <table> <tr> <td>EC brainpool224r1</td> <td>EC secp256r1</td> </tr> <tr> <td>EC brainpool256r1</td> <td>EC secp384r1</td> </tr> <tr> <td>EC brainpool320r1</td> <td>EC secp521r1</td> </tr> <tr> <td>EC brainpool384r1</td> <td>EC x962 p239v1</td> </tr> <tr> <td>EC brainpool512r1</td> <td>EC x962 p239v2</td> </tr> <tr> <td>EC frp256v1</td> <td>EC x962 p239v3</td> </tr> <tr> <td>EC secp224k1</td> <td>RSA 2048</td> </tr> <tr> <td>EC secp224r1</td> <td>RSA 3072</td> </tr> <tr> <td>EC secp256k1</td> <td>RSA 4092</td> </tr> <tr> <td></td> <td>RSA 8192</td> </tr> </table>	EC brainpool224r1	EC secp256r1	EC brainpool256r1	EC secp384r1	EC brainpool320r1	EC secp521r1	EC brainpool384r1	EC x962 p239v1	EC brainpool512r1	EC x962 p239v2	EC frp256v1	EC x962 p239v3	EC secp224k1	RSA 2048	EC secp224r1	RSA 3072	EC secp256k1	RSA 4092		RSA 8192
EC brainpool224r1	EC secp256r1																				
EC brainpool256r1	EC secp384r1																				
EC brainpool320r1	EC secp521r1																				
EC brainpool384r1	EC x962 p239v1																				
EC brainpool512r1	EC x962 p239v2																				
EC frp256v1	EC x962 p239v3																				
EC secp224k1	RSA 2048																				
EC secp224r1	RSA 3072																				
EC secp256k1	RSA 4092																				
	RSA 8192																				
<b>Signature Algorithm</b>	<p>The signature algorithm used when issuing the certificate</p> <table> <tr> <td>sha224</td> <td>sha224/PSS/MfG1</td> <td>sha3-224</td> <td>sha3-224/PSS/MfG1</td> </tr> <tr> <td>sha256</td> <td>sha256/PSS/MfG1</td> <td>sha3-256</td> <td>sha3-256/PSS/MfG1</td> </tr> <tr> <td>sha384</td> <td>sha2384/PSS/MfG1</td> <td>sha3-384</td> <td>sha3-384/PSS/MfG1</td> </tr> <tr> <td>sha512</td> <td>sha2512/PSS/MfG1</td> <td>sha3-512</td> <td>sha3-512/PSS/MfG1</td> </tr> </table>	sha224	sha224/PSS/MfG1	sha3-224	sha3-224/PSS/MfG1	sha256	sha256/PSS/MfG1	sha3-256	sha3-256/PSS/MfG1	sha384	sha2384/PSS/MfG1	sha3-384	sha3-384/PSS/MfG1	sha512	sha2512/PSS/MfG1	sha3-512	sha3-512/PSS/MfG1				
sha224	sha224/PSS/MfG1	sha3-224	sha3-224/PSS/MfG1																		
sha256	sha256/PSS/MfG1	sha3-256	sha3-256/PSS/MfG1																		
sha384	sha2384/PSS/MfG1	sha3-384	sha3-384/PSS/MfG1																		
sha512	sha2512/PSS/MfG1	sha3-512	sha3-512/PSS/MfG1																		
<b>Key Form</b>	<p>PKCS Used for issuing certificates (even for Microsoft)</p> <p>Microsoft Used for hardware key injection on Smart Cards used with Microsoft. Microsoft private key blob have non-standard PKCS encoding</p>																				

















<b>HSM Partition</b>	Drop down with a list of available HSM partitions
<b>Exportable</b>	Generates an exportable private key when generated on the HSM (only supported for Primus HSM)
<b>Use existing key alias</b>	<p>For HSM key generation type only, a reference to a pre generated key is available. The value of the field is the key alias (CKA_LABEL) of the pre generated key pair on the HSM partition. When generating a key pair and the external key reference is available, then the key is resolved in place of a key generation.</p> <p>Note that the pre generated key pair must be a key pair using both the same alias for CKO_PUBLIC and CKO_PRIVATE attributes.</p>



### 12.3.1.2.4 Subject Distinguished Name

Certificate subject distinguished name with field encoding. Top level element in list matches most left attribute/value pair in encoded subject distinguished name.

#### ^ Subject Distinguished Name

	General Name	Encoding	Value	
☰	Country	Printable String	CH	   
☰	Organization	Printable String	libC	   
☰	Organizational Unit	Printable String	SwissPKI	   
☰	Common Name	UTF8 String	SwissPKI Staging Root CA RSA 4096	   
+ Add DN item				

Fields	Description
<b>General Name</b>	<p>The SDN's general name. The following choices are available:</p> <ul style="list-style-type: none"> <li>• Business Category</li> <li>• Common Name</li> <li>• Country</li> <li>• DN Qualifier</li> <li>• Domain Controller</li> <li>• Email</li> <li>• Given name</li> <li>• Initials</li> <li>• Jurisdiction of Incorporation Country</li> <li>• Jurisdiction of Incorporation Locality</li> <li>• Jurisdiction of Incorporation State</li> <li>• Locality</li> <li>• Name</li> <li>• Organization</li> <li>• Organization Id</li> <li>• Organizational unit</li> <li>• Postal code</li> <li>• SCION ISD-AS Number</li> <li>• Serial number</li> </ul>

	<ul style="list-style-type: none"> <li>• State</li> <li>• Street Address</li> <li>• Surname</li> <li>• TPM Manufacturer</li> <li>• TPM Model</li> <li>• TPM Version</li> <li>• Title</li> <li>• UID</li> <li>• Unique Identifier</li> </ul>
<b>Encoding</b>	<p>The encoding format of each field. The following formats are available:</p> <ul style="list-style-type: none"> <li>• UTF8 String</li> <li>• Printable String</li> <li>• IA5 String</li> <li>• T.61 String</li> <li>• BMP String</li> <li>• Universal String</li> </ul>
<b>Value</b>	The field's value

### 12.3.1.2.5 Certificate Validity

Define the lifespan of your certificate.

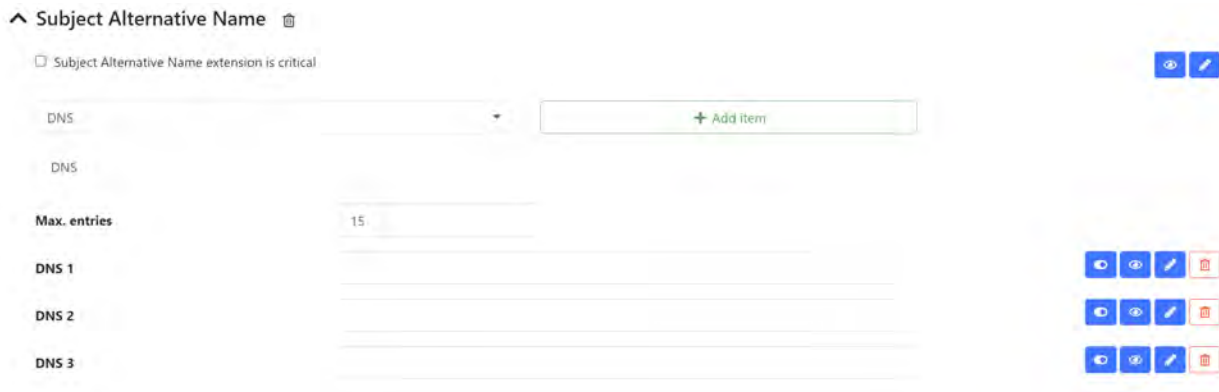
If the end validity of the issued certificate exceeds the end validity of the Issuing CA, then the end validity of the issued certificate is set to the end validity of the Issuing CA.



Fields	Description
<b>Validity</b>	Allows you to define the unit in which the certificate validity is defined. The following options are available: <ul style="list-style-type: none"> <li>• Years</li> <li>• Years (Options 1 ,2, or 3 years)</li> <li>• Months</li> <li>• Days</li> <li>• Time</li> </ul>
<b>Duration</b>	The actual duration of the certificate’s validity.

### 12.3.1.2.6 Subject Alternative Name

The subject alternative name extension



These identities can be given in the following formats:

Fields	Description
<b>DNS</b>	DNS or DNS with wild card
<b>Email</b>	RFC822 email
<b>IP</b>	IPv4 or IPv6
<b>UPN</b>	Microsoft UPN
<b>GUID</b>	Microsoft GUID
<b>Registered Id</b>	Object Identifier
<b>SVID</b>	Servitel ID
<b>Directory Name</b>	See Subject Distinguished Name
<b>URI</b>	IP Address or Hostname

The **Max. entries** field lets you set the maximum number of SAN elements. If the value is 0 or equal to the number of element types, then the maximum number of entries is **Max. entries**. If **Max entries** is larger than the number of entries, then the RA Operator or API can add as many entries of the type to the SAN as defined in the field.

### 12.3.1.2.7 Authority Information Access

OCSP and CA Issuer certificate extension

^ Authority Information Access 

Authority Information Access is critical

**CA Issuer**

URI

**OCSP**

URI



Fields	Description
<b>Type</b>	The authority information access type: <ul style="list-style-type: none"> <li>• CA Issuer</li> <li>• OCSP</li> </ul>
<b>Format</b>	The authority information access method (choice of names)
<b>Value</b>	The authority information access value.

### 12.3.1.2.8 Key usage

The key usage extension as defined in RFC 5280

^ Key Usage 

Key Usage extension is critical

Cert Sign

Content Commitment

CRL Sign

Data Encipherment

Decipher Only

Digital Signature

Encipher Only

Key Agreement

Key Encipherment




### 12.3.1.2.9 Extended Key Usage

This extension indicates one or more purposes for which the key and certificate may be used

#### Extended Key Usage

+ Custom Extended Key Usage

<input type="checkbox"/> Extended Key Usage extension is critical	 
<input checked="" type="checkbox"/> Server Authentication (1.3.6.1.5.5.7.3.1)	 
<input type="checkbox"/> Client Authentication (1.3.6.1.5.5.7.3.2)	 
<input type="checkbox"/> Code Signing (1.3.6.1.5.5.7.3.3)	 
<input type="checkbox"/> Email Protection (1.3.6.1.5.5.7.3.4)	 
<input type="checkbox"/> IPSec Tunnel (1.3.6.1.5.5.7.3.6)	 
<input type="checkbox"/> IPSec System (1.3.6.1.5.5.7.3.5)	 
<input type="checkbox"/> IPSec User (1.3.6.1.5.5.7.3.7)	 
<input type="checkbox"/> Time Stamping (1.3.6.1.5.5.7.3.8)	 
<input type="checkbox"/> OCSP Signing (1.3.6.1.5.5.7.3.9)	 
<input type="checkbox"/> OCSP Signing, OCSP No Check (1.3.6.1.5.5.7.48.1.5)	 
<input type="checkbox"/> SCION KP Root (1.3.6.1.4.1.55324.1.3.3)	 

You add custom extended key usages by clicking on the button at the top of the list. This will open a pop up where you provide an OID and name for it.

Add custom Extended Key Usage
×

**OID**

**Extension Name**

### 12.3.1.2.10 Authority Key Information

The authority key identifier extension provides a means of identifying the public key corresponding to the private key used to sign a certificate. This extension is used where an issuer has multiple signing keys (either due to multiple concurrent key pairs or due to changeover). The identification may be based on either the key identifier (the subject key identifier in the issuer's certificate) or the issuer name and serial number.

#### ^ Authority Key Identifier

- Authority Key Identifier is critical
- Include Authority Certificate Issuer
- Include Authority Certificate Serial Number
- Include Authority Key Identifier



### 12.3.1.2.11 Subject Key Information

The subject key information extension provides a means of identifying certificates that contain a particular public key. The keyIdentifier is composed of the 160-bit SHA-1 hash of the value of the BIT STRING subjectPublicKey (excluding the tag, length, and number of unused bits).

#### ^ Subject Key Identifier

- Subject Key Identifier extension is critical



### 12.3.1.2.12 Basic Constraint

The basic constraints extension identifies whether the subject of the certificate is a CA and the maximum depth of valid certification paths that include this certificate.



Fields	Description
Is CA	If selected, indicates that the certificate is of type certification authority
Path Length Constraint	Maximum depth of valid certification path including the certificate For indefinite length, use -1

### 12.3.1.2.13 Domain Controller

The domain controller sets the Microsoft Domain Controller Extension.



### 12.3.1.2.14 OCSP must staple

X.509v3 Transport Layer Security (TLS) extension OID 1.3.6.1.5.5.7.1.24





### 12.3.1.2.15 OCSP No Check

Sets the OCSP No Check Extension

#### ^ OCSP No Check

OCSP No Check is critical



### 12.3.1.2.16 CRL Distribution Point

The CRL distribution points extension identifies how CRL information is obtained. The CRL are mapped at the Policy Instance/CA level using a CA CDP (see section *12.3.1.1.1.2.4 Policy Instance CDP Mappings*).

#### ^ CRL Distribution Point

CRL Distribution Point is critical



### 12.3.1.2.17 Private Key Usage Period

Private key usage period extension for allowing the certificate issuer to specify a different validity period for the private key than the certificate. This extension is intended for use with digital signature keys.

#### ^ Private key usage period

- Private key usage period is critical
- Private key usage period is optional

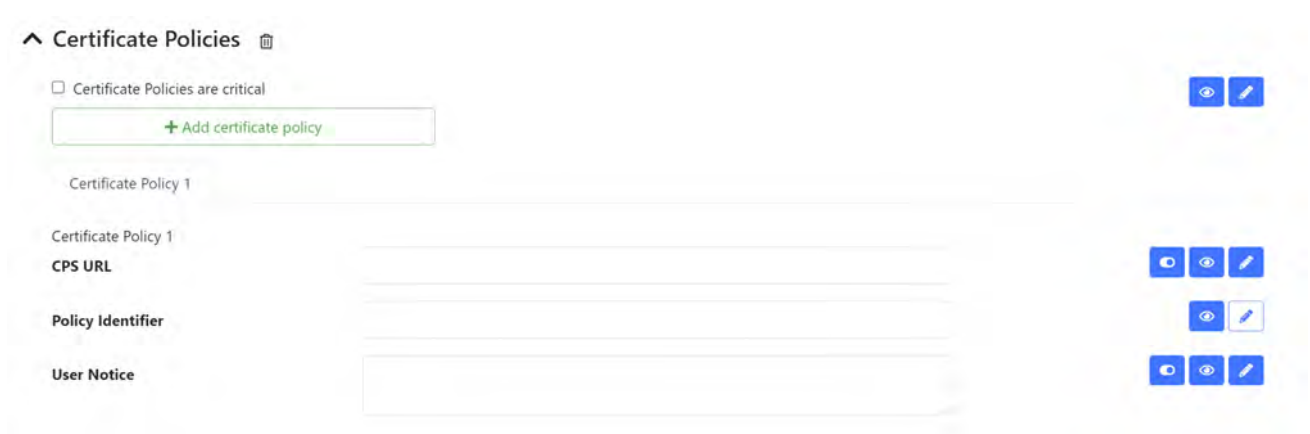
Not before

Not after



### 12.3.1.2.18 Certificate Policies


The certificate policies extension contains a sequence of one or more policy information terms, each of which consists of an object identifier (OID) and optional qualifiers. The following fields are available when configuring your certificate policies:





Fields	Description
<b>CPS URL</b>	The certificate policy's URL
<b>Policy Identifier</b>	The certificate policy's identifier. This field is mandatory
<b>User Notice</b>	A custom message that is sent to the user.

### 12.3.1.2.19 Name Constraints

The name constraints extension, which must be used only in a CA certificate, indicates a name space within which all subject names in subsequent certificates in a certification path must be located. Restrictions apply to the subject distinguished name and apply to subject alternative names. Restrictions apply only when the specified name form is present. If no name of the type is in the certificate, the certificate is acceptable.

^ Name Constraints 

Name constraints extension is critical  

Permitted sub trees

Min.	Max.	Base
0	<input type="checkbox"/>	Directory Name




General Name	Encoding	Value
No selection	Nothing selected	





[+ Add DN item](#)

[+ Permitted sub trees](#)

Excluded sub trees

[+ Excluded sub trees](#)

### 12.3.1.2.20 Qualified Statement

Qualified Statement v2 is an extension for certificates qualified by the ETSI TS 101 862 norm.

^ Qualified Statement v2 

Qualified Statement is critical  

ETSI QC Compliant  

SSCD Secure Signature Creation Device  

Certificate for electronic signatures as defined in Regulation (EU) No 910/2014  

Certificate for electronic seals as defined in Regulation (EU) No 910/2014  

Certificate for website authentication defined in Regulation (EU) No 910/2014  

Include ETSI Retention Period attribute

Include ETSI Limit Value

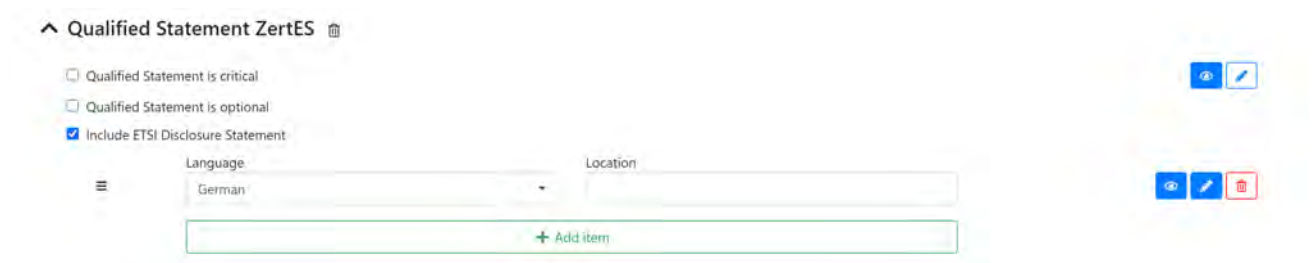
Include ETSI Disclosure Statement

Include ETSI Qualified Certificate Country Legislation

SwissPKI also disposes of two pre-configured Qualified Statements for eIDAS and ZerteS.

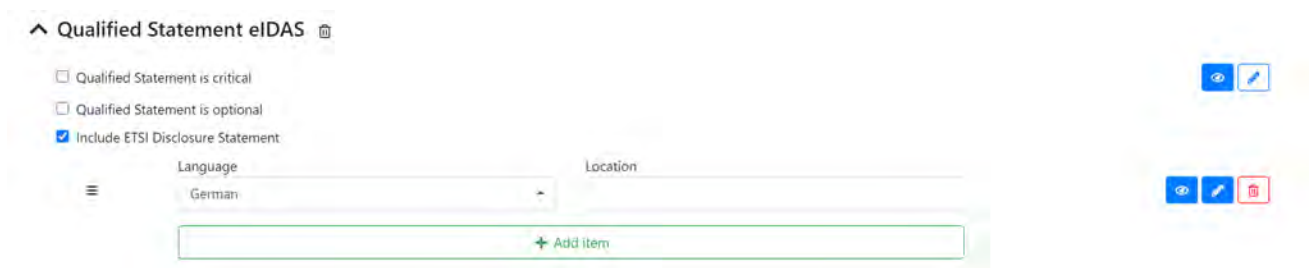
### 12.3.1.2.21 ZertES Qualified Statement

Produces a ZertES conform certificate extension



### 12.3.1.2.22 eIDAS Qualified Statement

Produces an eIDAS conform certificate extension




### 12.3.1.2.23 ETSI Short Term Qualified Statement

Produces an ETSI Short Term validation extension. This extension is used in conjunction with short term validity certificates.



### 12.3.1.2.24 Corda Role Extension

Corda X.509v3 extension as specified in <https://trust.corda.network/trust-root/certificate-policy.html> :



### 12.3.1.2.25 Microsoft Application Policies

The Microsoft application policies extension can be used by an application to filter certificates based on permitted use. Permitted uses are identified by OIDs. This extension is like the extended key usage extension but with stricter semantics applied to the parent CA. The extension is Microsoft specific <https://docs.microsoft.com/en-us/windows/win32/api/certenroll/nn-certenroll-ix509extensionmsapplicationpolicies>.

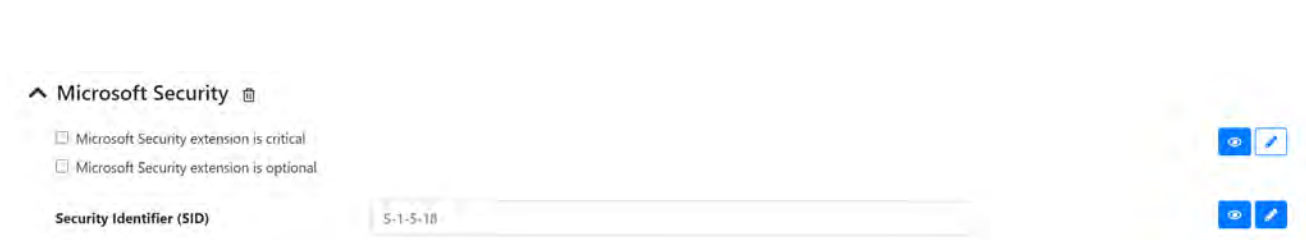
### 12.3.1.2.26 Microsoft Certificate Template

The Microsoft Certificate Template extension allows the setting of a Template OID, as well as a Major and Minor Version number.



### 12.3.1.2.27 Microsoft SID

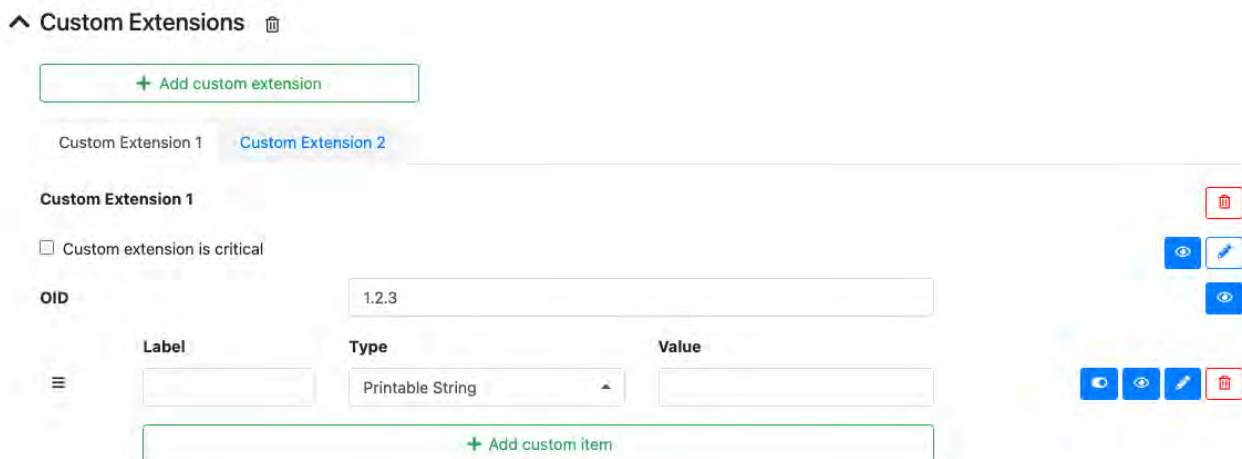
The Microsoft Security Identifier<sup>18</sup>.



<sup>18</sup> <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understand-security-identifiers>

### 12.3.1.2.28 Custom Extensions

Custom extensions can be added by specifying an Object Identifier (OID) for the section and adding items to the section. Every item consists of a label (which helps to identify the field), a type and a value.



### 12.3.1.2.29 CAA Rule

Perform a CAA check when issuing a certificate for this template using a CAA Rule as defined in section 12.2.4.4 CAA Rules



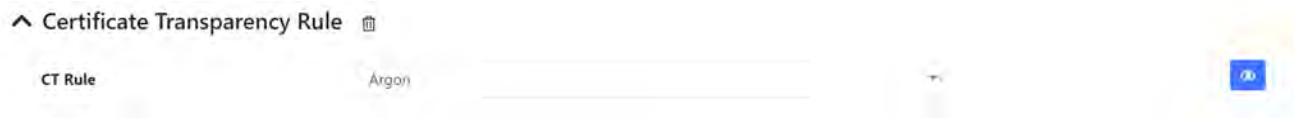
### 12.3.1.2.30 DNS Owner Rule

Perform a DNS Owner check when issuing a certificate for this template using a DNS Owner Check Rule as defined in section 12.2.4.5 DNS Owner Check Rules



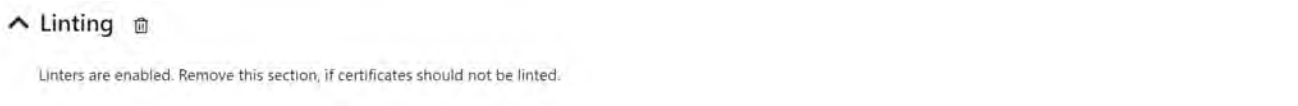
### 12.3.1.2.31 Certificate Transparency Rule

Produce a CT log entry when issuing a certificate for this template using a CT Rule as defined in section [12.2.4.6 CT Rules](#)



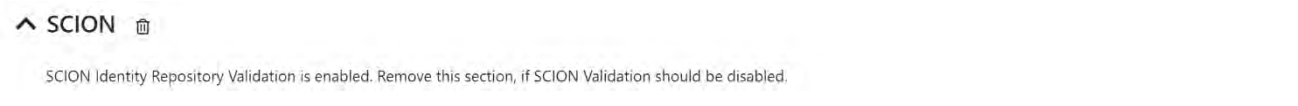
### 12.3.1.2.32 Linting

Perform certificate linting when issuing a certificate for this template using the defined Linters as per [11.5.8 Realm Linters](#)



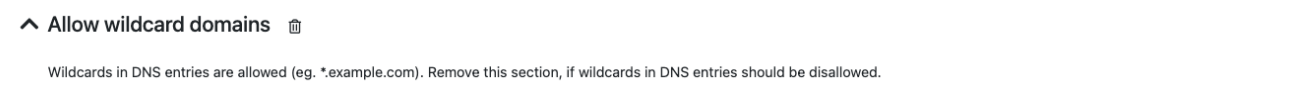
### 12.3.1.2.33 SCION

Indicates that the issued certificate using this template is a SCION device certificate. This attribute triggers a validation to the SCION Identity Repository. SCION Repository is defined in section [11.5.10 Realm SCION](#).



### 12.3.1.2.34 Wildcards

Indicates if wildcards (\*, ?) can be used as part of domain names in DNS entries. When this extension is active, values like \*.example.com are possible.



### 12.3.1.2.35 Microsoft Policy

Applies only to certificate templates of type ‘Microsoft.’

For Microsoft autoenrollment, specific Microsoft policy attributes are required. Depending on your deployment strategy, you can let SwissPKI manage the Microsoft policy templates or use Microsoft AD policy templates by redirecting the CES protocol to SwissPKI.

- Policy Template is handled by Microsoft.

If ‘Policy Template is managed by Microsoft’ is set to false, then SwissPKI becomes the certificate enrollment policy server. If it is set to true, then Microsoft manages the CEP requests and redirects the CES requests to SwissPKI. For detailed configuration settings and deployment, please contact libC Technologies for consulting.

#### 12.3.1.2.35.1 Enrollment flags

CA instructions

##### ^ Microsoft Policy

- Policy Template is handled by Microsoft.

Enrollment Flags   Subject Name Flags   General Flags   Schema   Permission   Private Key Flags

- Instructs the client and CA to include an S/MIME extension
- Instructs the CA to append the issued certificate to the userCertificate attribute, on the user object in AD
- Instructs the CA to check the user’s userCertificate attribute in AD
- Instructs the client to sign the renewal request using the private key of the existing certificate.
- Instructs the client to get a user’s consent before attempting to enroll
- Instructs the client to delete any expired, revoked, or renewed certificate from the user’s certificate stores
- Instructs the client to reuse the private key for a smart card-based certificate renewal

#### 12.3.1.2.35.2 Subject Name Flags

CA Subject Name instructions

##### ^ Microsoft Policy

- Policy Template is handled by Microsoft.

Enrollment Flags   Subject Name Flags   General Flags   Schema   Permission   Private Key Flags

- The client supplies the Subject field value in the certificate request.
- The client supplies the Subject Alternative Name field value in the certificate request.
- The CA adds the value of the DNS of the root domain to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value of the userPrincipalName attribute from the requestor’s user object in AD to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value of the objectGUID attribute from the requestor’s user object in AD to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value of the userPrincipalName attribute from the requestor’s user object in AD to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value of the mail attribute from the requestor’s user object in AD to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value obtained from the dNSHostName attribute of the requestor’s user object in AD to the Subject Alternative Name extension of the issued certificate.
- The CA adds the value obtained from the dNSHostName attribute of the requestor’s user object in AD as the CN in the Subject extension of the issued certificate.
- The CA adds the value of the mail attribute from the requestor’s user object in AD as the Subject extension of the issued certificate.
- The CA sets the Subject Name to the cn attribute value of the requestor’s user object in AD.
- The CA sets the Subject Name to the distinguishedName attribute value of the requestor’s user object in AD.
- The client reuses the values of the Subject Name and Subject Alternative Name extensions from an existing, valid certificate when creating a renewal certificate request



### 12.3.1.2.35.3 General Flags

#### Microsoft certificate template type

Microsoft Policy

Policy Template is handled by Microsoft.

Enrollment Flags Subject Name Flags General Flags Schema Permission Private Key Flags

This certificate template is for an end entity that represents a machine.

A certificate request for a CA certificate.

A certificate request for cross-certifying a certificate.

### 12.3.1.2.35.4 Schema

#### Microsoft policy template schema version

Microsoft Policy

Policy Template is handled by Microsoft.

Enrollment Flags Subject Name Flags General Flags Schema Permission Private Key Flags

Microsoft Certificate Template OID: 2.16.756.3.2.1

Schema version: 3

The major version number: 1

The minor version number: 0

### 12.3.1.2.35.5 Permission

#### Microsoft enrollment permissions

Microsoft Policy

Policy Template is handled by Microsoft.

Enrollment Flags Subject Name Flags General Flags Schema Permission Private Key Flags

Enrollment enabled

Auto Enrollment enabled

### 12.3.1.2.35.6 Private Key Flags

#### Microsoft private key handling instructions to the client

Microsoft Policy

Policy Template is handled by Microsoft.

Enrollment Flags Subject Name Flags General Flags Schema Permission Private Key Flags

Instructs the client to archive the private key.

Instructs the client to allow the private key to be exported.

Instructs the client to protect the private key.

### 12.3.1.2.36 Swiss Sign Product Name

Applies only to certificate templates of type 'SwissSign' and 'Microsoft SwissSign.'

Identifies the Swiss Sign product name you want to issue. This field should contain the SwissSign product UUID. This list of all the available products with their corresponding UUIDs can be found in section 12.3.2.1.9 Products. For instance: [pma-56cf9392-4547-b56b-8580ec2f73a6](#)

#### ^ SwissSign Product Name

Provide the SwissSign product name you have registered.

My Swiss Sign Product Name





















## 12.3.2 Entities

Before creating PKI entities, you must define Certificate Policy Templates as described in section *12.3.1 Certificate Policy Templates*.

**Search PKI Entities**

Search

[Add CA](#)
[Add OCSP](#)
[Add TSA](#)
[Add DSS](#)
[Add CMP](#)
[Add CES/CEP](#)
[Add Publisher](#)

- >  SwissPKI Staging Root CA RSA 4096 (HSM) 
- >  SwissPKI Staging Root CA RSA 4096 (SW) 
- ∨  SwissPKI Staging Root CA EC 512 (HSM) 
- ∨  SwissPKI Staging Issuing CA EC 512 (HSM) 
  - ↶ SwissPKI Publisher 
  - 👁️ SwissPKI OCSP 
  - 🕒 SwissPKI TSA EC (HSM) 
  - ↶ SwissPKI Publisher 
- >  SwissPKI Staging Root CA EC 512 (SW) 
-  External CA 
-  Public Trust CA 
- 👁️ SwissPKI OCSP 
- 🕒 SwissPKI TSA RSA (HSM) 

Types	Description
CA	Certification Authority
OCSP	Online Certificate Server Protocol
TSA	Time Stamp Authority
DSS	Document Signer Service
CMP	Certificate Management Protocol
CES/CEP	Microsoft CES and CEP for autoenrollment
Publisher	Certificate and CRL/ARL publisher

### 12.3.2.1 Certification Authority

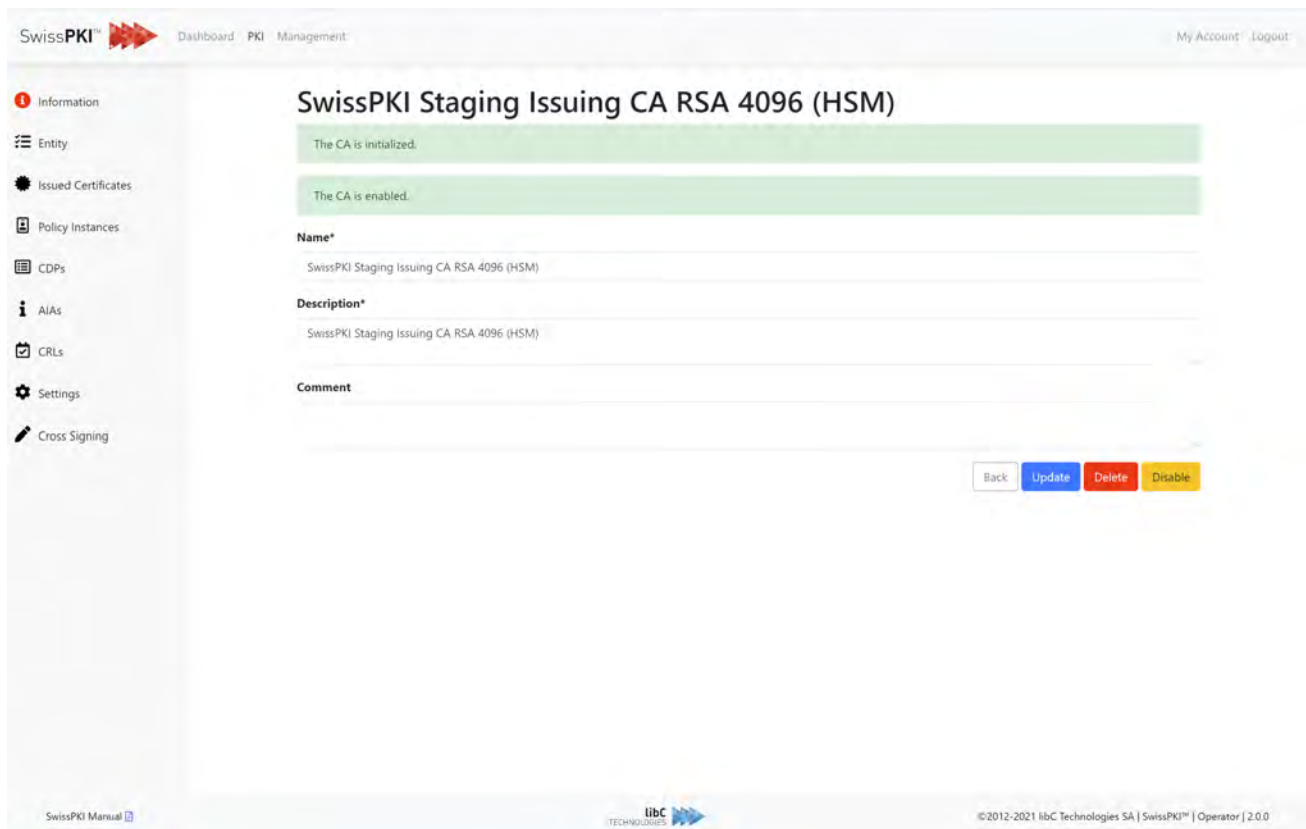
Certification authorities are divided in three different subcategories:

1. SwissPKI Certification Authorities  
CAs with generated Software and/or Hardware key pairs.
2. SwissPKI Air Gaped CA  
Sealed CA linked with an Offline CA
3. SwissSign Certification Authorities  
Integration of Public Trust certificates with SwissSign AG (requires a SwissSign CMC account)
4. External Certification Authorities  
Virtual CA to manage certificates imported from various CAs

### 12.3.2.1.1 Information

Applies to Certification Authorities of type **SwissPKI**, **External** and **SwissSign**.

Certification Authority information pane for its logical name and corresponding description.



The screenshot shows the 'SwissPKI Staging Issuing CA RSA 4096 (HSM)' information pane. It features a left sidebar with navigation options like Information, Entity, Issued Certificates, Policy Instances, CDPs, AIAs, CRLs, Settings, and Cross Signing. The main content area displays the CA name and description, with status messages indicating initialization and enabling. Action buttons for Back, Update, Delete, and Disable are visible at the bottom right.

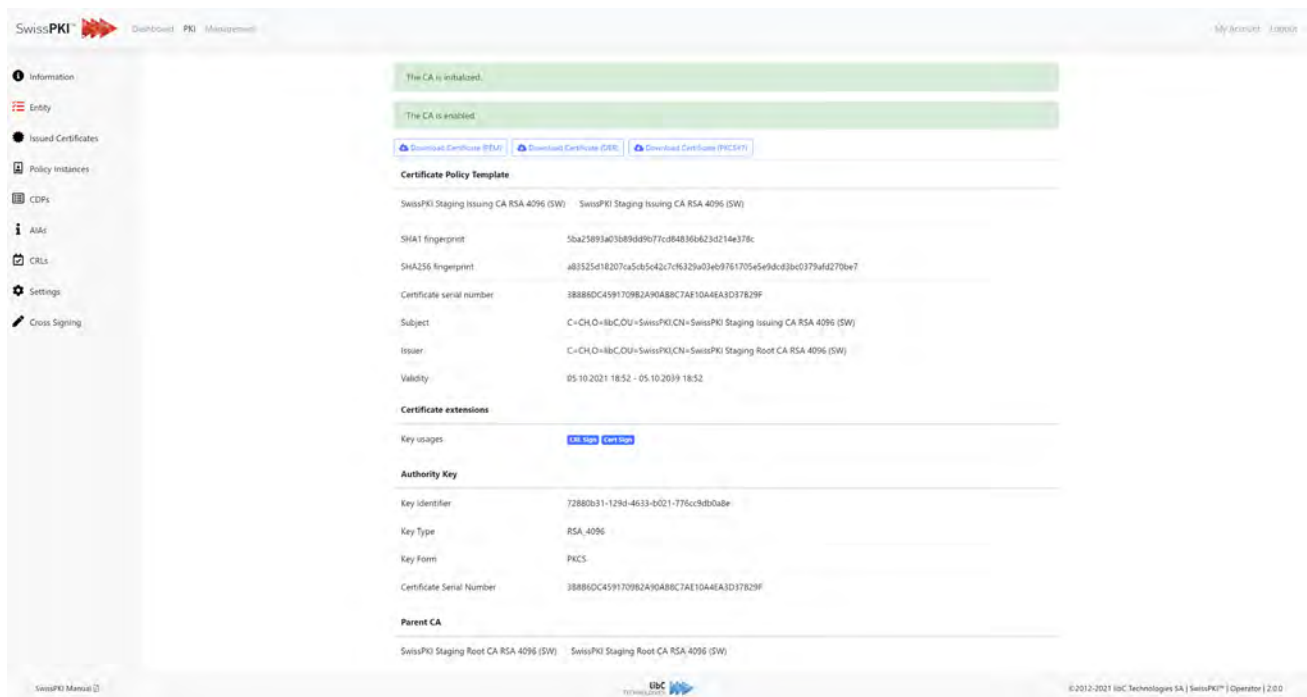
Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the CA service performs a soft delete and does not revoke the CA except the associated services such as TSA, OCSP, CMP and DSS. The instance is marked as deleted.
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests (certificates and CRLs)

### 12.3.2.1.2 Entity

Applies to Certification Authorities of type **SwissPKI**.

The Certification Authority's configuration information including:

1. Download of the CA certificate is PEM, DER, or PKCS#7. If the Certification Authority is cross signed, downloading the PKCS#7 certificate chain includes the path to Root certificate as selected in the '**Authority Key**' section.
2. Key usage extensions and Subject Alt Names
3. Authority key and reference to alias on HSM if the key pair is a hardware key. Optionally a list of additional Authority Keys if the Certification Authority is cross signed.
4. List of services linked to the Certification Authority



If your Certification Authority is cross signed and its configuration is set to use the cross signed path, then you have the option to switch between the different authorities using in the '**Authority Key**' section.

**Authority Key**

Key Identifier aed60c92-0176-4ed4-b26f-625a3678a391

Key Type RSA\_2048

Key Form PKCS

Certificate Serial Number 54B33C7EE2653D1474D03F2324483C8D5459A97D

Other CA | 54B33C7EE2653D1474D03F2324483C8D5459A97D | Digital Signature (cross signed)

C=CH,O=Foo,OU=Bar,CN=Other CA | 7F5A35F28DCB594C13C0DCD66384479945003283 | Digital Signature

Other CA | 54B33C7EE2653D1474D03F2324483C8D5459A97D | Digital Signature (cross signed)

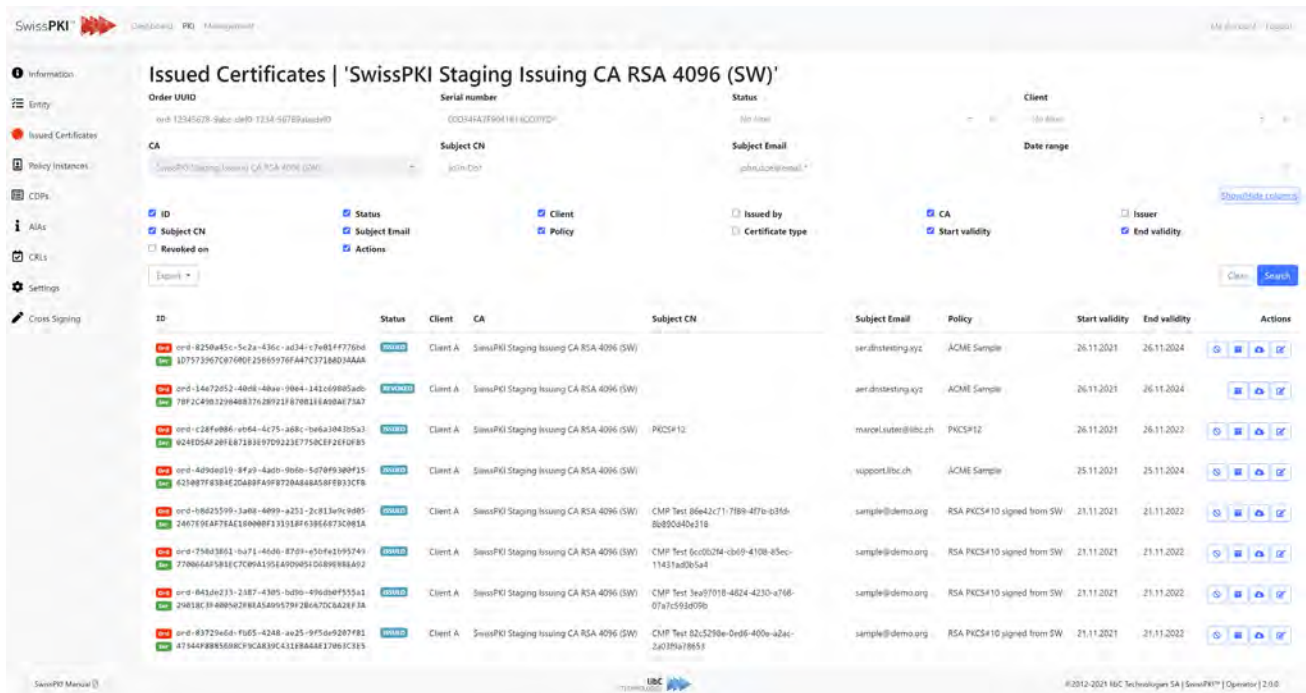
Update

### 12.3.2.1.3 Issued Certificates

Applies to Certification Authorities of type **SwissPKI**, **External** and **SwissSign**.

#### 12.3.2.1.3.1 Locally Issued Certificates

This page contains a list of all the locally issued certificate orders for the selected CA.



ID	Status	Client	CA	Subject CN	Subject Email	Policy	Start validity	End validity	Actions
ord-8250455c-5e2a-436c-ad34-7e031f7700d	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)		ser.dn@staging.xyz	ACME Sample	26.11.2021	26.11.2024	Show/Hide
ord-14e72052-4008-480a-9064-141c69905adb	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)		ser.dn@staging.xyz	ACME Sample	26.11.2021	26.11.2024	Show/Hide
ord-128f4e86-eb64-4c75-a08c-b0ca3a0305a2	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)	PKCS#12	marco.luten@libc.ch	PKCS#12	26.11.2021	26.11.2022	Show/Hide
ord-4090e010-8f49-44dd-90b6-5d70f9300f35	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)		support@libc.ch	ACME Sample	25.11.2021	25.11.2024	Show/Hide
ord-16025791-3a08-4099-a251-2c913a9c0405	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Test 86e42c71-7f89-47fb-63fb-8b050406216	sample@demo.org	RSA PKCS#10 signed from SW	21.11.2021	21.11.2022	Show/Hide
ord-750d3861-ba71-46d0-8701-e50f421992749	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Test 6c00274-cb69-4108-85ec-11431ad0e5a4	sample@demo.org	RSA PKCS#10 signed from SW	21.11.2021	21.11.2022	Show/Hide
ord-043ad0273-2187-4305-bd36-496dbef555a13	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Test 3ea7f018-4024-4230-a766-07a7c5938090	sample@demo.org	RSA PKCS#10 signed from SW	21.11.2021	21.11.2022	Show/Hide
ord-83721e6d-f065-4248-aa25-9f5da9207f81	Issued	Client A	SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Test 02c5270e-0e06-400e-a2ac-2a039a708551	sample@demo.org	RSA PKCS#10 signed from SW	21.11.2021	31.11.2022	Show/Hide

You can choose which column are displayed/hidden by clicking on the 'Show/Hide' link and selecting the desired search columns.

The action column allows you to:

- Revoke a certificate (if not already revoked). The 'revoke permission' must be enabled for your role to revoke a certificate
- Request the certificate's publication. This action is enabled if the Issuing CA is linked to a Publisher instance.
- Download the certificate in a PEM format to a local file
- Access the certificate details



The search filters at the top of the page allow you to narrow down the certificates in the list.

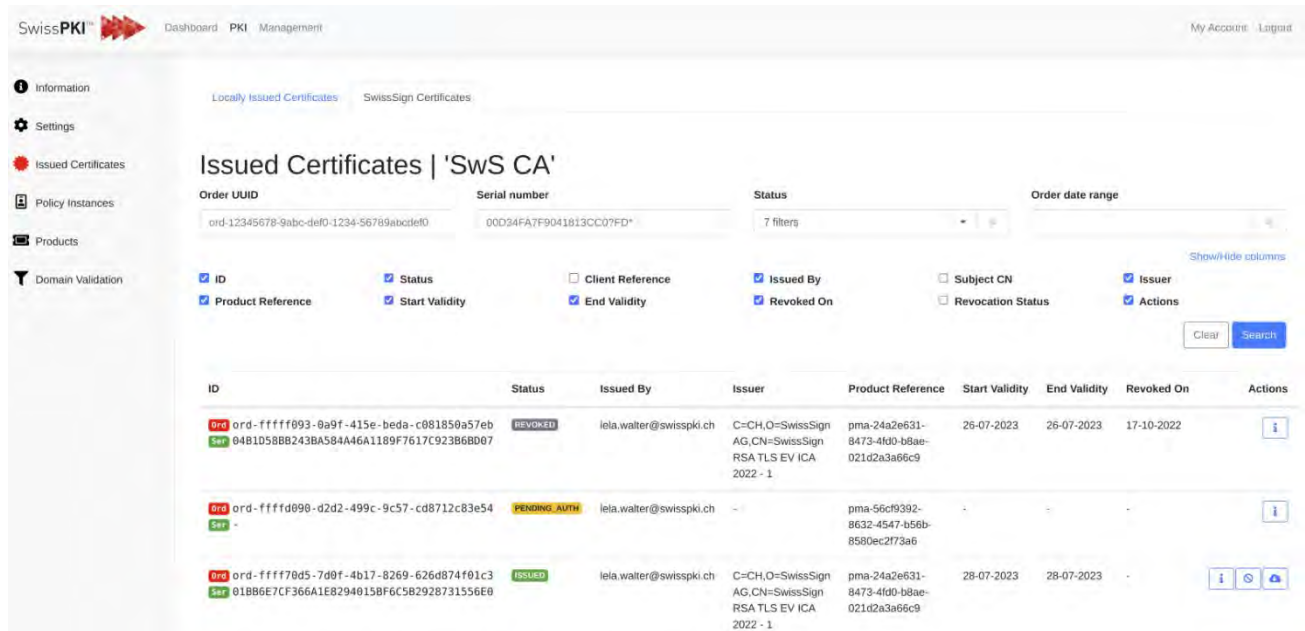
Filter	Description
<b>Order UUID</b>	Allows to filter certificates by order UUID
<b>Serial Number</b>	Allows to filter certificates by serial number
<b>Status</b>	Allows to filter certificates by their status: <ul style="list-style-type: none"> <li>- NEW</li> <li>- KEY_VALIDATION</li> <li>- PRE_VALIDATION</li> <li>- GENERATE_TBS</li> <li>- PENDING_AUTH</li> <li>- PRE_ISSUE</li> <li>- ISSUE</li> <li>- POST_VALIDATION</li> <li>- FINALIZE ISSUANCE</li> <li>- ISSUED</li> <li>- REVOKED</li> <li>- FAILED</li> <li>- REJECTED</li> <li>- PENDING CSR RENEWAL</li> <li>- UNKNOWN</li> </ul>
<b>Client</b>	Allows to filter certificates by clients. This filter contains a list of all existing clients
<b>Subject CN</b>	Allows to filter certificates by Subject CN
<b>Subject Email</b>	Allows to filter certificates by Subject Email
<b>Date Range</b>	Allows to filter certificate that were issued in the defined date range

### 12.3.2.1.3.2 SwissSign Certificates

Applies only to Certification Authorities of type 'SwissSign.'

This tab allows you to view and manage all the certificates from the SwissSign CA that were not issued locally. You have the possibility to search for certificates by Order UUID, Serial Number, Status or by selecting an Issued Date Range. It is also possible to choose which columns are displayed using the 'Show/Hide' button. Note that the querying of SwissSign certificates is done through an API call. Queries are limited to a maximum of 300 certificates / query and may accept to 60 seconds. Consider using the search options.

For SwissSign certificates that were not issued locally, you do not have the option to request the publication of the certificate.



SwissPKI™ Dashboard PKI Management My Account Logout

Locally Issued Certificates SwissSign Certificates

### Issued Certificates | 'SwS CA'

Order UUID: ord-12345678-9abc-def0-1234-56789abcdef0 Serial number: 00D94FA7F9041813CC07FD\* Status: 7 filters Order date range: [ ]

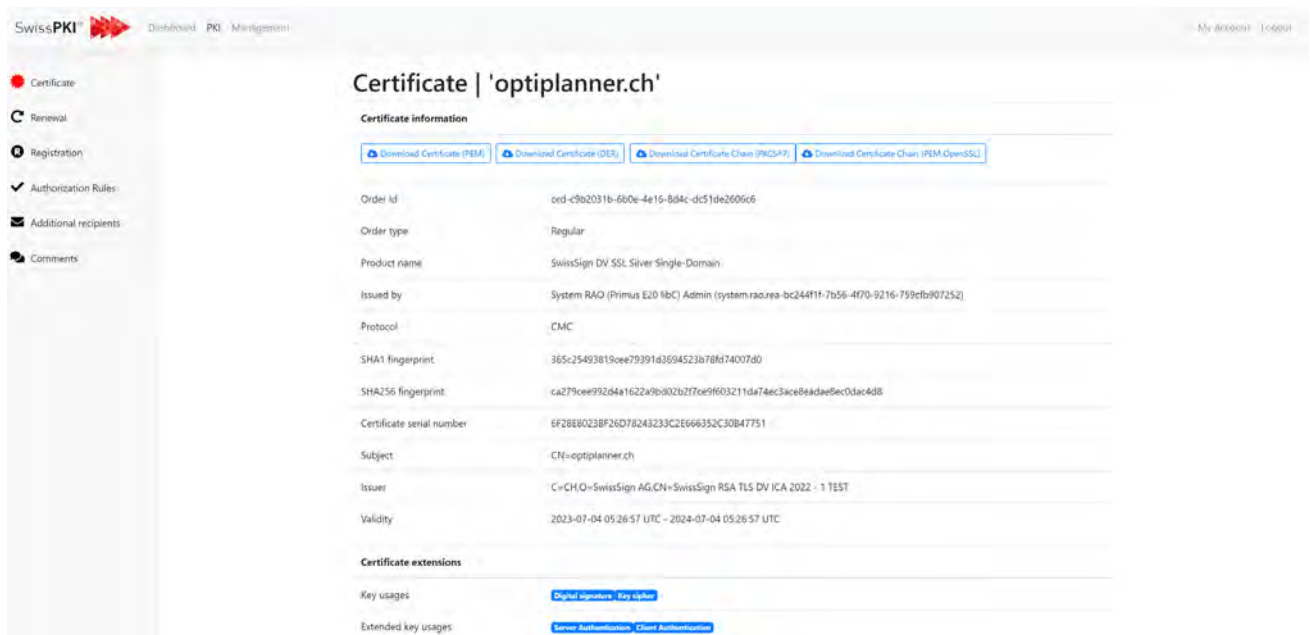
ID  Status  Client Reference  Issued By  Subject CN  Issuer  
 Product Reference  Start Validity  End Validity  Revoked On  Revocation Status  Actions

Clear Search

ID	Status	Issued By	Issuer	Product Reference	Start Validity	End Validity	Revoked On	Actions
ord-fffff093-0a9f-415e-beda-c881859a57eb Ser 04B1D58BB243BA584A46A1189F7617C92386BD07	REVOKED	lela.walter@swisspki.ch	C=CH,O=SwissSign AG,CN=SwissSign RSA TLS EV ICA 2022 - 1	pma-24a2e631- 8473-4fd0-b8ae- 021d2a3a66c9	26-07-2023	26-07-2023	17-10-2022	[i]
ord-ffffd090-d2d2-499c-9c57-cd8712c83e54 Ser -	PENDING_AUTH	lela.walter@swisspki.ch	-	pma-56c9392- 8632-4547-b56b- 8580ec2773a6	-	-	-	[i]
ord-ffff78d5-7d0f-4b17-8269-626d874f01c3 Ser 01BBE77CF366A1E82940158F6C582928731556E0	ISSUED	lela.walter@swisspki.ch	C=CH,O=SwissSign AG,CN=SwissSign RSA TLS EV ICA 2022 - 1	pma-24a2e631- 8473-4fd0-b8ae- 021d2a3a66c9	28-07-2023	28-07-2023	-	[i] [G] [A]

### 12.3.2.1.3.3 Certificate

The certificate details page contains general information about the issued certificate. The three download buttons at the top of the page allow you to download your certificate in PEM, DER, PKCS#7, or OpenSSL PEM formats.



The screenshot shows the 'Certificate | 'optiplanner.ch'' page in the SwissPKI management interface. It includes a sidebar with navigation options like Certificate, Renewal, Registration, Authorization Rules, Additional recipients, and Comments. The main content area displays 'Certificate information' with a table of details and 'Certificate extensions' with key usages.

Certificate information	
Order id	ord-c9b2031b-6b0e-4e15-8d4c-dc51de2606c6
Order type	Regular
Product name	SwissSign DV SSL Silver Single-Domain
Issued by	System RAO (Primus E20 libC) Admin (system.rao.rae-bc244f11-7b56-4f70-9216-759cfb907252)
Protocol	CMC
SHA1 fingerprint	365c25493819cee7939143694523b78f674007d0
SHA256 fingerprint	ca279cee93264a1622d9b002b27ce9f603211da74ec3ace8eadae8ec0dad488
Certificate serial number	6F28E80238F26D78243233C2E666352C30B47751
Subject	CN=optiplanner.ch
Issuer	C=CH,O=SwissSign AG,CN=SwissSign RSA TLS DV ICA 2022 - 1 TEST
Validity	2023-07-04 05:26:57 UTC - 2024-07-04 05:26:57 UTC

Certificate extensions	
Key usages	Digital signature, Key encipher
Extended key usages	Server Authentication, Client Authentication

### 12.3.2.1.3.4 Renewal

The certificate renewal rule is enabled when a manual or automatic renewal rule is set for a certificate.

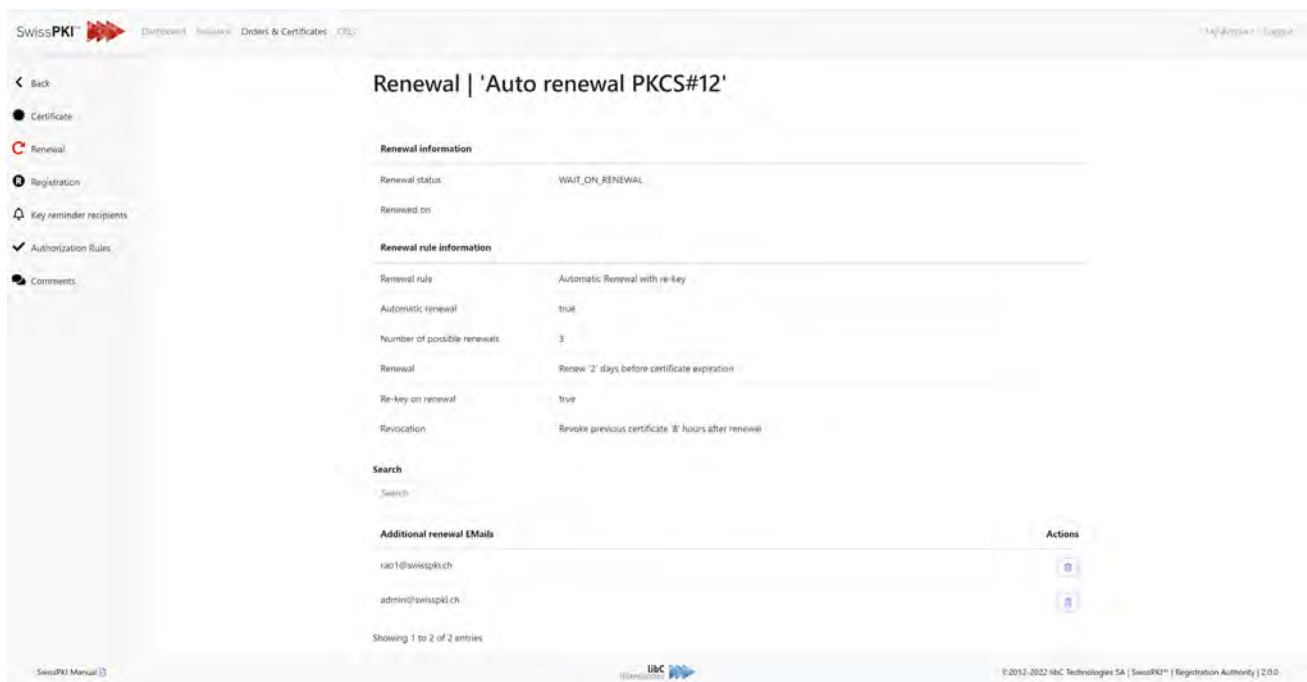
The renewal information section provides information about renewal status. The renewal status can be one of:

- WAIT\_ON\_RENEWAL indicates that there was no renewal performed yet.
- RENEWAL\_AUTHORIZATION\_PENDING indicates that a renewal started and that the request is pending authorization. This status is reached whenever an authorization on renewal is mandatory.
- ALREADY\_RENEWED indicates that the certificate did get renewed.
- RENEWAL\_LIMIT\_REACHED indicates that the maximum number of renewals was reached, and no further renewal of the certificate will occur.

The renewal rule information provides information about the specific renewal rule applied to the certificate. The renewal rule may be manual or automatic.

When renewals are executed, a list of previous certificates is displayed for the selected certificate. If the status of the renewal is `WAIT_ON_RENEWAL`, the list is not displayed as no renewal did occur yet.

The additional renewal Email section displays a list of the additional renewal emails. Optionally, you can add/remove additional email recipient. The maximum number of renewal emails is 5.



The screenshot shows the SwissPKI web interface. The main content area is titled "Renewal | 'Auto renewal PKCS#12'". It contains several sections:

- Renewal information:**
  - Renewal status: `WAIT_ON_RENEWAL`
  - Renewed on: (empty)
- Renewal rule information:**
  - Renewal rule: Automatic Renewal with re-key
  - Automatic renewal: `true`
  - Number of possible renewals: `3`
  - Renewal: Renew 21 days, before certificate expiration
  - Re-key on renewal: `true`
  - Revocation: Revoke previous certificate 8 hours after renewal
- Search:** A search input field.
- Additional renewal Emails:** A table with two columns: "Additional renewal Emails" and "Actions".
  - Row 1: `rao1@swisspk.ch` with a plus icon in the Actions column.
  - Row 2: `admin@swisspk.ch` with a plus icon in the Actions column.

At the bottom of the table, it says "Showing 1 to 2 of 2 entries".

### 12.3.2.1.3.5 Publications

If the certificate is associated with one or several publishers, information about certificate publications can be found on this page. Every publication event concerning the certificate will be listed here and the option to un-publish will be available as well.



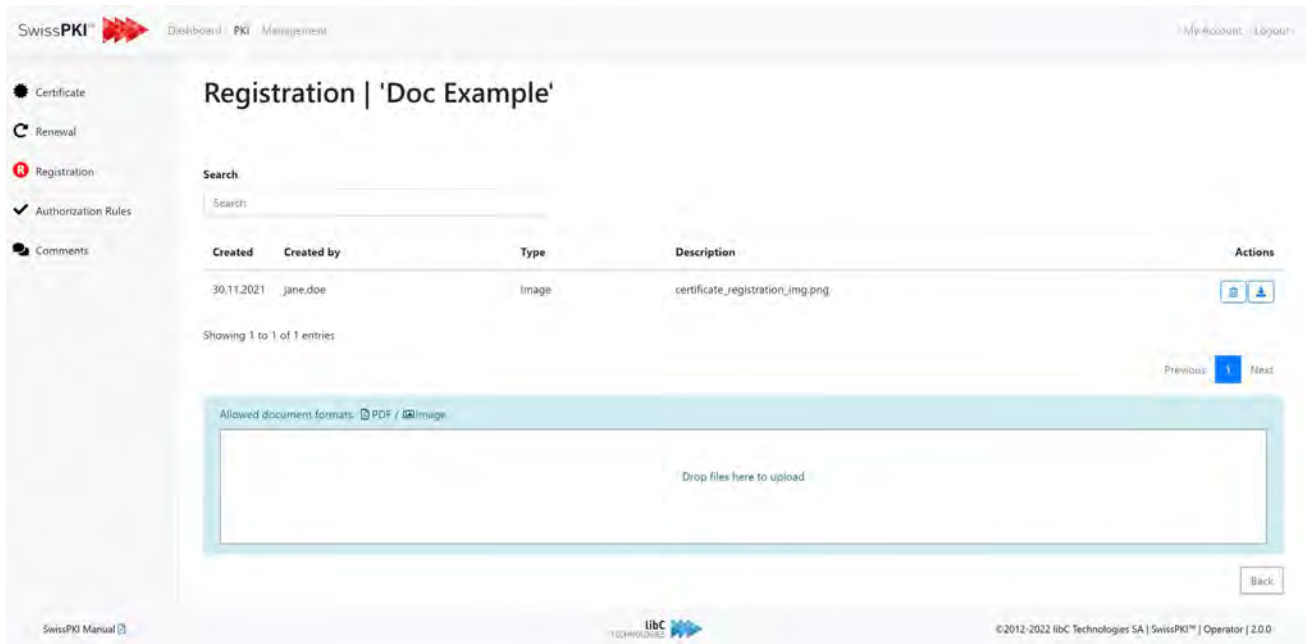
The screenshot shows the SwissPKI web interface. The top navigation bar includes 'SwissPKI', 'Dashboard', 'Issuance', 'Orders & Certificates', and 'CRLs'. The user is logged in as 'My Account' and 'Logout'. The main content area is titled 'Publications | 'Sample End User''. It displays a table with two entries:


Created	Status	Type	Name	Actions
01.02.2022 06:50	UNPUBLISHED	LDAP	ldap.swisspki.com	
01.02.2022 06:50	PUBLISHED	LDAP	ldap.swisspki.com	

Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom right, there are 'Previous' and 'Next' navigation buttons, with '1' selected.

### 12.3.2.1.3.6 Registration

When a registration rule is enabled for the certificate, you can add/remove images or PDF documents to the certificate using the drag/drop box below the document list. Depending on the registration rule's settings, you may be allowed to add only PDF documents, images (jpeg or png) or both.





SwissPKI  Dashboard PKI Management My Account Logout

**Registration | 'Doc Example'**



Search

Search

Created	Created by	Type	Description	Actions
30.11.2021	jane.doe	Image	certificate_registration_img.png	 


Showing 1 to 1 of 1 entries

Previous **1** Next

Allowed document formats:  PDF /  Image

Drop files here to upload

[Back](#)

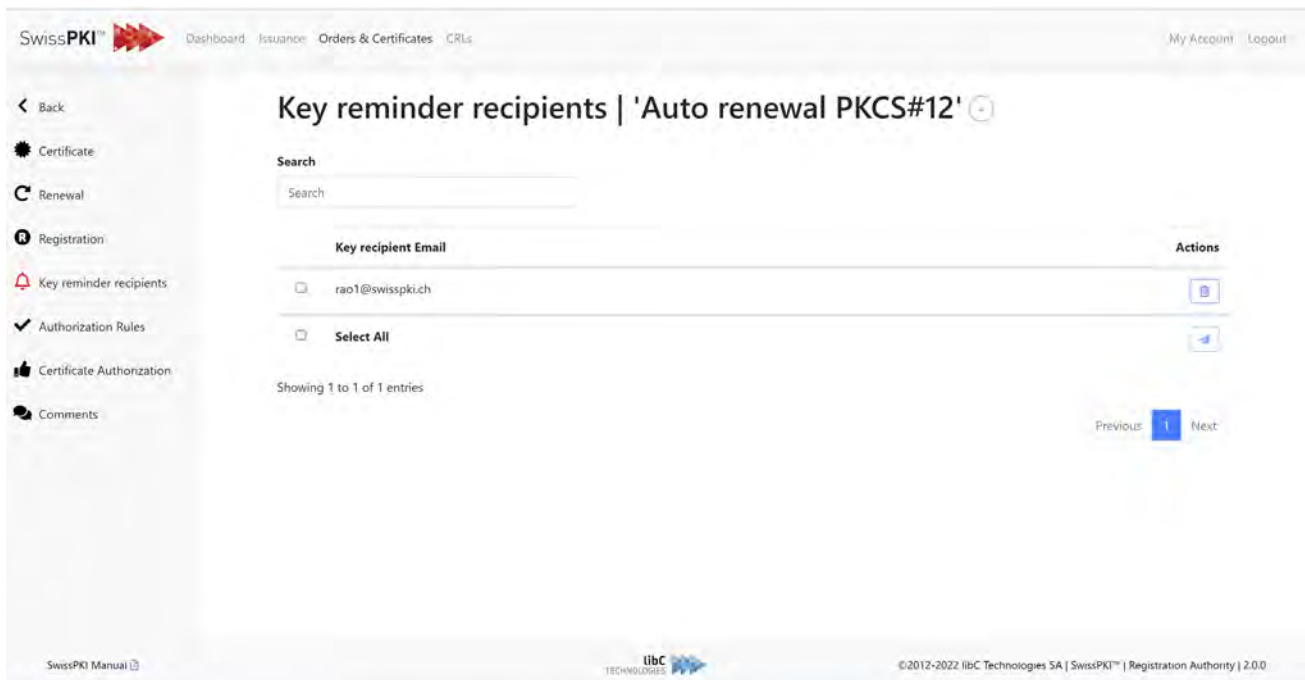
SwissPKI Manual  libC TECHNOLOGIES ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.3.2.1.3.7 Key Reminder Recipients



When the issued certificate's key generation policy is of type PKCS#12, the 'Key Reminder Recipients' menu is enabled. This section allows you to define recipient emails which will receive notifications for PKCS#12 downloads.

When issuing a leaf certificate with a key generation policy of type PKCS#12, the RA Officer must provide at least one recipient Email for the PKCS#12 download.

The PKCS#12 download link redirects the recipient to a Self-service page where he/she needs to provide the TOTP (issued with the notification) and provide a PIN for securing the PKCS#12. The PKCS#12 is packaged using the recipient provided PIN and emailed to the recipient list.



The screenshot shows the SwissPKI web interface. The top navigation bar includes 'Dashboard', 'Issuance', 'Orders & Certificates', and 'CRLs'. The left sidebar contains a menu with items like 'Certificate', 'Renewal', 'Registration', 'Key reminder recipients', 'Authorization Rules', 'Certificate Authorization', and 'Comments'. The main content area is titled 'Key reminder recipients | 'Auto renewal PKCS#12''. It features a search bar and a table with the following content:

Key recipient Email	Actions
<input type="checkbox"/> rao1@swisspki.ch	
<input type="checkbox"/> <b>Select All</b>	

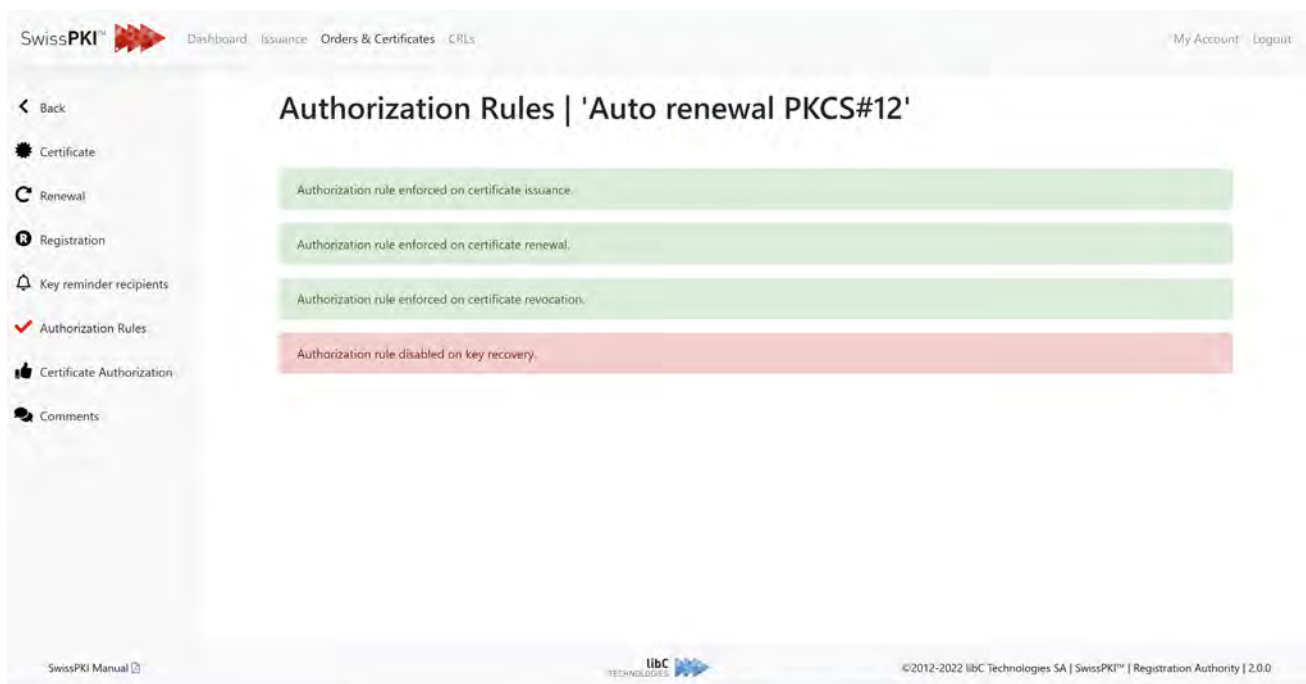
Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom right of the table area, there are 'Previous' and 'Next' navigation buttons. The footer of the page includes 'SwissPKI Manual', the libC Technologies logo, and the copyright notice: '©2012-2022 libC Technologies SA | SwissPKI™ | Registration Authority | 2.0.0'.

**Note:** This option pane is not available if the key generation is PKCS#12 with PIN or PKCS#12 with CA PIN. In this key pair generation mode, the end user must provide the PKCS#12 protection PIN before key generation. This implies that the PKCS#12 private key cannot be escrowed and therefore not available for download to other recipients for recovery.

### 12.3.2.1.3.8 Authorization Rules

The Authorization Rules tab inform you about the enabled authorizations on the issued certificate. Authorizations can be any of:

- Authorization on certificate issuance
- Authorization on certificate renewal
- Authorization on certificate revocation
- Authorization on key recovery




The following example shows a certification authorization rule.

When certificate issuance authorization is active, the issued certificate order enters an authorization state and informs authorizers to accept or reject the issuance workflow.







A user with an Authorizer role searches for pending certificate issuance authorizations. From the RA web interface, the logged in authorizer accesses the authorization details by clicking on the 'edit' button of the pending requests.




SwissPKI  Dashboard My Account Logout

Order UUID:  Auth UUID:  Status:

CA:  Type:  Date range:

ID	Status	CA	Policy	Type	Created	Actions
 ord-488ff3f-89ee-4605-8a85-2809980192ba  ora-f0228027-c7dc-4b34-8be0-a0dcf7eda2b5	PENDING	CA	General	ISSUANCE	03.12.2021	
 ord-a2428937-748a-4f18-9c1e-1ffdb4cab6f6  ora-57f06242-9984-44fd-8b42-948b6919f305	ACCEPTED	CA	General	ISSUANCE	03.12.2021	

Showing 1 to 2 of 2 entries

SwissPKI Manual [↗](#)  ©2012-2021 libC Technologies SA | SwissPKI™ | Registration Authority | 2.0.0

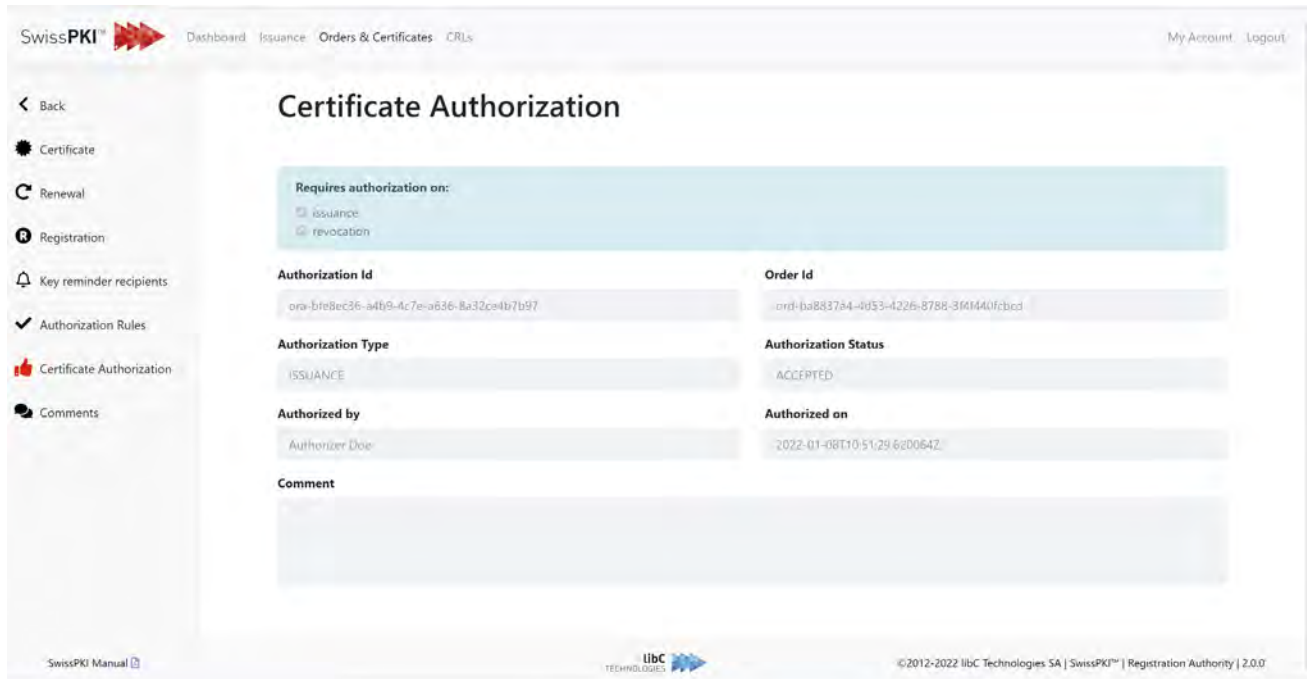




### 12.3.2.1.3.9 Certificate Authorization

When authorizations are enabled for a certificate, the Certificate Authorization section provides detailed information about the authorization events.

The illustration below displays information about a certificate issuance authorization accepted by authorizer 'Authorizer Doe'



The screenshot shows the 'Certificate Authorization' page in the SwissPKI interface. The page title is 'Certificate Authorization'. A summary box indicates 'Requires authorization on: Issuance, Revocation'. The main details are as follows:

<b>Authorization Id</b>	ora-bfe8cc36-a4b9-4c7e-a836-8a32ce4b7b97	<b>Order Id</b>	ord-ba883784-8b53-4226-8788-3f4440fcbcd
<b>Authorization Type</b>	ISSUANCE	<b>Authorization Status</b>	ACCEPTED
<b>Authorized by</b>	Authorizer Doe	<b>Authorized on</b>	2022-01-08T10:51:29.6e0064Z
<b>Comment</b>			

For multi-authorization issuance the table with all comments, editions and approvals by all authorizers is shown additionally.

Status	Created	Name	Username	Comment	Actions
ACCEPTED	11.08.2023 09:41	Autho Rizer	auth.a2	Ok, let's issue	
ACCEPTED	11.08.2023 09:40	Authy Rizer	auth.a	You're right - I approve	
EDITED	11.08.2023 09:39	Autho Rizer	auth.a2	No, it's France	<a href="#">i</a>
EDITED	11.08.2023 09:38	Authy Rizer	auth.a	changed it to Germany now	<a href="#">i</a>
COMMENTED	11.08.2023 09:37	Authy Rizer	auth.a	I think it should be Germany	

Showing 1 to 5 of 5 entries

Previous **1** Next

Clicking on the info icon, will show a read-only view of the policy including all the values of the edition at that time.

Policy Template ✕

^ Key Generation parameters

<b>Key generation source</b>	<b>Key type and minimum size</b>	<b>Certificate Hash Algorithm</b>
PKCS10 ▾	RSA_204 ▾	SHA256 ▾

^ Subject Distinguished Name

Unused Subject Attributes from CSR: c=CH,state=Zurich,l=Thalwil

General Name	Encoding	Value
cn ▾	utf8 ▾	libc.ch <span style="float: right;">* required</span>
o ▾	bmp ▾	libC Technologies SA <span style="float: right;">* required</span>
ou ▾	bmp ▾	SPKI Development <span style="float: right;">* required</span>
c ▾	printable ▾	FR <span style="float: right;">* required</span>

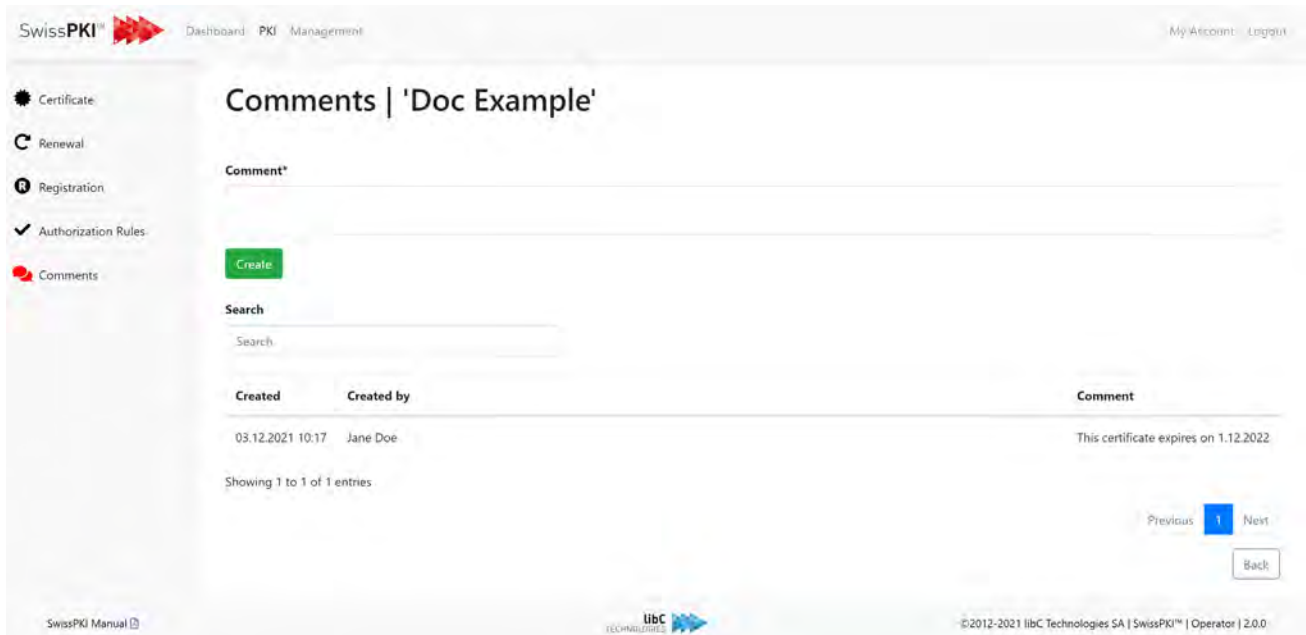
^ Certificate validity

<b>Validity</b>	<b>Duration</b>
years ▾	13

OK

### 12.3.2.1.3.10 Comments

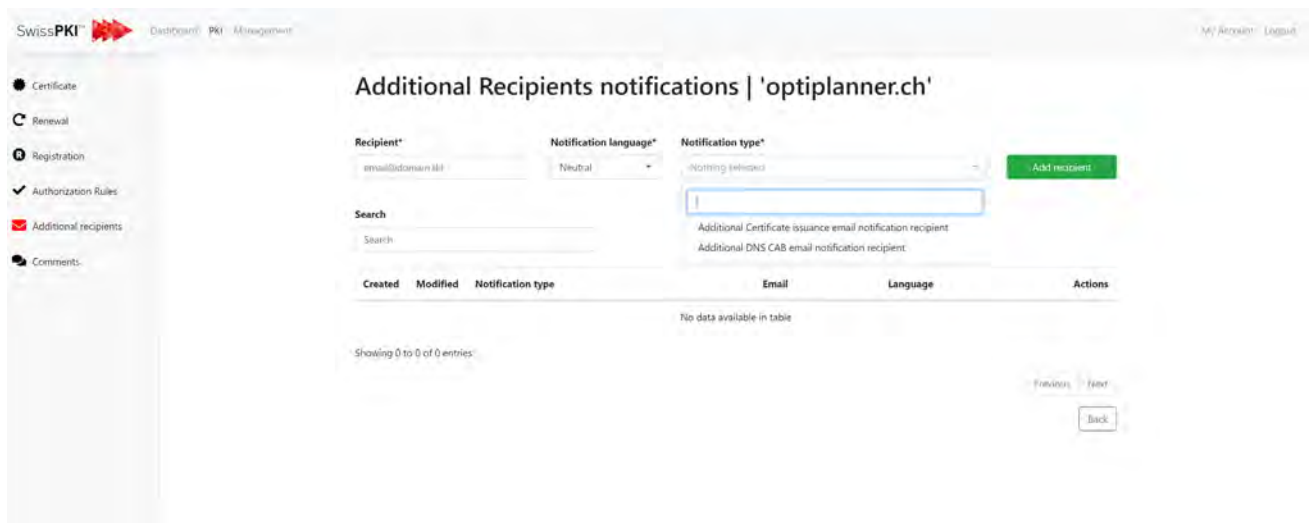
Comments can be added to your certificate. To add a comment, simply write it in the textbox at the top of the page and then click on the create button.



The screenshot shows the SwissPKI web interface. The top navigation bar includes 'SwissPKI', 'Dashboard', 'PKI Management', and 'My Account | Logout'. A left sidebar contains navigation links for Certificate, Renewal, Registration, Authorization Rules, and Comments. The main content area is titled 'Comments | 'Doc Example'' and features a 'Comment\*' text input field with a green 'Create' button below it. A search bar is also present. Below the search bar is a table with columns 'Created', 'Created by', and 'Comment'. One entry is shown: '03.12.2021 10:17' by 'Jane Doe' with the comment 'This certificate expires on 1.12.2022'. The table indicates 'Showing 1 to 1 of 1 entries'. At the bottom right of the table are 'Previous', '1', 'Next', and 'Back' buttons. The footer contains 'SwissPKI Manual', the libC Technologies logo, and copyright information: '© 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.3.2.1.3.11 Additional Recipients

When additional recipients are enabled on the notification templates (see section 12.2.5 Notifications Templates), the RA Operator has the possibility to manage registered recipients for this specific instance of the certificate. The list of available notification types depends on the notification template settings.

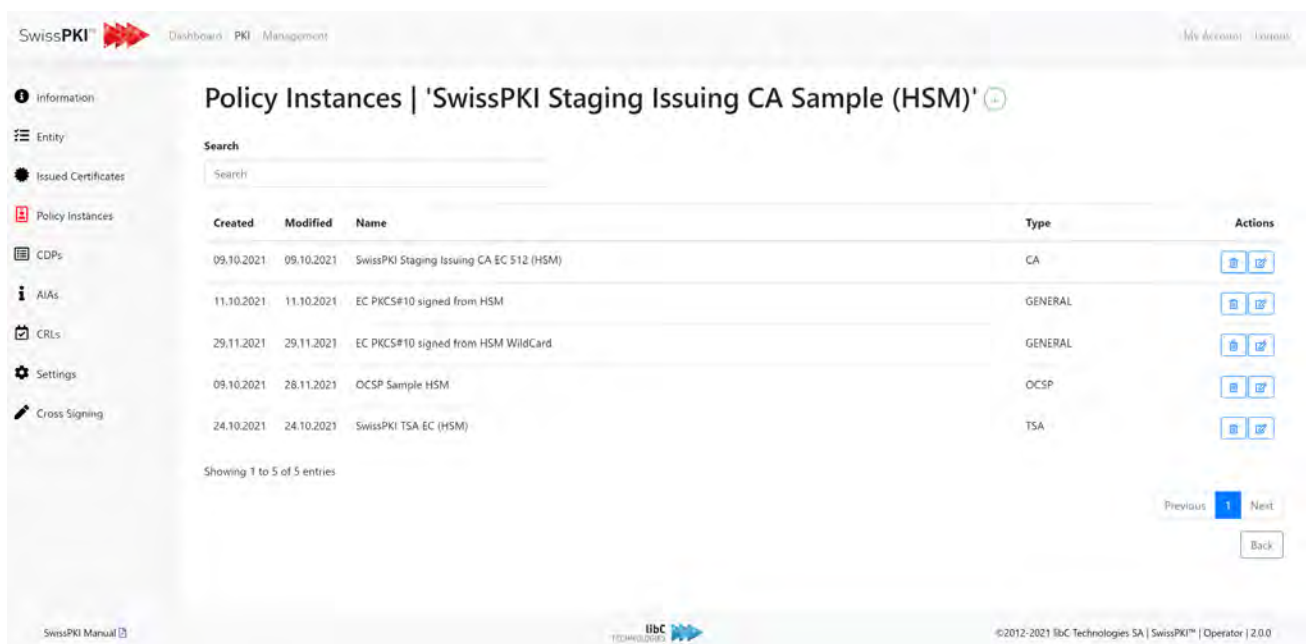


### 12.3.2.1.4 Policy Instances

Applies to Certification Authorities of type **SwissPKI**, **External** and **SwissSign**.

A Policy Instance is the assignment of a Certificate Policy Template to an Issuing Certification Authority. The assignment indicates that the Issuing Certification Authority is allowed to issue certificates. You assign a Certificate Policy Template by clicking on the “+” sign. A list of Certificate Policy Templates is displayed.

**Note:** a Certificate Policy Template can be assigned multiple time to an Issuing Certification Authority. This allows for a same type of certificate template to be issued using different validation rules at the Policy Instance level. The name of the assigned certificate policy template should be edited along with the description to avoid confusion working with identical Policy Instance names.



Created	Modified	Name	Type	Actions
09.10.2021	09.10.2021	SwissPKI Staging Issuing CA EC 512 (HSM)	CA	[Edit] [Delete]
11.10.2021	11.10.2021	EC PKCS#10 signed from HSM	GENERAL	[Edit] [Delete]
29.11.2021	29.11.2021	EC PKCS#10 signed from HSM WildCard	GENERAL	[Edit] [Delete]
09.10.2021	28.11.2021	OCSP Sample HSM	OCSP	[Edit] [Delete]
24.10.2021	24.10.2021	SwissPKI TSA EC (HSM)	TSA	[Edit] [Delete]

1. If your Issuing CA is of type **SwissPKI**, then you can assign any type of Certificate Policy Templates to it.
2. If your Issuing CA is of type **External**, then you only can assign Certificate Policy Template of type *EXTERNAL*. The management of the assigned Policy Instance is also reduced with the settings.
3. If your Issuing CA is of type **SwissSign**, then you only can assign Certificate Policy Templates of type *SwissSign Public Trust* and *Microsoft Public Trust*. The management of the assigned Policy Instance is also reduced with the settings




Deleting a Policy Instance removes it from all Clients or PKI Entities. If no certificate was issued, then the Policy Instance is deleted. If certificates were issued, then the Policy Instance is retired. In both cases, the Policy Instance is not available to end users.

Refer to ***Error! Reference source not found. Error! Reference source not found.*** for Certificate Policy Template types.

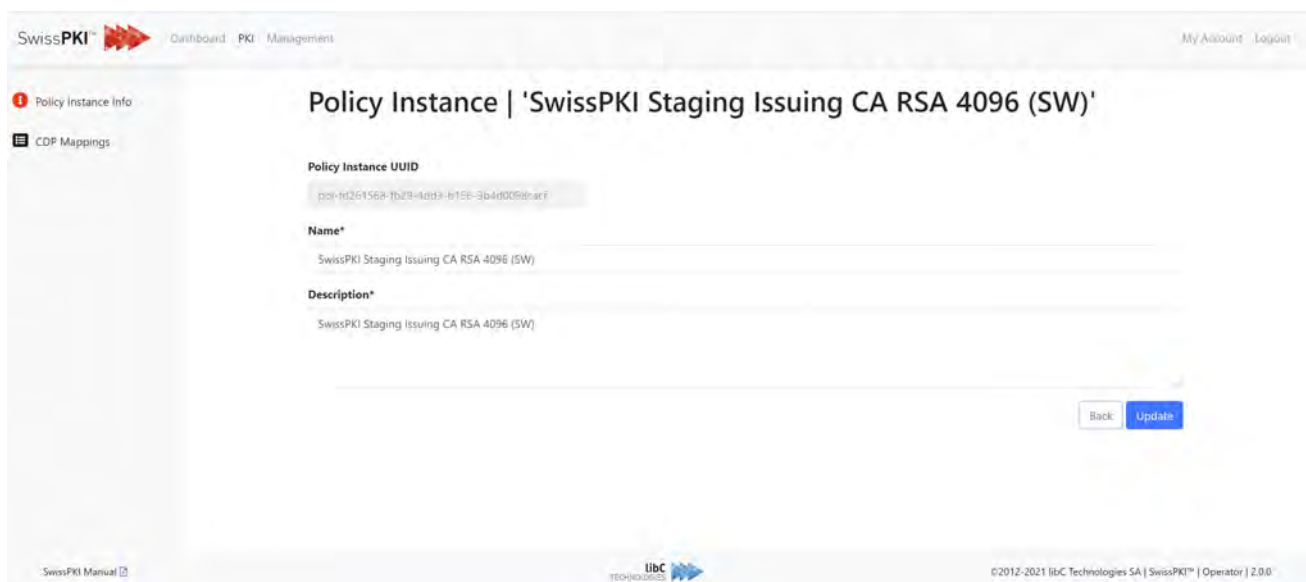
### 12.3.2.1.4.1.1 Policy Instance

Before using a Policy Instance for issuing certificates, you must configure its attributes depending on the Certificate Policy Template type:


1. If the Certificate Policy Template is of type **PKI Entity** (see **Error! Reference source not found. Error! Reference source not found.**), then you may configure the CDPs for the Policy Instance if the Certificate Policy Template includes CDP extensions. Alternatively, you can set the CDP (see 12.3.2.1.5 CRL Distribution Points).
2. If the Certificate Policy Template is of type **End User** (see **Error! Reference source not found. Error! Reference source not found.**), then you may configure the CDPs for the Policy Instance if the Certificate Policy Template includes CDP extensions. Alternatively, you can set the CDP (see 12.3.2.1.5 CRL Distribution Points). Additionally, you must assign the Policy instance to a Client (see 12.2.3 Clients) to expose the certificate template to the end user or end user protocols.

By default, the Policy Instance name and description is taken from the Certificate Policy Template when making the assigning. You should override the name and description with meaningful values as the name and description are displayed to the end user. Edit the Policy Instance from the list using the edit  button.

PKI Entity Policy Instance configuration consists only of naming and CDP mappings.



Whereas End User Policy Instance configuration involves defining *Policy Instance Validators* and *Client Protocol Mappings*

SwissPKI™  Dashboard PKI Management My Account Logout



**Policy Instance | 'Sample SSL Server'**

**Policy Instance UUID**  
c01-398ee499-eb3a-4bb0-977d-f11bc233869

**Name\***  
Sample SSL Server

**Description\***  
This is a sample to show the product description

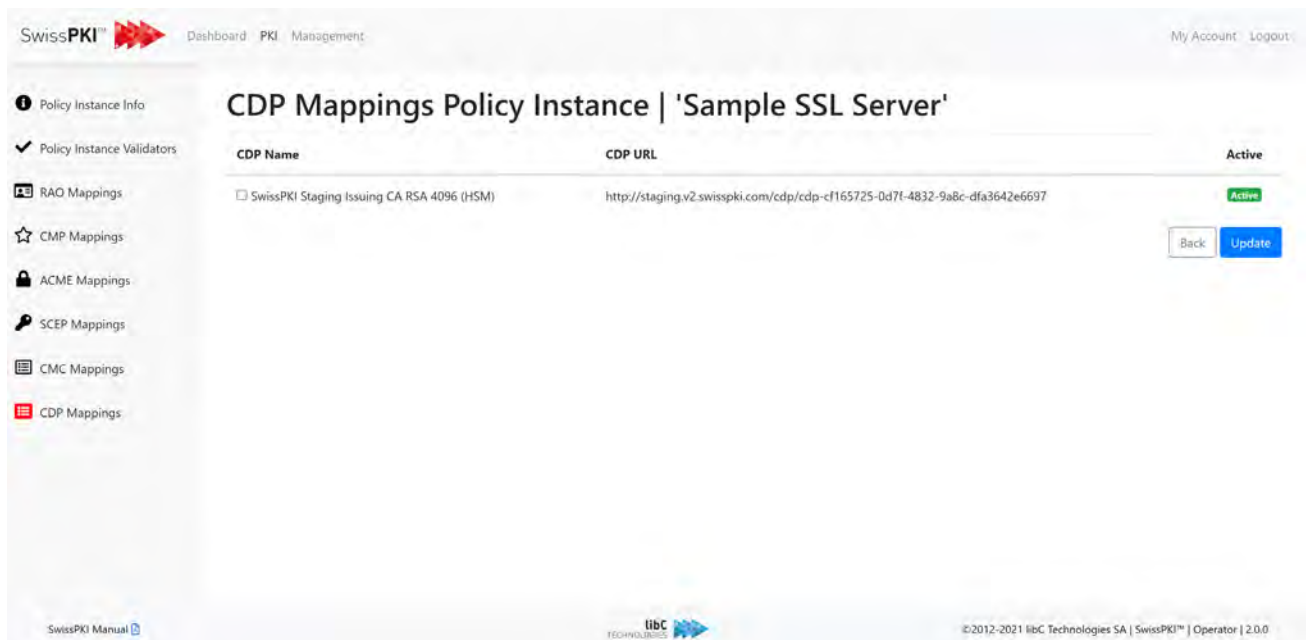
[Back](#) [Update](#)

SwissPKI Manual   © 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.3.2.1.4.1.1.1 CDP Mappings

Select the associated CDPs if the Certificate Policy Template enforces CDP extensions. The operator, whether CA Operator or RA Officer, will not be allowed to issue a certificate if the CDP mapping is omitted.

**Note:** you may add multiple CDPs to a Policy Instance.



The screenshot shows the SwissPKI web interface. The main heading is "CDP Mappings Policy Instance | 'Sample SSL Server'". Below this is a table with columns for "CDP Name", "CDP URL", and "Active".

CDP Name	CDP URL	Active
<input type="checkbox"/> SwissPKI Staging Issuing CA RSA 4096 (HSM)	http://staging.v2.swisspki.com/cdp/cdp-cf165725-0d7f-4832-9a8c-dfa3642e6697	Active

At the bottom right of the table, there are "Back" and "Update" buttons. The footer of the page includes "SwissPKI Manual", the libC TECHNOLOGIES logo, and the copyright notice "© 2012-2021 libC, Technologies SA | SwissPKI™ | Operator | 2.0.0".

### 12.3.2.1.4.1.1.2 Policy Instance Validators

Policy Instance validation enables you to define certificate content validation identical to all associated Client Protocol Mappings. This type of validation is typically used when validating runtime content when deploying SwissPKI within an organization. For individual Client content validation, please refer to 0

### *Client Protocol Mappings.*

In addition to Policy Instance validation, you also have the possibility to use custom validator by implementing Client validation Rules (see *12.2.3.6 Client Validation Rules*)

#### Validator types

1. Regex validator
2. SAN Domain Name validator
3. Client domain validator (see *12.2.3.12 Client Domains*)
4. SDN validator
5. ETSI Validator
6. CN and/or MAIL match SAN RFC822 validator
7. CN and/or DNS match SAN validator
8. Overwrite Subject Distinguished Name values
9. Fill in Serial Number with UUID if MAIL is not present in SDN or match MAIL to match at least one SAN RFC822 if present
10. Require a pseudo or first/last name but no EMAIL in the CN
11. Fill in Serial number with UUID if not present
12. WWW domain name validator
13. Wildcard base domain validator



### *Client Protocol Mappings.*

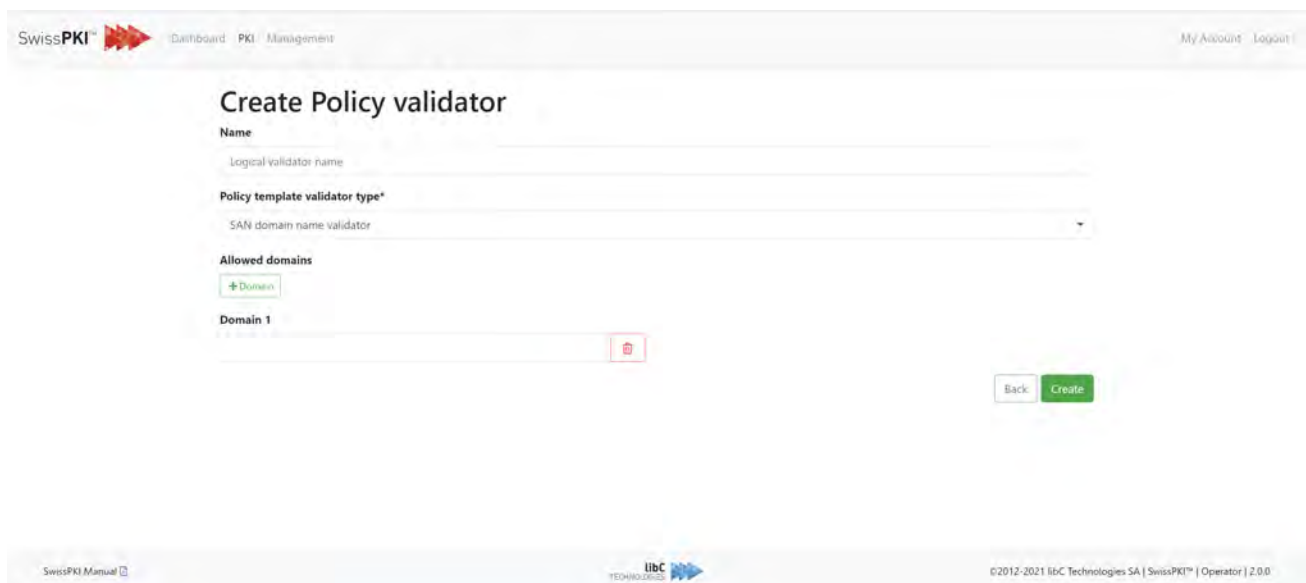
In addition to Policy Instance validation, you also have the possibility to use custom validator by implementing Client validation Rules (see *12.2.3.6 Client Validation Rules*)

#### Validator types

1. Regex validator
2. SAN Domain Name validator
3. Client domain validator (see *12.2.3.12 Client Domains*)
4. SDN validator
5. ETSI Validator
6. CN and/or MAIL match SAN RFC822 validator
7. CN and/or DNS match SAN validator
8. Overwrite Subject Distinguished Name values
9. Fill in Serial Number with UUID if MAIL is not present in SDN or match MAIL to match at least one SAN RFC822 if present
10. Require a pseudo or first/last name but no EMAIL in the CN
11. Fill in Serial number with UUID if not present
12. WWW domain name validator
13. Wildcard base domain validator

### 12.3.2.1.4.1.1.2.1 SAN Domain Name validator

1. Validate content using a SAN Domain validator. Inspects the content of CN, Email, SAN DNS, and SAN RFC 822. The value validates with *'ends with'* case insensitive
2. Multiple validation rules can be associated to the Policy Instance.
3. At least one rule per validator must validate.




SwissPKI Dashboard PKI Management My Account Logout

### Create Policy validator

**Name**  
Logical validator name

**Policy template validator type\***  
SAN domain name validator

**Allowed domains**  
+ Domain

**Domain 1** 

Back Create

SwissPKI Manual libC TECHNOLOGIES ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

Field	Description
<b>Name</b>	Logical name of the validation rule
<b>Type</b>	SAN domain name validator
<b>Attributes</b>	List of allowed domains Content check: <i>'ends with'</i> case insensitive



### 12.3.2.1.4.1.1.2.2 CN and/or EMAIL SAN RFC822 validator

1. Validation rule to enforce that the CN and/or MAIL Subject DN attribute value matches at least one of the SAN RFC822 filed value in the issued certificate

## Create Policy validator

Name\*

Logical validator name

Policy template validator type\*

CN and/or MAIL match SAN RFC822 validator

Check that at least one SAN RFC822 matches Subject Name attribute values.

Back

Create

### 12.3.2.1.4.1.1.2.3 CN and/or DNS match SAN validator

1. Validation rule to enforce that the CN Subject DN attribute value matches at least one of the SAN DNS filed value in the issued certificate

## Create Policy validator

Name\*

Logical validator name

Policy template validator type\*

CN and/or DNS match SAN validator

Check that at least one SAN DNS matches Subject Name attribute values. Handles wildcard domains when enabled on certificate policy template.

Back

Create

### 12.3.2.1.4.1.1.2.4 ETSI validator

Validation rule to enforce that the Subject DN attribute values match ETSI NCP, OVCP or EVCP rules. Each rule also validates the certificate policy CP Object Identifier for its presence and correct value.

## Create Policy validator

**Name\***

Logical validator name

---

**Policy template validator type\***

ETSI validator

---

**ETSI validation type\***

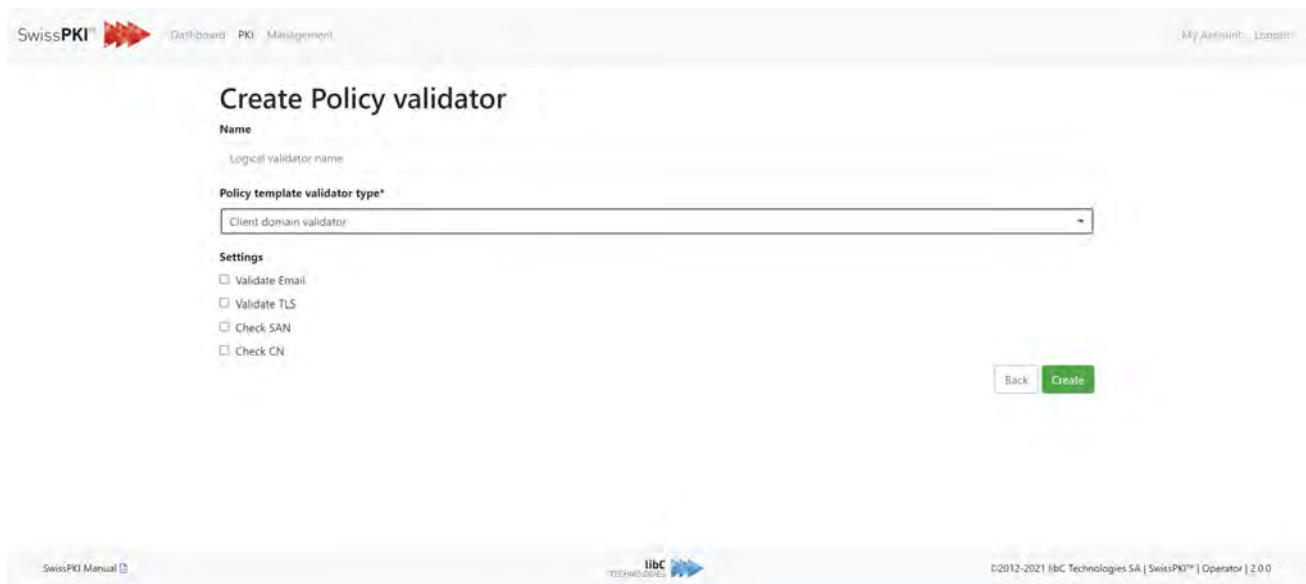
Nothing selected

- extended-validation-certificate-policy
- organizational-validation-certificate-policy
- normalized-certificate-policy
- qualified-validation-certificate-policy

Type	Description
<b>EVCP</b>	A certificate policy with OID 0.4.0.2042.1.4 State or Locality set in subject distinguished name Jurisdiction of Incorporation State or Jurisdiction of Incorporation Locality set in subject distinguished name
<b>OVCP</b>	A certificate policy with OID 0.4.0.2042.1.7 State or Locality set in subject distinguished name
<b>NCP</b>	A certificate policy with OID 0.4.0.2042.1.1 State or Locality set in subject distinguished name
<b>Qualified</b>	A certificate policy with OID 0.4.0.194112.1.2

### 12.3.2.1.4.1.1.2.5 Client domain validator

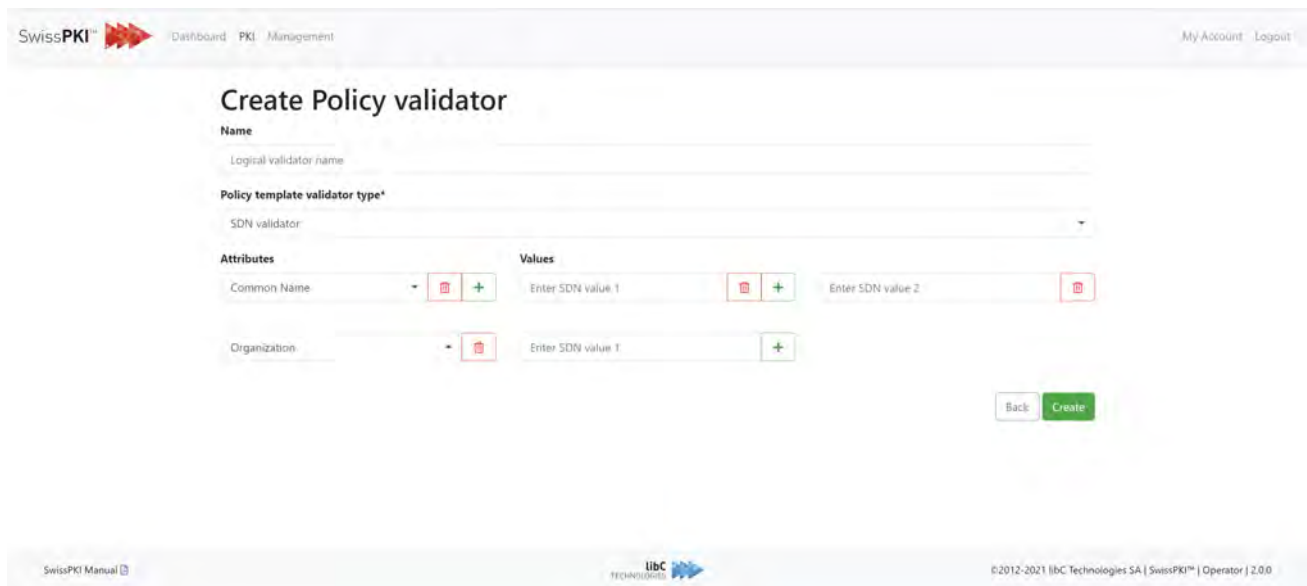
1. Validate content using a Client domain validator (see 12.2.3.12 Client Domains).
2. Multiple validation rules can be associated to the Policy Instance.
3. At least one rule per validator must validate.



Field	Description
<b>Name</b>	Logical name of the validation rule
<b>Type</b>	Client domain validator
<b>Attribute</b>	Using the defined list of Client domain values, inspect CN, Email in Subject Distinguished name, SAN email and DNS Content check: <i>'ends with'</i> case insensitive

### 12.3.2.1.4.1.1.2.6 SDN validator

1. Validate content using a Subject Distinguished Name validator.
2. Multiple validation rules can be associated to the Policy Instance.
3. At least one rule per validator must validate.



Field	Description
<b>Name</b>	Logical name of the validation rule
<b>Type</b>	SDN validator
<b>Attributes</b>	List of Subject Distinguished Name attributes as per definition in <i>12.3.1.2.4 Subject Distinguished Name</i> Multiple values 'exact match' case insensitive.

### 12.3.2.1.4.1.1.2.7 SDN overwrite

1. Overwrite the SubjectDN attribute content (you can use the overwrite policy editor option set to have the fields non-visible and non-editable)
2. Multiple validation rules can be associated to the Policy Instance.

## Create Policy validator

**Name\***

Logical validator name

**Policy template validator type\***

Overwrite Subject Distinguished Name values

**Attributes**

Locality

**Values**

Zurich

Field	Description
<b>Name</b>	Logical name of the validation rule
<b>Type</b>	Overwrite Subject Distinguished Name values
<b>Attributes</b>	List of Subject Distinguished Name attributes as per definition in <i>12.3.1.2.4 Subject Distinguished Name</i>  Unique value 'exact match' case insensitive.

### 12.3.2.1.4.1.1.2.8 Serial Number with auto generated UUID

Check that we have a matching Email attribute value in the SAN RFC822 or generate random UUID to UUID serial number attribute field.

Recommendations for the certificate policy template:

- for Mail attribute is SDN: visible, non-mandatory editable
- for Serial Number: non-mandatory, non-editable, override. You can set it to visible or not, whatever you prefer

## Create Policy validator

Name\*

Policy template validator type\*

Check that the Mail attribute matches at least one SAN RFC822 when present. If Mail is not present, then generate a random serial number attribute value.

### 12.3.2.1.4.1.1.2.9 Require pseudo

Subject DN CN attribute must not contain an Email. The CN attribute can start with 'pseudo:' or match first/last name pattern with space or coma.

#### Create Policy validator

**Name\***

**Policy template validator type\***

Check that the CN does not contain an email. CN must start with 'pseudo:' or represent a first/last name separated with dot, coma or space (special characters are excluded).

### 12.3.2.1.4.1.1.2.10 Serial Number

Fill in automatically a Serial Number in the Subject Distinguished Name if attribute value is not present. Note that the certificate policy template must have the flag settings 'overwrite' enable for the validator to be invoked.

#### Create Policy validator

**Name\***

**Policy template validator type\***

Fill in a random serial number attribute value if not present.

### 12.3.2.1.4.1.1.2.11 WWW Domain Name

For single domain products, only one Subject Alternative Name is allowed with the optional 'www' domain entry. If two SAN entries are included, this validator checks that one of them is equal to the other with the addition of 'www..' *Ex: example.com / www.example.com*

### Create Policy validator

**Name\***

**Policy template validator type\***

Only one Subject Alternative Name is allowed (With optional www domain entry Ex: example.com / www.example.com)

[Back](#) [Create](#)

### 12.3.2.1.4.1.1.2.12 Wildcard Base Domain

For wildcard products, the user has the option to also include the wildcard base domain. If two SAN entries are included, this validator checks that one is a wildcard domain and the other it is base domain. *Ex: \*.example.com / example.com*

### Create Policy validator

**Name\***

**Policy template validator type\***

Only one Subject Alternative Name is allowed (With optional wildcard base domain entry Ex: \*.example.com / example.com)

[Back](#) [Create](#)



### 12.3.2.1.4.1.1.3 Client Protocol Mappings

Based on Certificate Policy Template End User type associated to the Policy Instance for the Issuing Certification Authority, you correlate and define the behavior of Policy Instance for a selected Client.

Key generation	Type	Allowed Client protocol
PKCS#10	General	RAO, CMP, ACME, SCEP and CMC
PKCS#10	SCION	SCION
PKCS#10	Microsoft	Microsoft CES/CEP
PKCS#12	General	RAO

You associate a Policy Instance with Clients



The screenshot shows the SwissPKI management interface. The main heading is "RAO Mappings Policy Instance | 'Doc Example'". On the left, there is a navigation menu with options: Policy Instance Info, Policy Instance Validators, RAO Mappings (selected), CMP Mappings, ACME Mappings, SCEP Mappings, CMC Mappings, and CDP Mappings. The main content area displays a table with the following columns: Created, Modified, Expires on, Client name, and Actions. A single entry is shown with "Created" and "Modified" dates of 30.11.2021 and "Client name" as "Client A". Below the table, it says "Showing 1 to 1 of 1 entries". Navigation buttons for "Previous", "Next", and "Back" are visible. The footer includes "SwissPKI Manual", the libC Technologies logo, and copyright information: "©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".

### 12.3.2.1.4.1.1.3.1 RAO Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance to the Registration Authority UI for the selected Client.

You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Recovery Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	Rule definition reference, see 12.2.4.1 Registration Rules
<b>Renewal Rule</b>	Rule definition reference, see 12.2.4.3 Renewal Rules
<b>Publish certificate</b>	Publishes the certificate via the associated Publisher instance(s), see 12.3.2.7 Publisher
<b>Allow client publication override</b>	Allows the Client to override publication when issuing a certificate. Only available when the Issuing CA is associated with a Publisher instance(s), see 12.3.2.7 Publisher
<b>Default publication override value</b>	If the Client is allowed to override publication settings, then this check box defines the default behavior when displaying the option in the Registration UI.
<b>Enable for API Access</b>	Allow accounts to issue via REST API
<b>Generate revocation code for use in Self Service</b>	When enabled, the issuing CA generates a revocation code delivered to the recipient in the form of an HTTP link. The revocation code can then be used by the recipient to revoke the certificate via the self-service page.
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier

<b>Registration sources</b>	List of registration sources used to select user/system information when issuing the certificate
<b>Validators</b>	See <i>12.3.2.1.4.1.1.2 Policy Instance Validators</i>

### Notifications

#### Certificate issuance notification

Certificate Issuance

#### Certificate revocation notification

Certificate Revocation

#### Certificate recovery notification

No selection

### Rules

#### Authorization rule

No selection

#### Registration rule

No selection

#### Renewal rules

Manual Renewal RAC

- Publish certificate
- Enabled for API access
- Generate revocation code for use in Self Service
- Allow client publication override ⓘ
- Default publication override value (check/uncheck)

#### Expiration date

tt.mm.jjjj

#### External Reference

EV

#### Partner Reference

### Registration Sources

#### Registration sources (multiple)

Nothing selected

[Back](#) [Update](#)

## Create validator

#### Search

Search

Created	Modified	Name	Rule	Actions
---------	----------	------	------	---------

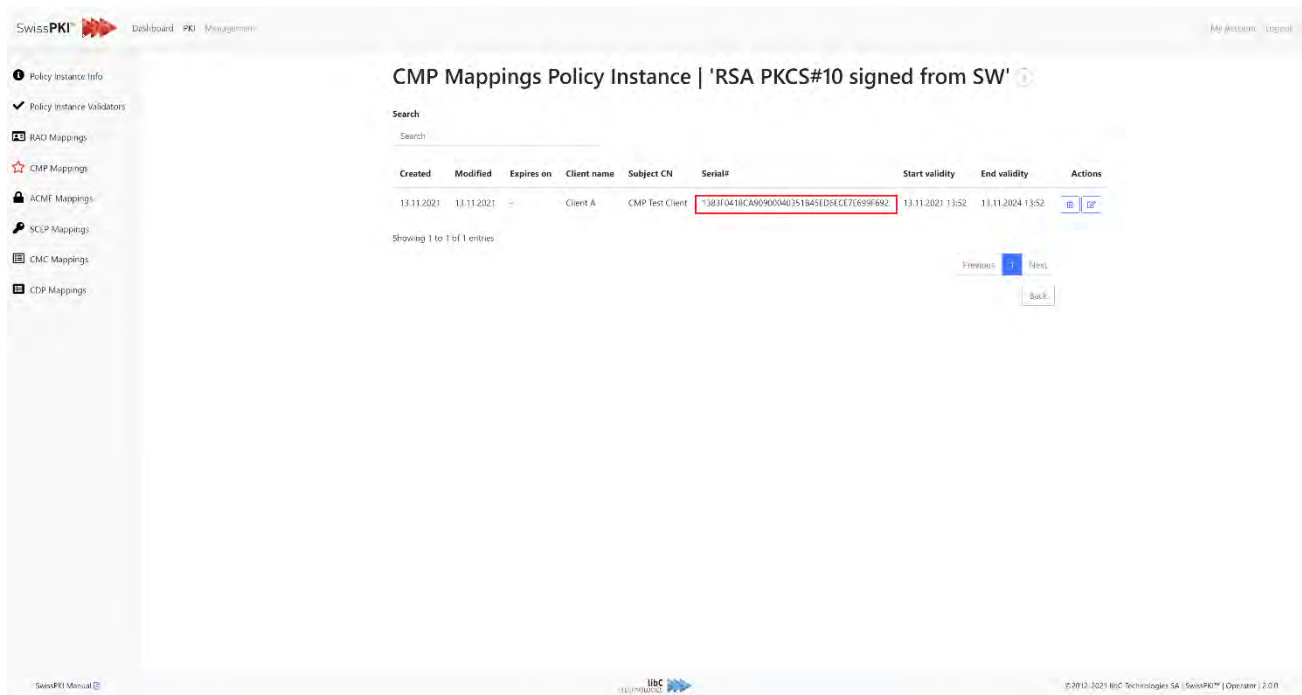
No data available in table

Showing 0 to 0 of 0 entries

[Previous](#) [Next](#)

### 12.3.2.1.4.1.1.3.2 CMP Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via CMC for the selected Client. This requires a CMC signing certificate in the Client setting, see [12.2.3.14 Client CMC Serial Number](#)



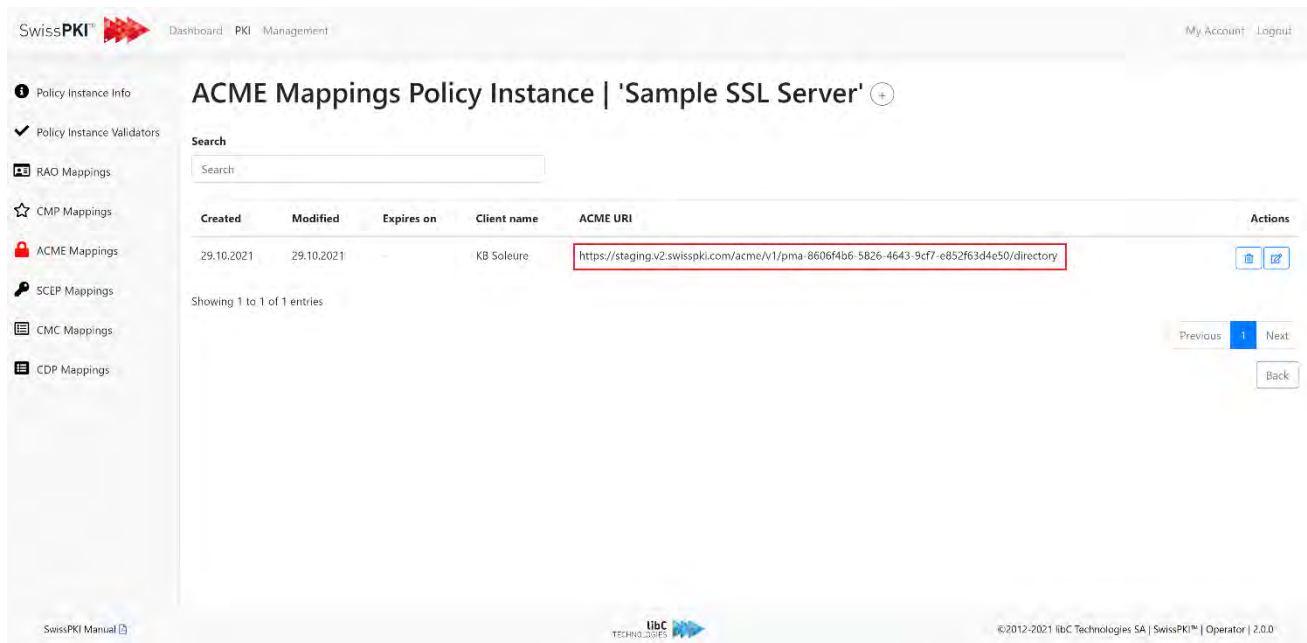
You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see <a href="#">12.2.5 Notifications Templates</a>
<b>Certificate Revocation Notification</b>	Template notification reference, see <a href="#">12.2.5 Notifications Templates</a>
<b>Certificate Recovery Notification</b>	Template notification reference, see <a href="#">12.2.5 Notifications Templates</a>
<b>Authorization Rule</b>	Rule definition reference, see <a href="#">12.2.4.2 Authorization Rules</a>
<b>Registration Rule</b>	Rule definition reference, see <a href="#">12.2.4.1 Registration Rules</a>
<b>Renewal Rule</b>	Rule definition reference, see <a href="#">12.2.4.3 Renewal Rules</a>
<b>Publish certificate</b>	Publishes the certificate via the associated Publisher instance(s), see <a href="#">12.3.2.7 Publisher</a>

<b>Enable for API Access</b>	n/a
<b>Generate revocation code for use in Self Service</b>	When enabled, the issuing CA generates a revocation code delivered to the recipient in the form of an HTTP link. The revocation code can then be used by the recipient to revoke the certificate via the self-service page.
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See <i>12.3.2.1.4.1.1.2 Policy Instance Validators</i>

### 12.3.2.1.4.1.1.3.3 ACME Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via an ACME for the selected Client. An ACME URL is generated for the Client.



You configure the behavior rules for the Policy Instance.

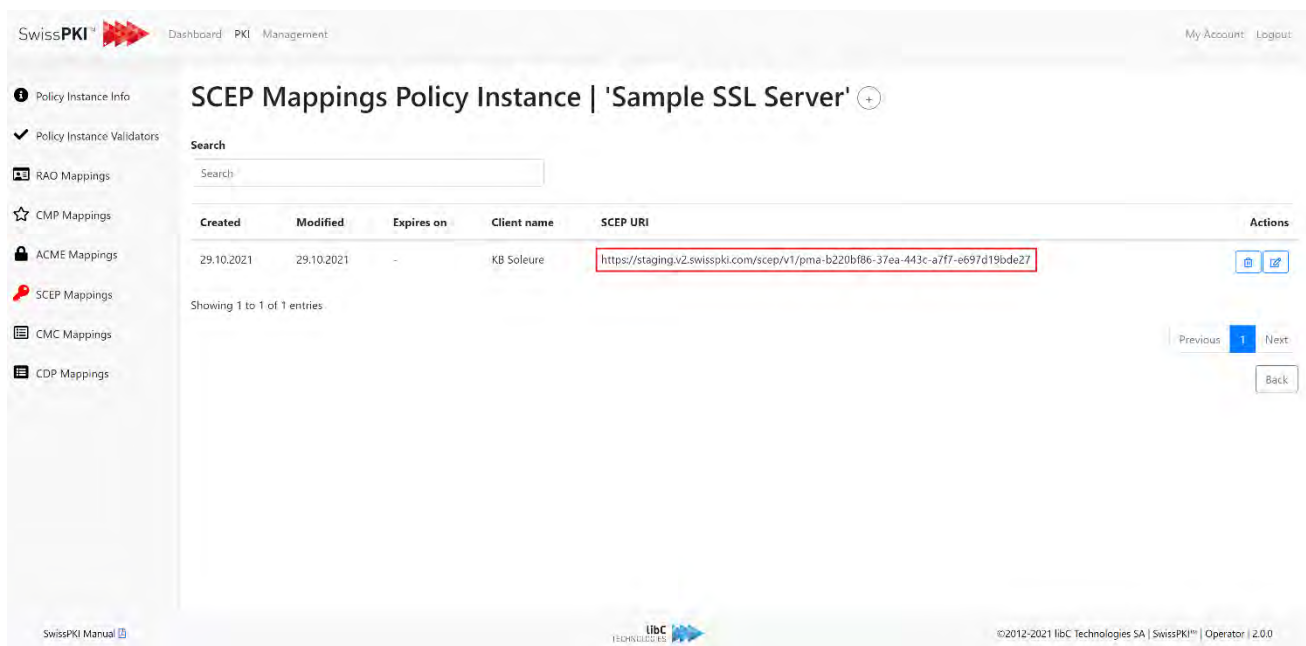
Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Recovery Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	n/a
<b>Renewal Rule</b>	n/a
<b>Generate revocation code for use in Self Service</b>	n/a

<b>Publish certificate</b>	Publishes the certificate via the associated Publisher instance(s), see <i>12.3.2.7 Publisher</i>
<b>ACME Token validity</b>	Defines the validity of the generated DNS ACME token. After expiration of the ACME, the Client must register a new DNS token.
<b>Enable for API Access</b>	n/a
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See <i>12.3.2.1.4.1.1.2 Policy Instance Validators</i>

Generated ACME tokens are listed for the Client in *12.2.3.8 ACME Tokens*.

### 12.3.2.1.4.1.1.3.4 SCEP Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via SCEP for the selected Client. An SCEP URL is generated for the Client.



The screenshot shows the 'SCEP Mappings Policy Instance | 'Sample SSL Server'' page in the SwissPKI management console. A table lists the mappings with columns for Created, Modified, Expires on, Client name, and SCEP URI. The SCEP URI for the client 'KB Soleure' is highlighted with a red box: `https://staging.v2.swisspki.com/scep/v1/pma-b220bf86-37ea-443c-a7ff-e697d19bde27`. The interface also includes a search bar, navigation buttons (Previous, Next, Back), and a footer with copyright information.

You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Recovery Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	n/a
<b>Renewal Rule</b>	Rule definition reference, see 12.2.4.1 Registration Rules
<b>Publish certificate</b>	Publishes the certificate via the associated Publisher instance(s), see 12.3.2.7 Publisher
<b>Enable for API Access</b>	n/a



<b>Generate revocation code for use in Self Service</b>	When enabled, the issuing CA generates a revocation code delivered to the recipient in the form of an HTTP link. The revocation code can then be used by the recipient to revoke the certificate via the self-service page.
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See <i>12.3.2.1.4.1.1.2 Policy Instance Validators</i>

A SCEP PIN valid for a period of 7 days is generated/updated for the Policy Mapping

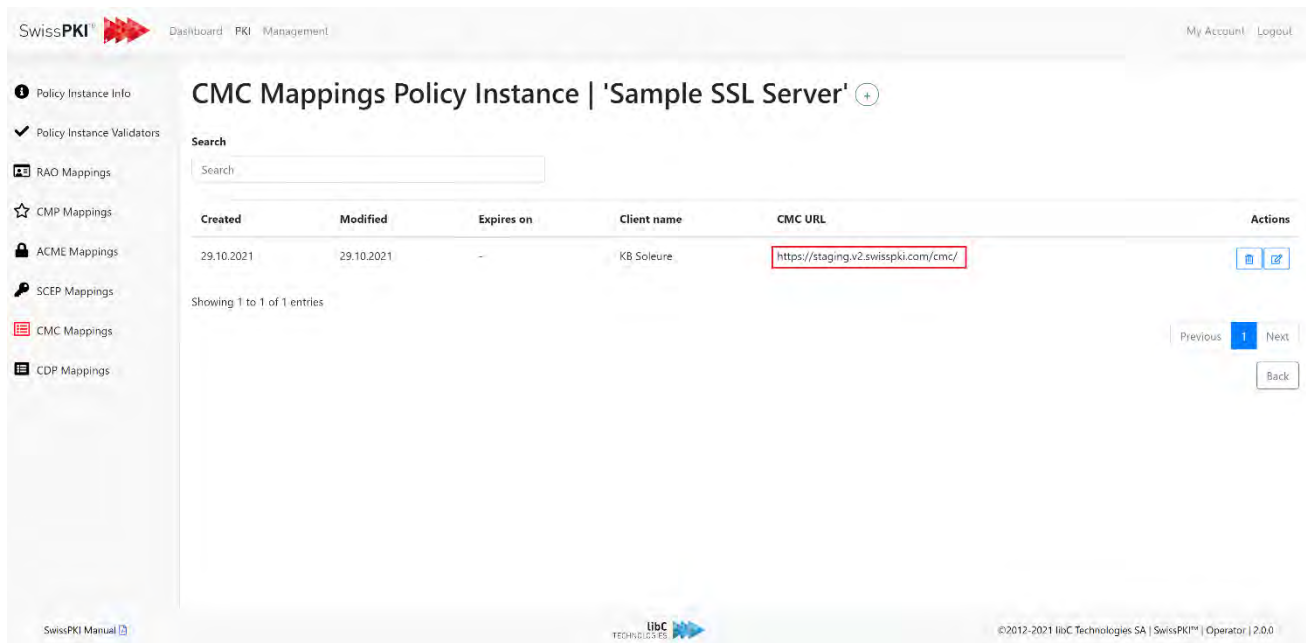
## SCEP Pin

Active	true
Valid until	29.11.2021 00:00
Password	ydWH6cmk

Generated SCEP PINs are listed for the Client in *12.2.3.9 SCEP*.

### 12.3.2.1.4.1.1.3.5 CMC Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via CMC for the selected Client.



You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Recovery Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	n/a
<b>Renewal Rule</b>	Rule definition reference, see 12.2.4.1 Registration Rules
<b>Publish certificate</b>	Publishes the certificate via the associated Publisher instance(s), see 12.3.2.7 Publisher
<b>Enable for API Access</b>	n/a

<b>Generate revocation code for use in Self Service</b>	When enabled, the issuing CA generates a revocation code delivered to the recipient in the form of an HTTP link. The revocation code can then be used by the recipient to revoke the certificate via the self-service page.
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See <i>12.3.2.1.4.1.1.2 Policy Instance Validators</i>

CMC Policy Instances must be configured at Client, see *12.2.3 Clients* (CMC Account) and *12.2.3.14 Client CMC Serial Number*

### 12.3.2.1.4.1.1.3.6 Microsoft Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via Microsoft CES/CEP for the selected Client.

You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Recovery Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	n/a
<b>Renewal Rule</b>	n/a
<b>Publish certificate</b>	n/a
<b>Enable for API Access</b>	n/a
<b>Generate revocation code for use in Self Service</b>	n/a
<b>Expiration date</b>	Set an expiration date for the Policy Instance. If unset, it is always valid
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See 12.3.2.1.4.1.1.2 Policy Instance Validators

Microsoft Policy Instances must be configured at MSCA CES/CEP level, see 12.3.2.6.3 *Microsoft Policies*

### 12.3.2.1.4.1.1.3.7 SCION Mappings

Enable the Policy Instance *P* for Client *C*. This exposes the Policy Instance via SCION Adapter for the selected Client.

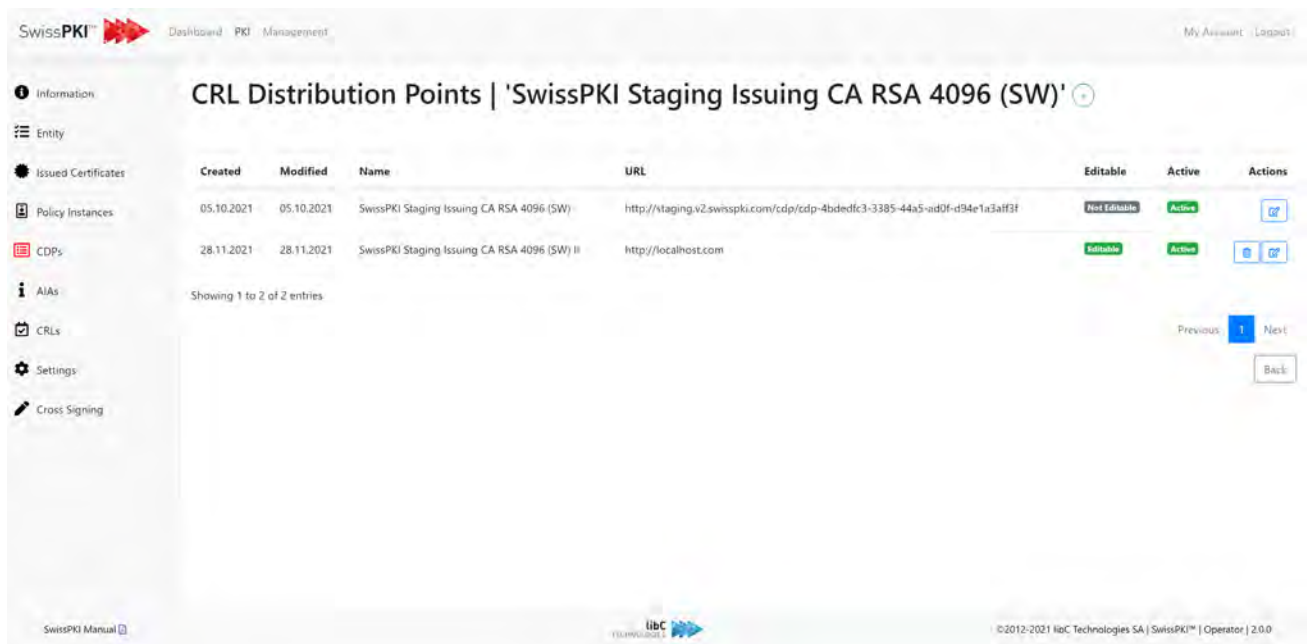
You configure the behavior rules for the Policy Instance.

Rule	Description
<b>Certificate Issuance Notification</b>	Template notification reference, see 12.2.5 Notifications Templates
<b>Certificate Revocation Notification</b>	n/a
<b>Certificate Recovery Notification</b>	n/a
<b>Authorization Rule</b>	Rule definition reference, see 12.2.4.2 Authorization Rules
<b>Registration Rule</b>	n/a
<b>Renewal Rule</b>	n/a
<b>Publish certificate</b>	n/a
<b>Enable for API Access</b>	n/a
<b>Generate revocation code for use in Self Service</b>	n/a
<b>Expiration date</b>	n/a
<b>External reference</b>	Reference to an external system/product identifier
<b>Partner reference</b>	Reference to an external system/product partner identifier
<b>Registration sources</b>	n/a
<b>Validators</b>	See 12.3.2.1.4.1.1.2 Policy Instance Validators

### 12.3.2.1.5 CRL Distribution Points

Applies to Certification Authorities of type **SwissPKI**.

Configure and manage the CDPs produced by the Certification Authority. The CDP URLs are included into the issued certificates when selected on the Policy Instance (see 12.3.2.1.4.1.1 *Policy Instance*). Additionally, the Certificate Policy Template must include a CDP extension (see 12.3.1.2.14 *CRL Distribution Point*)



Created	Modified	Name	URL	Editable	Active	Actions
05.10.2021	05.10.2021	SwissPKI Staging Issuing CA RSA 4096 (SW)	http://staging.v2.swisspki.com/cdp/cdp-4bdeffc3-3385-44a5-aid0f-d94e1a3aff3f	Not Editable	Active	
28.11.2021	28.11.2021	SwissPKI Staging Issuing CA RSA 4096 (SW) II	http://localhost.com	Editable	Active	

Showing 1 to 2 of 2 entries

Previous 1 Next

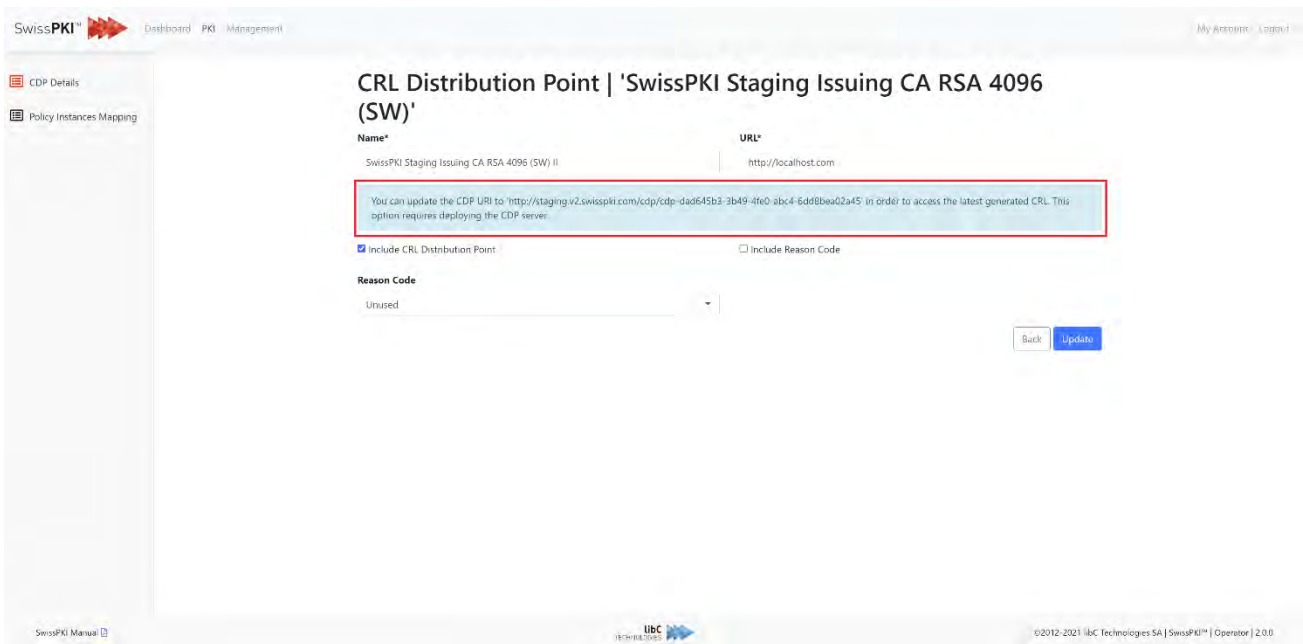
Back

Create a new CDP by clicking on the “+” sign:



Field	Description
<b>CDP logical name</b>	Logical name of the CDP
<b>Publication URL</b>	CDP publication end point, either one of HTTP or LDAP URL
<b>Include CRL Distribution Point</b>	Include CDP in CRL
<b>Include Reason Code</b>	Include Reason code in CDL
<b>Reason Code</b>	Select the revocation code (limited to 'unused')

Upon initial creation of the CDP, you have the possibility to use the generated CDP HTTP end point.

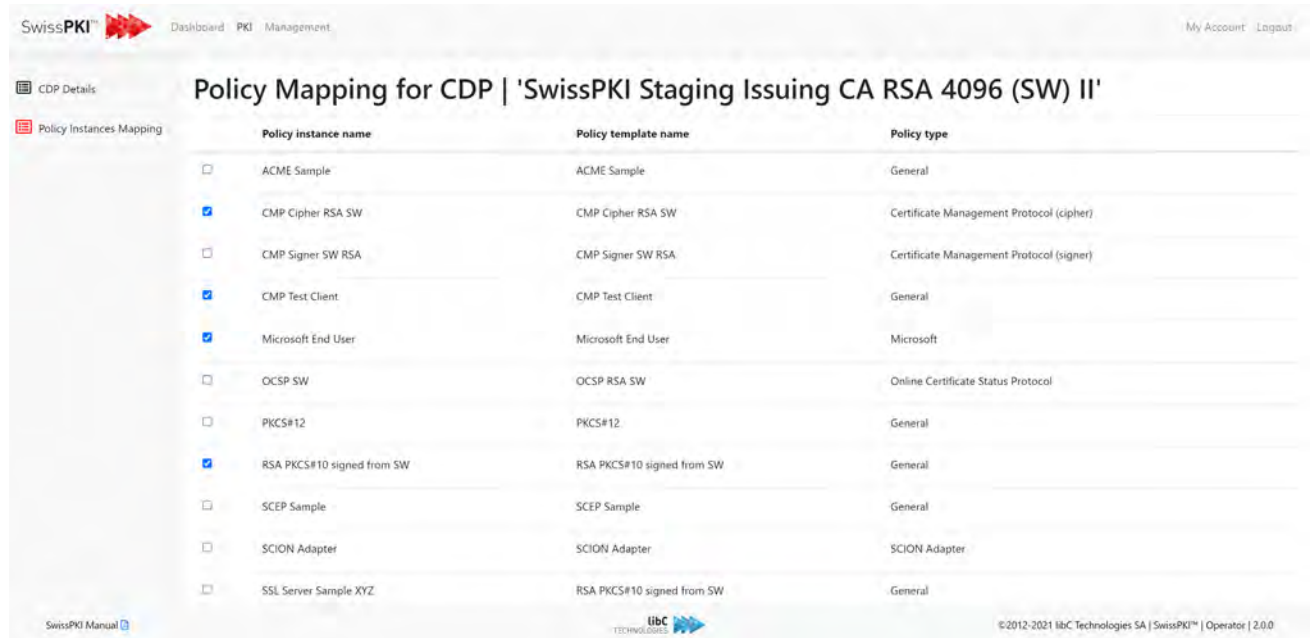


When used with the CDP Service (see 8.4.2 CRL Distribution Points (CDP)), the generated CDP end point will always serve the latest generated CRL. If you wish to use a CRL end point of defined on your own, then you will have to implement a script to copy the latest CRL to your defined end point or define rewrite rules on your reverse proxy to point to the generated CDP URI.

Deleting a CDP will either delete it if no CRL and no reference to the CDP exists or retire the CDP if at least one CRL is generated for the CDP.



You can directly edit the CDP assignment to Policy Instances (see 12.3.2.1.4.1.1 Policy Instance) from the Policy Instance Mapping by selecting which Policy Instances must use this CDP.



The screenshot shows the 'Policy Mapping for CDP | 'SwissPKI Staging Issuing CA RSA 4096 (SW) II'' interface. It features a table with columns for 'Policy instance name', 'Policy template name', and 'Policy type'. Several rows are checked, indicating they are assigned to the CDP.

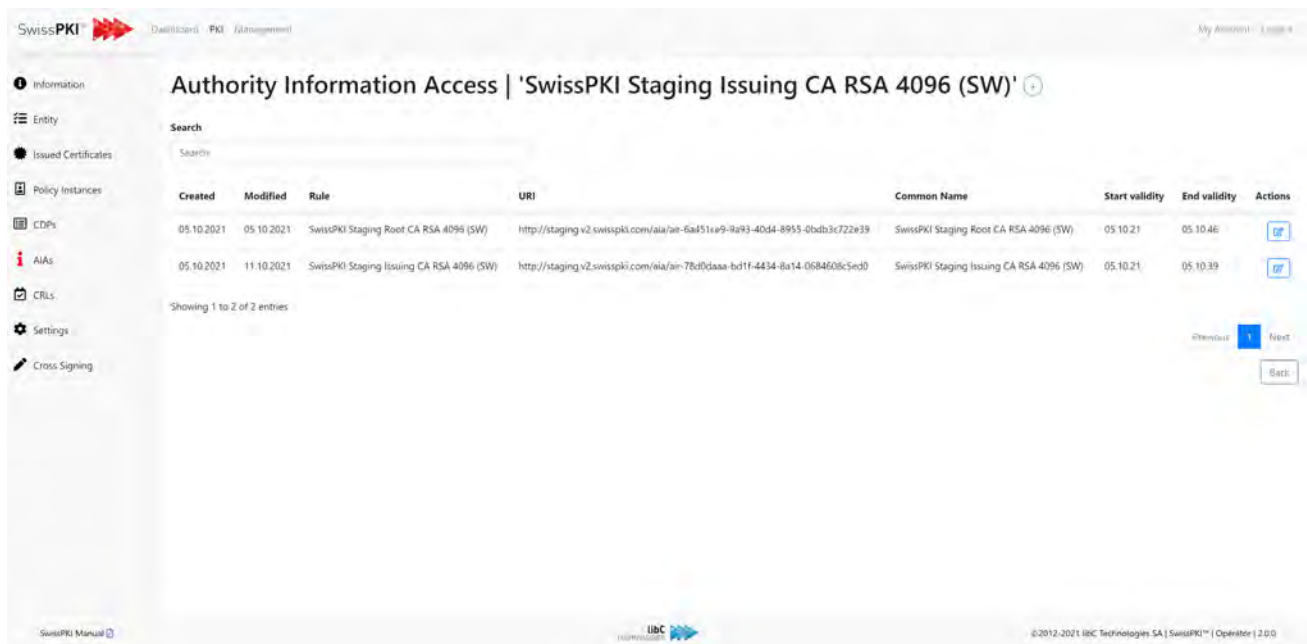
	Policy instance name	Policy template name	Policy type
<input type="checkbox"/>	ACME Sample	ACME Sample	General
<input checked="" type="checkbox"/>	CMP Cipher RSA SW	CMP Cipher RSA SW	Certificate Management Protocol (cipher)
<input type="checkbox"/>	CMP Signer SW RSA	CMP Signer SW RSA	Certificate Management Protocol (signer)
<input checked="" type="checkbox"/>	CMP Test Client	CMP Test Client	General
<input checked="" type="checkbox"/>	Microsoft End User	Microsoft End User	Microsoft
<input type="checkbox"/>	OCSP SW	OCSP RSA SW	Online Certificate Status Protocol
<input type="checkbox"/>	PKCS#12	PKCS#12	General
<input checked="" type="checkbox"/>	RSA PKCS#10 signed from SW	RSA PKCS#10 signed from SW	General
<input type="checkbox"/>	SCEP Sample	SCEP Sample	General
<input type="checkbox"/>	SCION Adapter	SCION Adapter	SCION Adapter
<input type="checkbox"/>	SSL Server Sample XYZ	RSA PKCS#10 signed from SW	General

Additional interface elements include 'SwissPKI Manual' and 'libC TECHNOLOGIES' logos, and a footer with copyright information: ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.3.2.1.6 Authority Information Access

Applies to Certification Authorities of type **SwissPKI**.

The Authority information access allows you to link the certificate with the AIA Service (see 8.4.3 *Authority Information Access (AIA)*) and the Certificate Policy Template '*caIssuer*' value (see 12.3.1.2.7 Authority Information Access) to automatically serve the file for the generated AIA URL. Optionally, you can copy the certificate to a static location and serve the file from your server.



The screenshot shows the 'Authority Information Access' configuration page for 'SwissPKI Staging Issuing CA RSA 4096 (SW)'. It features a search bar and a table with the following data:

Created	Modified	Rule	URI	Common Name	Start validity	End validity	Actions
05.10.2021	05.10.2021	SwissPKI Staging Root CA RSA 4096 (SW)	http://staging.v2.swisspki.com/aia/ae-6af51e49-9a93-40d4-8955-0bdb3c722e39	SwissPKI Staging Root CA RSA 4096 (SW)	05.10.21	05.10.46	<a href="#">+</a>
05.10.2021	11.10.2021	SwissPKI Staging Issuing CA RSA 4096 (SW)	http://staging.v2.swisspki.com/aia/aia-78d0daa-bd1f-4434-8a14-0684608c5ed0	SwissPKI Staging Issuing CA RSA 4096 (SW)	05.10.21	05.10.39	<a href="#">+</a>

Showing 1 to 2 of 2 entries

To create a new AIA <sup>19</sup> end point clicks on the “+” sign and select <sup>20</sup> a Certification Authority from the drop down.

<sup>19</sup> The SwissPKI AIA module is deployed and the *hosts.conf* contains the domain name of the deployed AIA module.

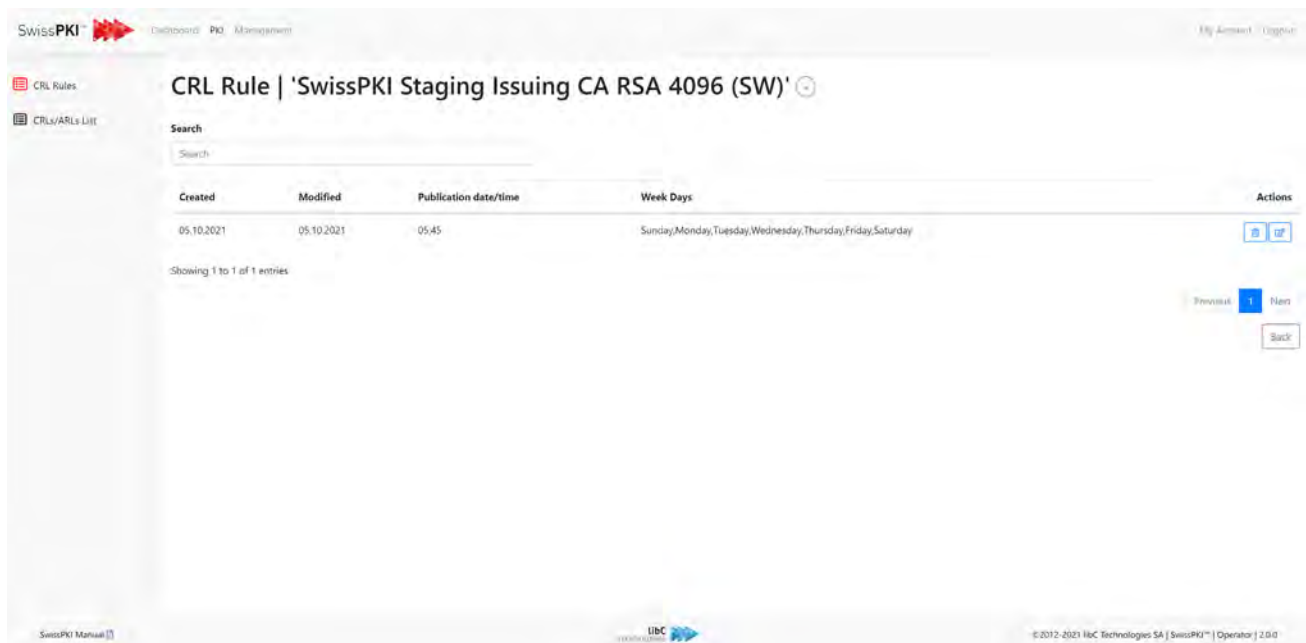
<sup>20</sup> If your CA is cross signed, you can switch the Authority Key when switching between CA certificates

### 12.3.2.1.7 Certificate Revocation Lists

Applies to Certification Authorities of type **SwissPKI**.

You can configure the automatic CRL generation schedules for the Certification Authority. For each schedule, you set a time and day. Each schedule is picked up by the Scheduler to generate CRLs/ARLs. Generated CRLs/ARLs include the CRL grace period set in the Certification Authority settings (see *Settings*).

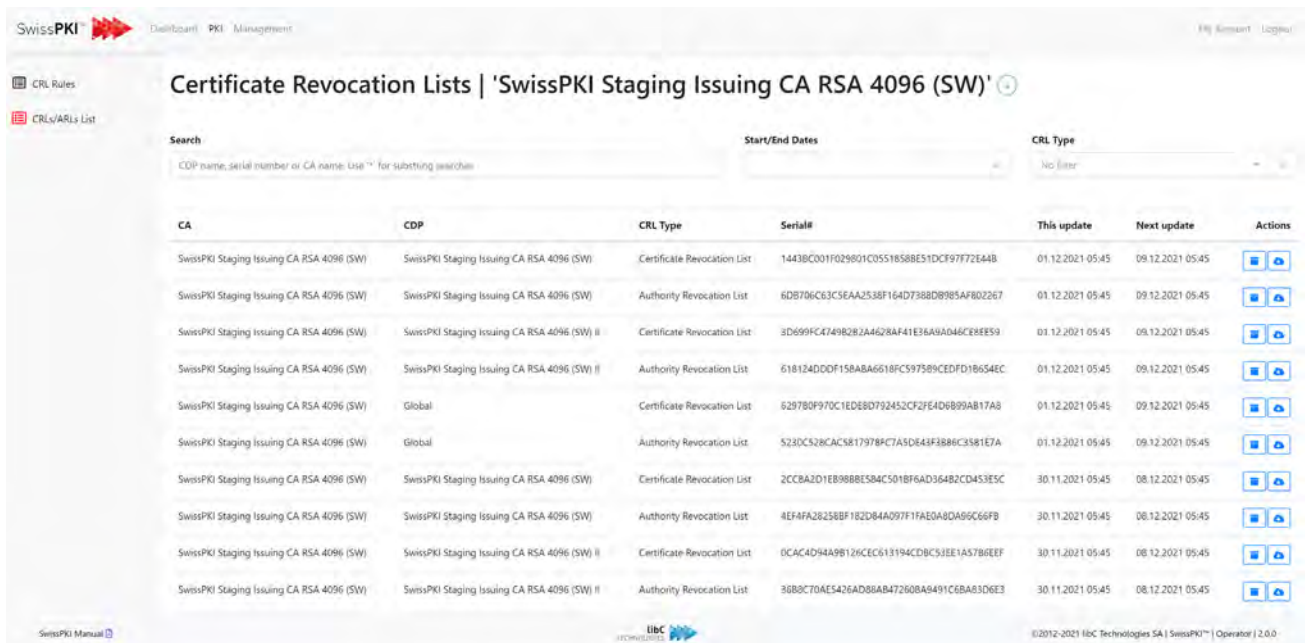
For each CDP, a CRL/ARL is generated using the automatic schedule as well as one global CRL and ARL.



Click on the “+” sign to a CRL Rule schedule:



Browse the issued CRL/ARL. You can download the generated CRL or optionally republish it by notifying the associated Publisher(s). The publish action is available when the Certification Authority is associated to at least one Publisher.



Click on the “+” sign to manually generate a CRL or Last CRL. If you have an offline CA, do not forget to manually generate the CRL for the Certification Authority before the expiration of the grace period.

### 12.3.2.1.8 Settings

Applies to Certification Authorities of type **SwissPKI** and **SwissSign**.

#### 12.3.2.1.8.1 SwissPKI

The Certification Authority general settings

Setting	Description
<b>Enforce unique public keys</b>	<p>The Certification Authority will reject certificate issuance for duplicate public keys.</p> <p>Note that this has an impact on the automatic certificate renewal rules if you enable unique public check (see <i>12.2.4.3 Renewal Rules</i>)</p>
<b>Generate CRL on every revocation</b>	Force a CRL generation upon every revocation.
<b>Include expired certificates in CRL</b>	<p>Include expired certificates in the CRL.</p> <p>The CRL includes the extension to mention that it contains expired certificates.</p> <p>The option is useful when the Certification Authority is used to issue signing certificates and no OCSP token is added to signed data structures except a reference to the CRL.</p>
<b>Produce Global CRL</b>	<p>Generate a global CRL and ARL.</p> <p>This generate a CRL and ARL containing the revoked certificates for the CA in addition of optional CDP end points. If no CDP end point is defined, then the global CRL/ARL must be published manually for the CDP in the issued certificates to be retrieved.</p>
<b>Produce ARLs</b>	<p>For each CDP produce an ARL in addition to the CDP CRL.</p> <p><b>Important:</b> to produce a unique CRL per CA linked to a CDP generated URL, you create one single CDP per CA. This will generate a CDP end point serving the unique CRL for the CA.</p>
<b>This instance is a Public Trust Certification Authority</b>	Indicates if the Certification Authority is used for Public or Private Trust. This item is not editable.

	The value is defined when creating the CA of type <i>SWISSPKI</i> .
<b>CRL grace period</b>	The TTL of the issued CRL/ARL in dd/hh/mm/ss



SwissPKI™  Dashboard PKI Management My Account Logout

**Settings | 'Sample CA'**

- Enforce unique public keys
- Generate CRL on every revocation
- Include expired certificates in CRL
- Produce a Global CRL
- Produce ARLs
- This Certification Authority is CAB Public Trust

**CRL grace period**

<b>Days*</b>	<b>Hours*</b>	<b>Minutes*</b>	<b>Seconds*</b>
<input type="text" value="8"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

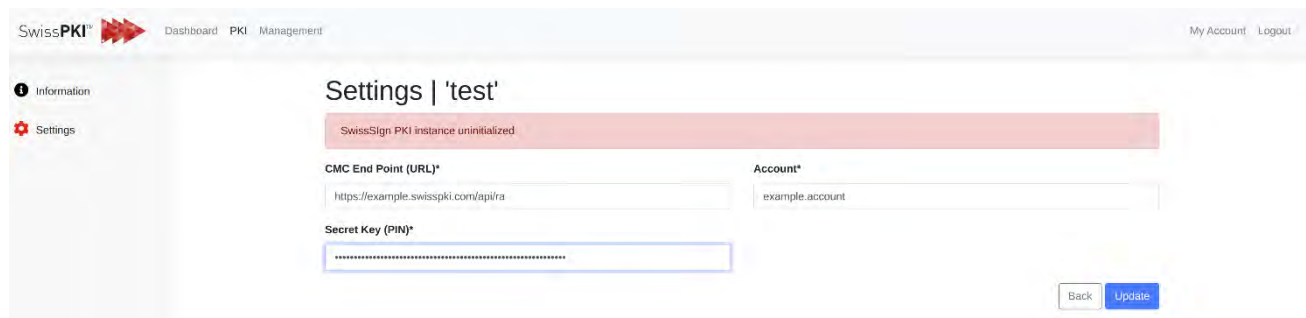
User manual  libC TECHNOLOGIES  ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0


### 12.3.2.1.8.2 SwissSign

The Certification Authority general settings.

A Certification Authority of type ‘**SwissSign**’ enables you to issue Public Trust certificates to your end users and systems. Certification requests are forwarded to the SwissSign Certification Authority. To enable this option, you need to register for an account <sup>21</sup> and certificate products with SwissSign.

Setting	Description
<b>SwissSign End Point URL</b>	Select the production or staging registration URL
<b>Account</b>	The account name (this information is delivered by SwissSign)
<b>Secret Key</b>	The account secret key (this information is delivered by SwissSign).



SwissPKI  Dashboard PKI Management My Account Logout

Information  
Settings

## Settings | 'test'

SwissSign PKI instance uninitialized

CMC End Point (URL)\*

Account\*

Secret Key (PIN)\*

[Back](#) [Update](#)

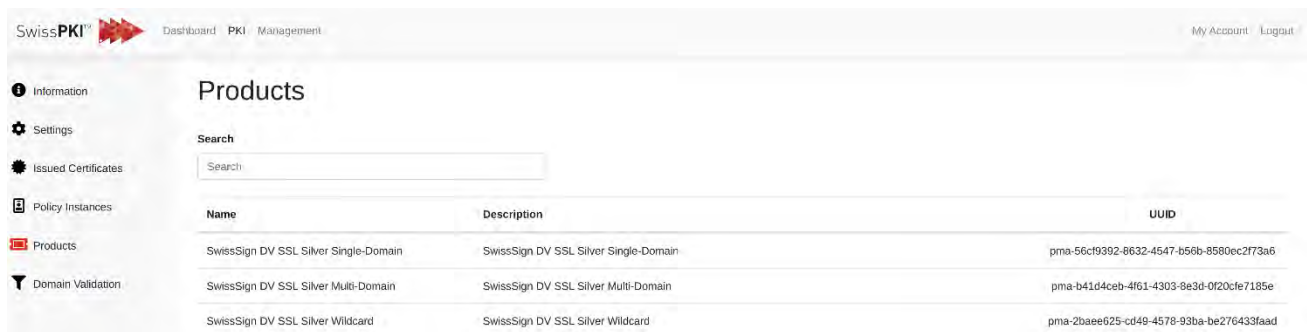
<sup>21</sup> Connection to the SwissSign Certification Authority is done using an API .

### 12.3.2.1.8.3 Products

Applies to Certification Authorities of type ‘SwissSign.’

This tab allows you to view all the available SwissSign products for which you can issue certificates.

Setting	Description
<b>Name</b>	SwissSign product name
<b>Description</b>	Description of the product
<b>UUID</b>	The SwissSign product UUID. This UUID should be entered in the policy template under the Swiss Sign product name section. View section 12.3.1.2.30 Swiss Sign Product Name.



The screenshot shows the 'Products' management page in the SwissPKI interface. The page title is 'Products' and it includes a search bar. A table lists three products with their names, descriptions, and UUIDs.

Name	Description	UUID
SwissSign DV SSL Silver Single-Domain	SwissSign DV SSL Silver Single-Domain	pma-56c9392-8632-4547-b56b-8580ec2f73a6
SwissSign DV SSL Silver Multi-Domain	SwissSign DV SSL Silver Multi-Domain	pma-b41d4ceb-4f61-4303-8e3d-0f20cfe7185e
SwissSign DV SSL Silver Wildcard	SwissSign DV SSL Silver Wildcard	pma-2baee625-cd49-4578-93ba-be276433faad



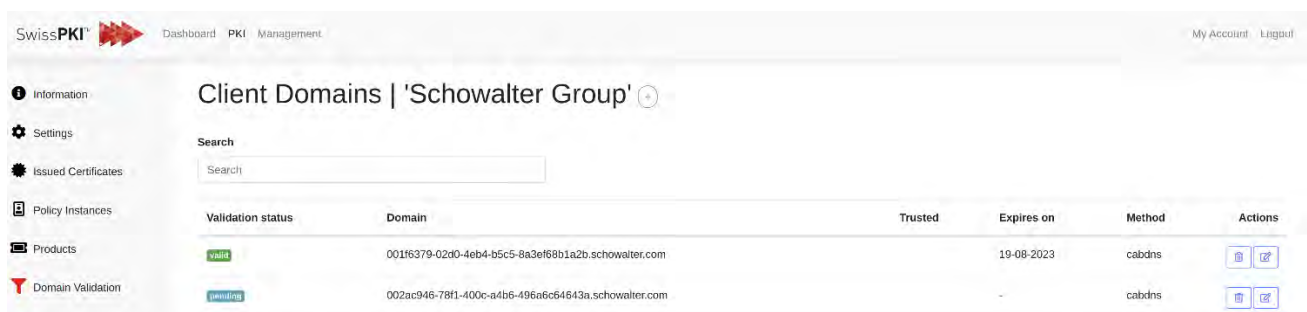
#### 12.3.2.1.8.4 Domain Validation





Applies to Certification Authorities of type ‘SwissSign.’

For policy templates that have a “DNS owner rule,” domain names must be validated during issuance of the certificate. To make it easier for the client, he has the option to pre-validate a domain (usually the client validates his top-level domain) so that he can issue certificates for that domain + subdomains without the need to validate every certificate request individually.

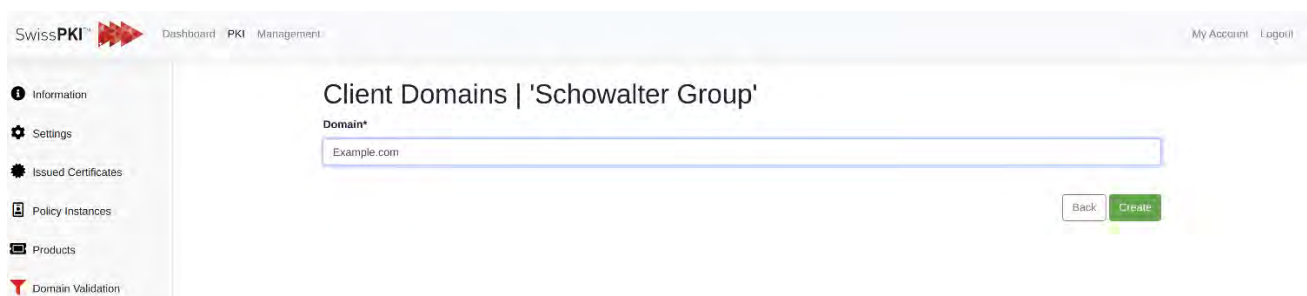
Note that you must have the ‘SwissSign domain pre-validation’ permissions to create, read, update, or delete pre-validated domains.

Start by selecting the client for which you want to manage the pre-validated domains. You can then view all the pre-validated domains. You have the option to delete or edit an existing domain or create a new one. IM

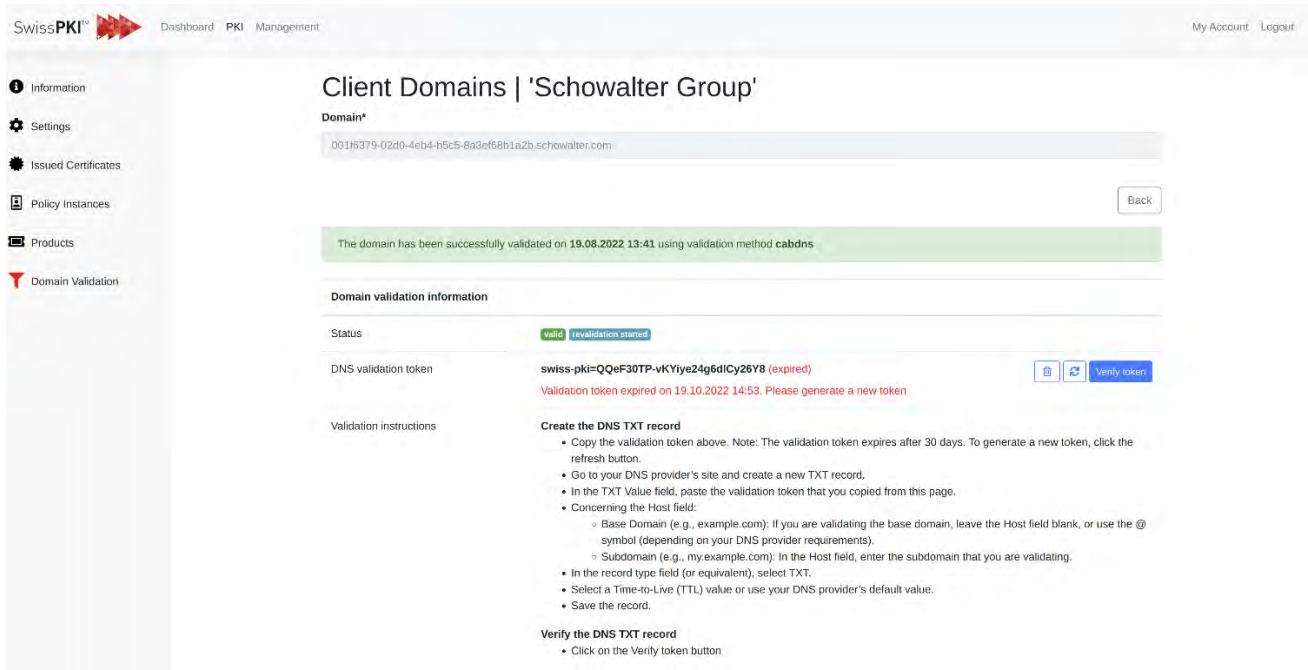


Validation status	Domain	Trusted	Expires on	Method	Actions
valid	001f6379-02d0-4eb4-b5c5-8a3ef68b1a2b.schowalter.com		19-08-2023	cabdns	 
pending	002ac946-78f1-400c-a4b6-496a6c64643a.schowalter.com		-	cabdns	 

By clicking on the “Add” button the operator can enter a new pre-validated domain for the selected CA and client. Simply enter the name of the domain which should be pre-validated.



After editing a domain or creating a new one, you will be redirected to a page where you can see more information on the domain. You now have the option to delete, generate or verify a validation token. Validation instructions are displayed on screen. The token is valid for 30 days. After this period, a new token must be generated.



SwissPKI Dashboard PKI Management My Account Logout

### Client Domains | 'Schowalter Group'

Domain\*

00116379-02d0-4eb4-b5c5-8a3ef68b1a2b.schowalter.com

The domain has been successfully validated on 19.08.2022 13:41 using validation method **cabdns**

Domain validation information

Status **valid** **revalidation started**

DNS validation token **swiss-pki=QeF30TP-vKYIye24g6dICy26Y8 (expired)**  
Validation token expired on 19.10.2022 14:53. Please generate a new token

Validation instructions

**Create the DNS TXT record**

- Copy the validation token above. Note: The validation token expires after 30 days. To generate a new token, click the refresh button.
- Go to your DNS provider's site and create a new TXT record.
- In the TXT Value field, paste the validation token that you copied from this page.
- Concerning the Host field:
  - Base Domain (e.g., example.com): If you are validating the base domain, leave the Host field blank, or use the @ symbol (depending on your DNS provider requirements).
  - Subdomain (e.g., my.example.com): In the Host field, enter the subdomain that you are validating.
- In the record type field (or equivalent), select TXT.
- Select a Time-to-Live (TTL) value or use your DNS provider's default value.
- Save the record.

**Verify the DNS TXT record**

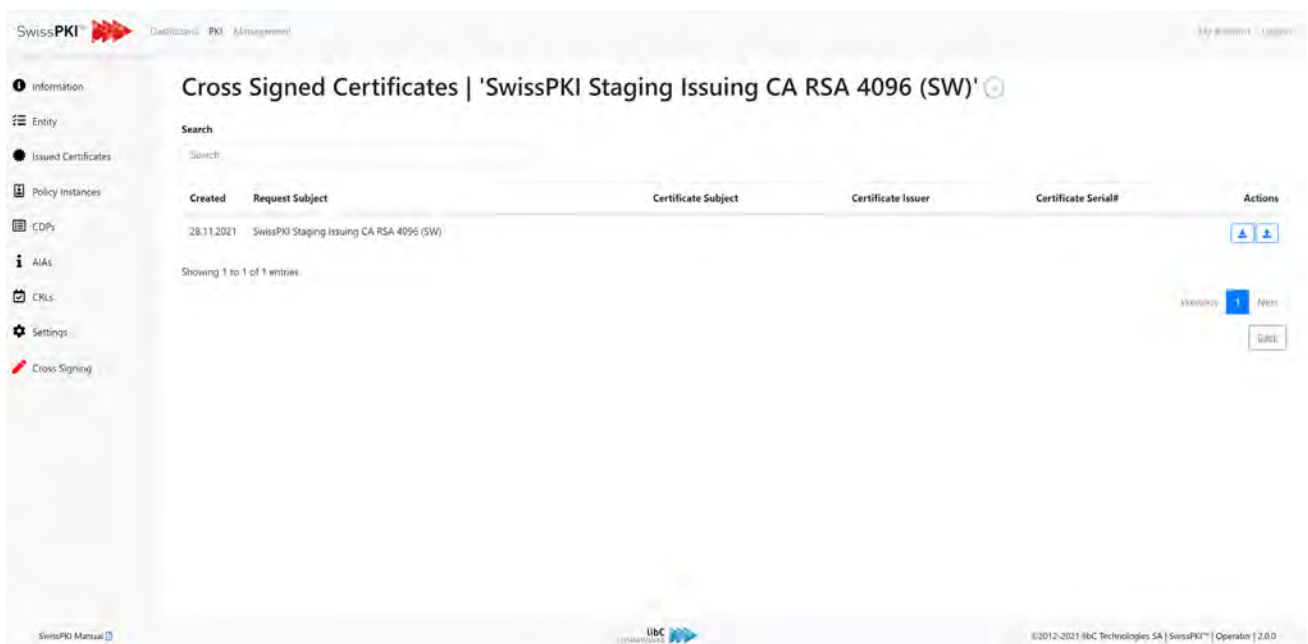
- Click on the Verify token button

### 12.3.2.1.9 Cross Signing

Applies to Certification Authorities of type **SwissPKI**.

You cross sign your Certification Authority by generating a PKCS#10 using the private key. The PKCS#10 signature algorithm is the one defined for the key generation of the Certification Authority.

For each cross signed request, an entry in the table is generated.





SwissPKI Dashboard PKI Management My Account Logout

### Cross Signed Certificates | 'SwissPKI Staging Issuing CA RSA 4096 (SW)'

Search

Created Request Subject Certificate Subject Certificate Issuer Certificate Serial# Actions

Created	Request Subject	Certificate Subject	Certificate Issuer	Certificate Serial#	Actions
28.11.2021	SwissPKI Staging Issuing CA RSA 4096 (SW)				 

Showing 1 to 1 of 1 entries

1/1

SwissPKI Manual

libC TECHNOLOGIES

©2012-2021 libC Technologies SA | SwissPKI™ | Operation | 2.0.0

Download the PKCS#10 and submit it for cross signing. The Issuing Certification Authority must deliver a certificate encoded in PKCS#7. Upload the certificate obtained from the Issuing Certification Authority for the pending request.

Once the issued cross signed certificate is uploaded, you can configure the Authority Key of the Certification Authority in the 'Entity' settings (see *12.3.2.1.2 Entity*).

#### **12.3.2.1.10 Air Gaped CA**

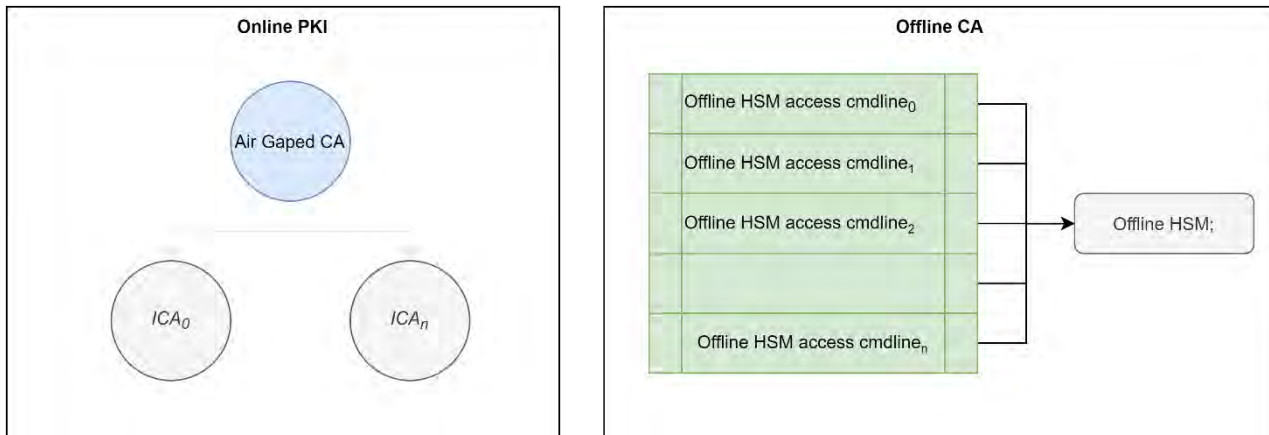
In the online mode, the Root CA instance constantly has access to the signing keys stored on an active partition on the HSM when the CA instance is effectively in enabled 'state.' This allows for unattended signing processes when the CA needs to e.g., sign new CRLs. The Root CA is constantly available to process signing requests.

In the offline mode, the Root CA process is deactivated and its HSM partition is offline. The CA therefore is not operational all the time. Before any signing process can be executed, authorized personnel need to activate the HSM partition and the CA instance. After the signing process has completed, both the HSM partition and the CA will be deactivated again. This ensures, that signing operations can only be executed under attendance of authorized personnel and this way the risk of unintended signing operations is strongly reduced.

Today, the *Network and Certificate System Security Requirements* which are part of the general CA/B *Baseline Requirements* for publicly trusted CAs require that TSPs maintain Root CA Systems in either a High Security Zone and in an offline state or in an air-gapped manner, separated from all other networks [Network and Certificate System Security Requirements, Section 1c.]

If the CA operating environment meets the requirements for a so-called High Security Zone, the CA must be operated as an offline root CA in accordance with the currently applicable regulations.

For this purpose, SwissPKI includes an 'Air Gaped' CA type. You run the Offline CA instance in the secured operating environment and keep it as part of the safe, well understood, and trained maintenance processes applicable for the standard deployment. Therefore, only the processes that require access to the CA's private key used to sign certificates and CRLs are moved to the bank safe. The Air Gaped CA instance and all its surrounding modules still live in the TSP data center and run in an always-on manner.



This concept combines the high security of an Air Gaped CA having the key material in the bank safe (Offline CA) with the retention of normal application maintenance in the data center. An audited CA subject to the CA/B public trust is subject to ongoing adjustments to the rules adopted in the forum. This would mean that for every access to the Root CA keys at the bank safe, the application would first have to be updated to the latest version.

With the implementation of this concept, the Signer module is only subject to updates if there are changes in the format or processes relating to the exchanged data, or if the requirements for the signature processes themselves change.

### 12.3.2.1.10.1 Supported Use Cases

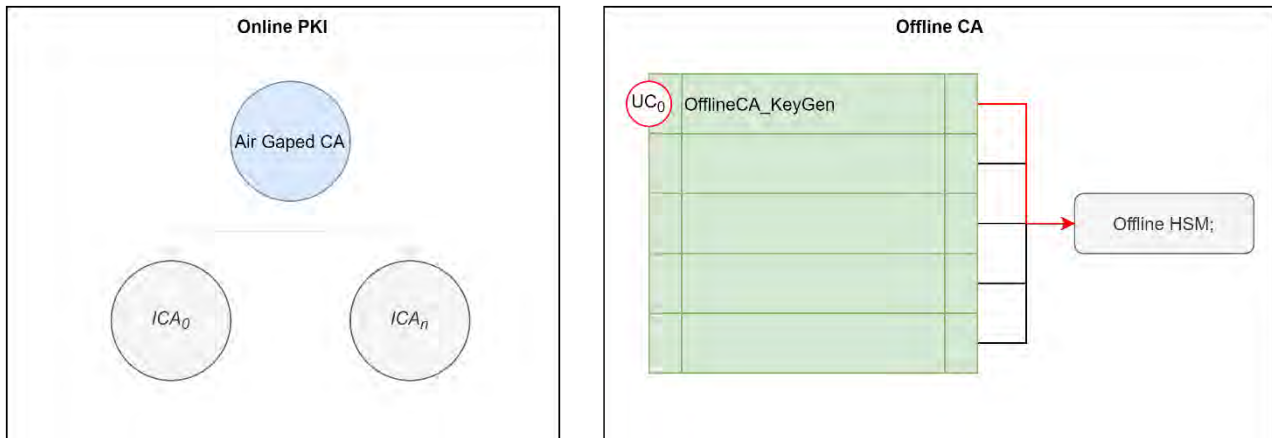
A SwissPKI Air Gaped CA instance supports the following use cases

ID	Use Case	Description
0	Offline CA key generation	Preparation of the Offline CA environment and issuance of the Root CA private key pairs following defined TSP procedures
1	Air Gaped CA initialization	<p>Creation of a SwissPKI of type 'Air Gapped' based on the standard certificate policy templates and generation of the Air Gaped sealing key pair.</p> <p>Production of the Air Gapped signed Offline CA order using the sealing key pair.</p> <p>Issuance of the Offline CA certificate using the Air Gaped order. This step usually involves a ceremony protocol.</p> <p>Finalization of the pending Air Gaped request with the result produced by the Offline CA.</p>
2	Air Gaped CA CRL	Generation of CRL/ARL on the Air Gaped CA instance.

		<p>Production of the Air Gapped signed Offline CA order using the sealing key pair.</p> <p>Issuance of the Offline CA CRL/ARL using the Air Gaped order. This step usually involves a ceremony protocol.</p> <p>Finalization of the pending Air Gaped request with the result produced by the Offline CA .</p>
3	Air Gaped CSR	<p>Generation of a CSR on the Air Gaped CA instance.</p> <p>Production of the Air Gapped signed Offline CA order using the sealing key pair.</p> <p>Issuance of the Offline CA CRL/ARL using the Air Gaped order. This step usually involves a ceremony protocol.</p> <p>Finalization of the pending Air Gaped request with the result produced by the Offline CA .</p>
4	Air Gaped ICA	<p>Generation of a Sub CA request on the Air Gaped CA instance.</p> <p>Production of the Air Gapped signed Offline CA order using the sealing key pair.</p> <p>Issuance of the Offline CA CRL/ARL using the Air Gaped order. This step usually involves a ceremony protocol.</p> <p>Finalization of the pending Air Gaped request with the result produced by the Offline CA .</p>

#### 12.3.2.1.10.2 Generating Offline CA Key Pair

Generate an Offline CA key pair in a controlled environment. The process description it out of scope of this user manual.



SwissPKI provides an optional command line tool to generate key pairs for an Offline CA if you do not plan to use the standard key generation tools provided by the HSM. Supported HSMs are Primus, LunaSA, Kryptus and ARCA.

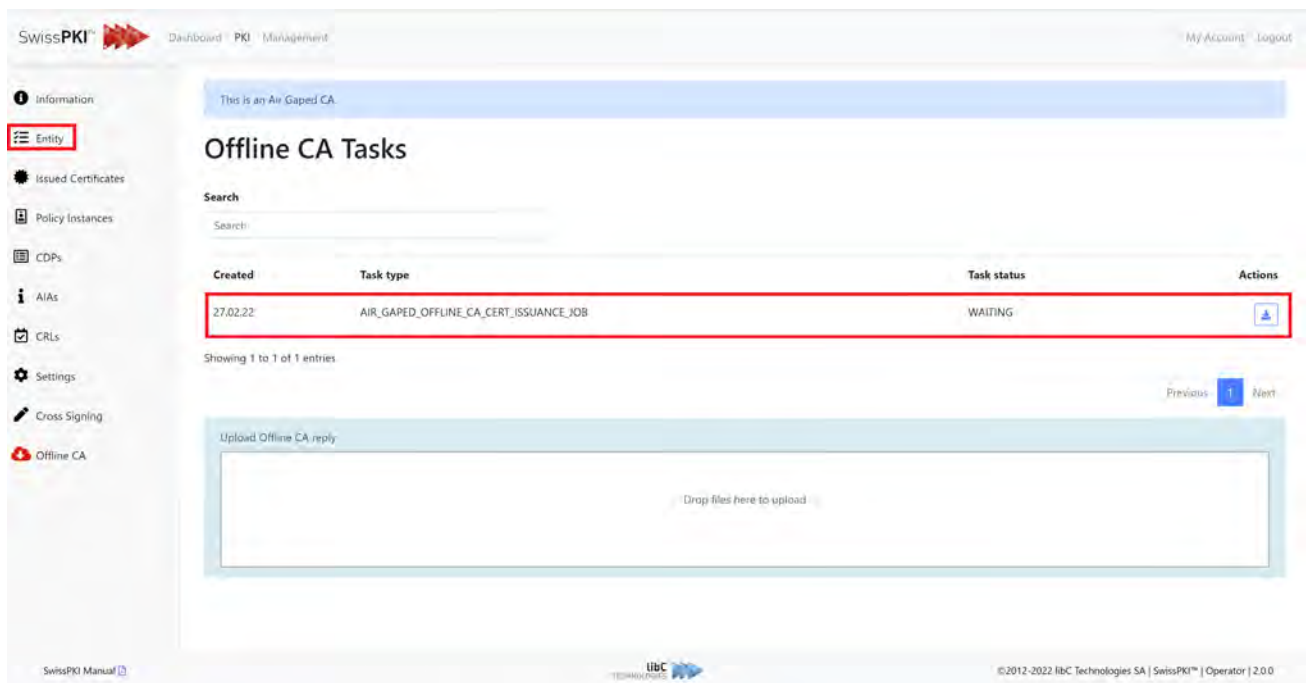
```
usage: OfflineCA_KeyGen
  -help          Generate a key pair on the selected HSM partition
  -keyType <algo>  rsa_2048, rsa_3072, rsa_4096, rsa_8192, ec_secp224k1,
                   ec_secp224r1, ec_secp256k1, ec_secp256r1,
                   ec_secp384r1, ec_secp521r1, ec_x962_p239v1,
                   ec_x962_p239v2, ec_x962_p239v3, ec_brainpool224r1,
                   ec_brainpool256r1, ec_brainpool320r1,
                   ec_brainpool384r1, ec_brainpool512r1
  -provider <type> Luna, Primus, ARCA or Kryptus
  -version       print the version information and exit
```

**Note:** When generating a key pair, you must ensure that the CKA\_LABEL of both private and public key objects on the HSM partition are set to a value of your choice. The subsequent Offline CA operations for signing Air Gaped requests reference the key pairs using the CKA\_LABEL when accessing the private/public key objects.


### 12.3.2.1.10.3 Air Gaped CA and Offline CA certificate issuance

Creating a new CA of type 'SwissPKI Air Gaped' will produce a request for the Offline CA. This process involves:

1. Creating CA type of type SwissPKI Air Gapped which applies only to offline Root CA
2. Generate Air Gapped CA based on certificate policy template. Note that Air Gaped CA types only support sealing key pairs generated (or referenced) on HSMs of type Primus, LunaSA, Kryptus and ARCA
3. Issuance of the sealing key pair used as transport authentication key to the Offline CA
  - o The key pair relies on the same key generation parameters as given by the policy template of the Offline CA instance.
  - o Key pair is generated (or referenced) on the HSM in line with the policy instance definition.
  - o The usage period of the transport sealing key is the same as the Offline CA signing key pair.
4. Produce a signed request for the Offline CA creation (the generated request file name is labelled *offline\_rca\_AirGapedOfflineCACertificateIssuanceJob.p7m*). Record the fingerprints of the seal key and export the Air Gaped certificate (available from the Entity menu)



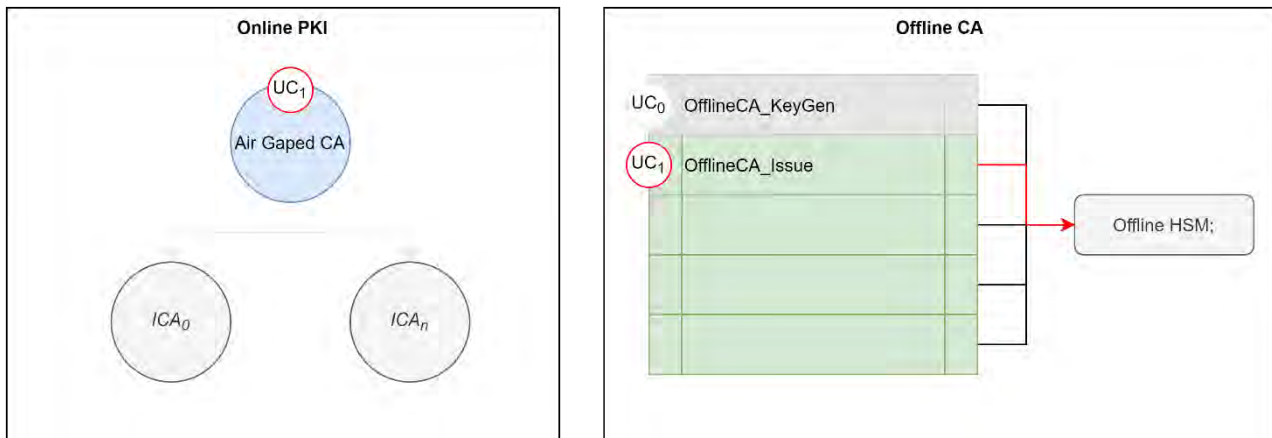
The screenshot shows the SwissPKI web interface. The left sidebar has a menu with 'Entity' highlighted. The main content area is titled 'Offline CA Tasks' and contains a table with the following data:

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	WAITING	

Below the table is an 'Upload Offline CA.reply' section with a 'Drop files here to upload' area.

5. Transfer the generated request *offline\_rca\_AirGapedOfflineCACertificateIssuanceJob.p7m* and Air Gaped certificate to the Offline CA

6. Issue the Offline CA certificate using the Air Gaped request files and Offline CA private key




7. Execute the `OfflineCA_Issue` command. This will issue the Offline CA certificate, an Air Gaped reply file and link the sealing key with the Offline CA key pair on the HSM partition. Record the generated certificate fingerprints and bring the produced Air Gaped reply file to the Online system to finalize the procedure.

**Note:** the generated CA certificate start/end validity is adjusted to the signing time during the Offline CA signing procedure. The local time on the signing machine must be set or synchronized with a NTP service. The validity of the issued certificate is provided by the Air Gaped request based on the certificate policy template settings.


```
usage: OfflineCA_Issue
-airGapedCACertificate <file> Air Gaped CA certificate in DER
-help Issue the Offline CA certificate based on
the Air Gaped request and the selected
private key.
-outAirGapedReply <file> Write Air Gaped reply to this output file
-outCertificate <file> Write certificate to this output file
-privateKeyLabel <file> Offline CA private key label
-provider <type> Luna, Primus, ARCA or Kryptus
-request <file> Air Gaped Certificate Issuance request in
DER
-version print the version information and exit
```













- Complete the Air Gaped certificate issuance by importing the generated Offline CA reply file. Record the fingerprints of the Offline CA certificate (available from the Entity menu). When processed, the Air Gaped request in WAITING status switches to SUCCESS.

27.02.22 AIR\_GAPED\_OFFLINE\_CA\_CERT\_ISSUANCE\_JOB SUCCESS 

---

SwissPKI™  Dashboard PKI Management My Account Logout


-  Information
-  Entity
-  Issued Certificates
-  Policy instances
-  CDPs
-  AIAs
-  CRLs
-  Settings
-  Cross Signing
-  Offline CA

This is an Air Gaped CA.

### Offline CA Tasks

Search

Search


Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	PENDING	

Showing 1 to 1 of 1 entries

Previous 1 Next

Upload Offline CA reply

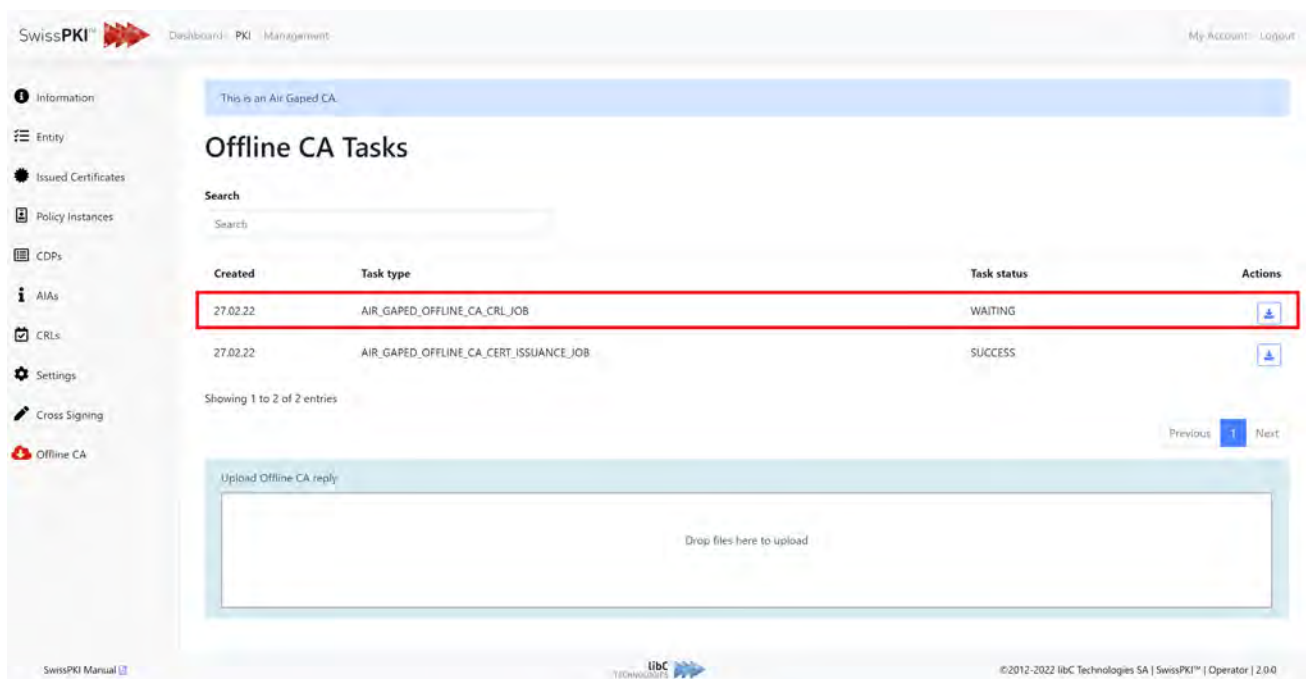
12.9 KB  
reply.p7m

SwissPKI Manual 
©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0



### 12.3.2.1.10.4 Air Gaped CA and Offline CA CRL/ARL issuance

Issue a CRL/ARL generation or Last CRL from the Air Gaped CRL menu . This process involves:

1. Producing an Air Gapped CRL/ARL request for the Offline CA using signed by the seal private key.
2. The CRL/ARL issuing time and next update is set according to the Air Gaped CA settings. The issuing date and next update are not adjusted when signing the CRL/ARL on the Offline CA.
3. Produce a signed request for the Offline CA CRL/ARL issuance (the generated request file name is labelled *offline\_rca\_AirGapedOfflineCACRLJob.p7m*).

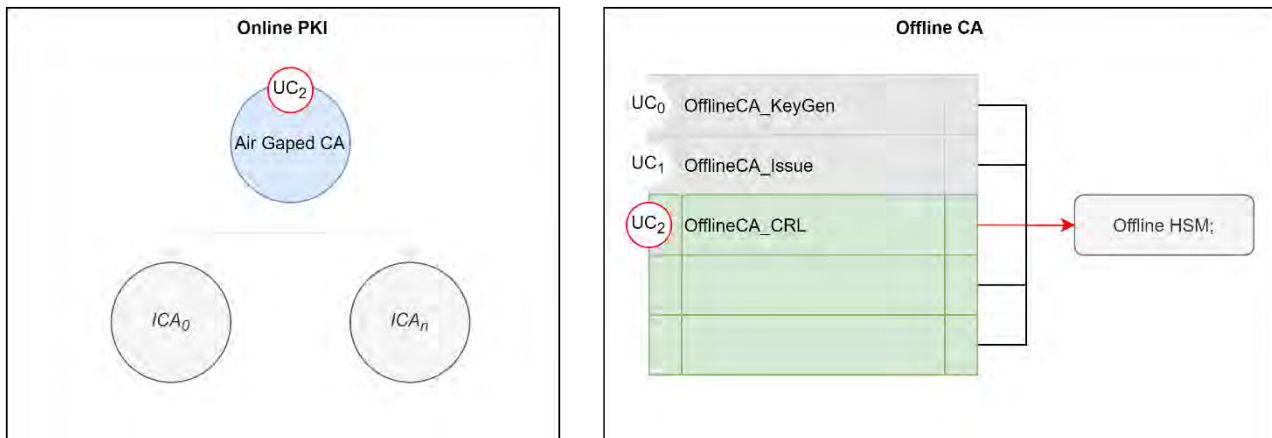


The screenshot shows the 'Offline CA Tasks' section of the SwissPKI management interface. At the top, a blue banner indicates 'This is an Air Gaped CA.' Below this, a search bar is present. The main content is a table with the following data:

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	WAITING	
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	

Below the table, it says 'Showing 1 to 2 of 2 entries' and includes 'Previous' and 'Next' navigation buttons. At the bottom of the interface, there is an 'Upload Offline CA reply' section with a large empty box and the text 'Drop files here to upload'.

4. Issue the Offline CA CRL/ARL using the Air Gaped request file and Offline CA private key




5. Execute the `OfflineCA_CRL` command. This will issue the Offline CA CRL/ARL and produce a reply file for the Air Gaped CA. The sealed object on the HSM partition is used to validate the request.


Record the generated CRL/ARL serial number and generated files. Bring the produced Air Gaped reply file to the Online system to finalize the procedure.

```
usage: OfflineCA_CRL
  -help                Issue CRLs using the Offline CA certificate
                       based on the Air Gaped request and the
                       selected private key.
  -outAirGapedReply <file> Write Air Gaped reply to this output file
  -outCRL <file>        Write CRLs to this output directory
  -privateKeyLabel <file> Offline CA private key label
  -provider <type>     Luna, Primus, ARCA or Kryptus
  -request <file>     Air Gaped CRL request in DER
  -version             print the version information and exit
```

- Complete the Air Gaped CRL/ARL issuance by importing the generated Offline CA reply file. When processed, the Air Gaped request in WAITING status switches to SUCCESS.

27.02.22 AIR\_GAPED\_OFFLINE\_CA\_CRL\_JOB SUCCESS 

---



SwissPKI™  Dashboard - PKI - Management My Account Logout

**Offline CA**

This is an Air Gaped CA

### Offline CA Tasks

Search


Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	PENDING	
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	

Showing 1 to 2 of 2 entries Previous **1** Next


Upload Offline CA reply

**37.5 KB**

reply\_crl.p7m

SwissPKI Manual  ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

- The CRL/ARL are available from the Air Gaped CRL menu.

SwissPKI™  Dashboard - PKI - Management My Account Logout



**CRLs/ARLs List**

This is an Air Gaped CA

### Certificate Revocation Lists | 'Air Gaped CA'

Search Start/End Dates CRL Type

- CDP name, serial number or CA name. Use \* for substring searches

CA	CDP	CRL Type	Serial#	This update	Next update	Actions
Air Gaped CA	Global	ARL	7D9377E3E41825D4151C8D29281145B08029476E	27.02.2022 13:54	07.03.2022 13:54	
Air Gaped CA	Global	CRL	6E9ADDD6FFBF00723013281F97DF6C8610C07963	27.02.2022 13:54	07.03.2022 13:54	

Showing 1 to 2 of 2 entries Previous **1** Next

[Back](#)

### 12.3.2.1.10.5 Air Gaped CA and Offline CA Sub CA issuance

Creating an Issuing CA signed by the Offline CA follows the same configuration and deployment process as the standard SwissPKI Issuing CA.

1. Create a new SwissPKI CA and select the Parent CA. In this case, select the Air Gaped CA



**Create Certificate Authority**

Name\*  
Sub CA RSA 4096

Description\*  
Sub CA RSA 4096

Comment

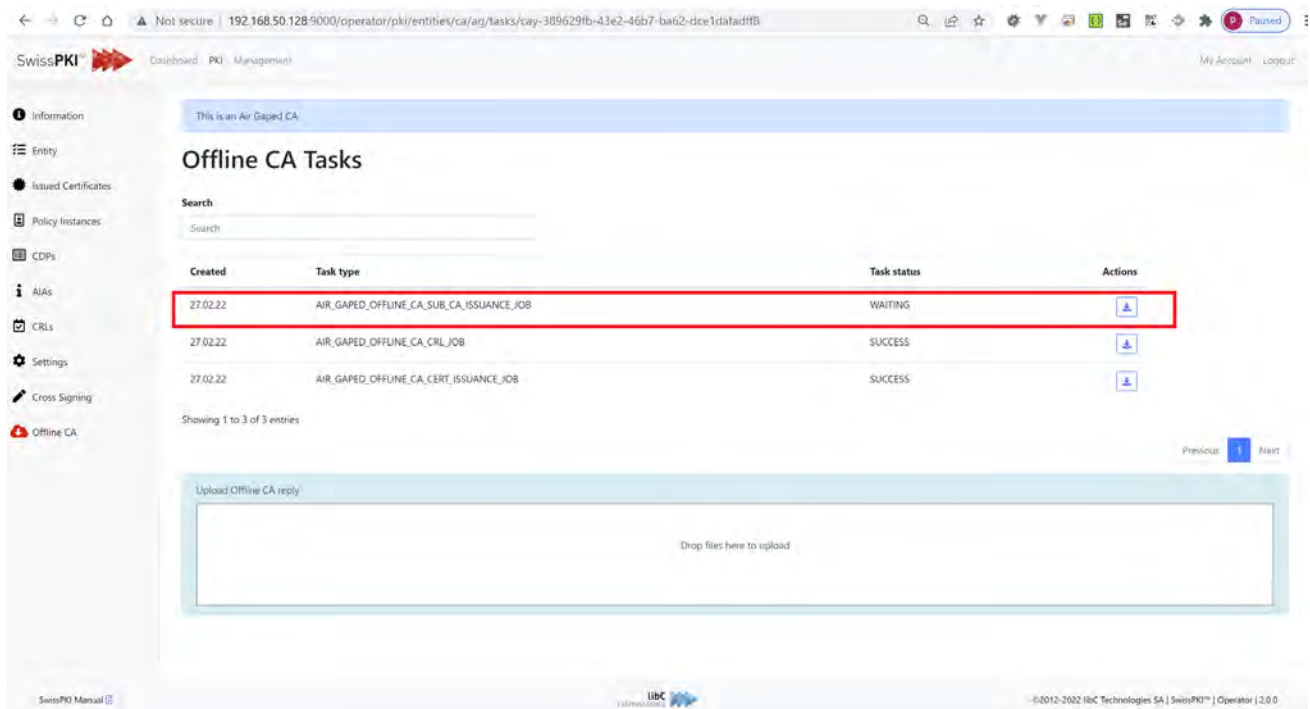
Certificate Authority Type  
Standard

**Parent Certificate Authority**  
Air Gaped CA

Public Trust

Save Create

2. Assign the Sub CA certificate policy template to the Sub CA
3. Generating a SubCA will produce a signed request for the Offline CA (the generated request file name is labelled *offline\_rca\_AirGapedOfflineCASubCAIssuanceJob.p7m*).



SwissPKI Dashboard PKI Management

This is an Air Gaped CA

### Offline CA Tasks

Search

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_SUB_CA_ISSUANCE_JOB	WAITING	<a href="#">↓</a>
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	SUCCESS	<a href="#">↓</a>
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	<a href="#">↓</a>

Showing 1 to 3 of 3 entries

Previous 1 Next

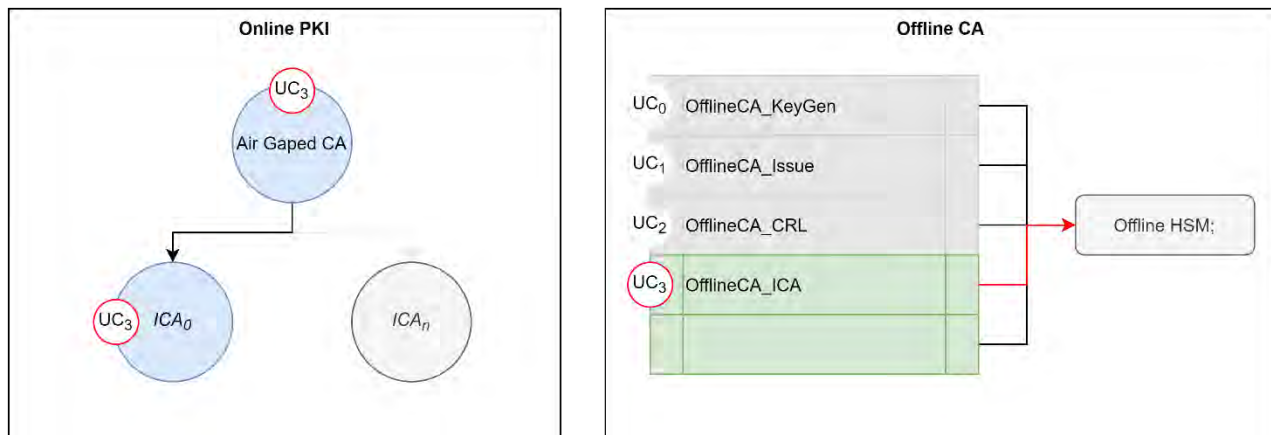
Upload Offline CA reply

Drop files here to upload

SwissPKI Manual libC TECHNOLOGIES ©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

- Execute the `OfflineCA_ICA` command. This will issue the Offline CA SubCA and produce a reply file for the Air Gaped CA. The sealed object on the HSM partition is used to validate the request.


Record the generated SubCA serial number and generated files. Bring the produced Air Gaped reply file to the Online system to finalize the procedure.




```
usage: OfflineCA_ICA
  -help                               Issue an Issuing CA certificate using the
  -outAirGapedReply <file>           Offline CA certificate based on the Air Gaped
  -outCertificate <file>             request and the selected private key.
  -privateKeyLabel <file>           Write Air Gaped reply to this output file
  -provider <type>                  Write certificate to this output file
  -request <file>                   Offline CA private key label
  -version                            Luna, Primus, ARCA or Kryptus
                                     Air Gaped ICA request in DER
                                     print the version information and exit
```

- Complete the Air Gaped SubCA issuance process by importing the generated Offline CA reply file. Record the fingerprints of the Offline CA issued SubCA (available from the SubCA Entity menu).

When processed, the Air Gaped request in WAITING status switches to SUCCESS.

27.02.22 AIR\_GAPED\_OFFLINE\_CA\_SUB\_CA\_ISSUANCE\_JOB SUCCESS 

---




SWISSPKI  Dashboard PKI Management My Account Logout

**Offline CA**

This is an Air Gaped CA.

### Offline CA Tasks

Search

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_SUB_CA_ISSUANCE_JOB	PENDING	
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	SUCCESS	
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	


Showing 1 to 3 of 3 entries

Upload Offline CA reply

12.9 KB

reply\_ca.p7m

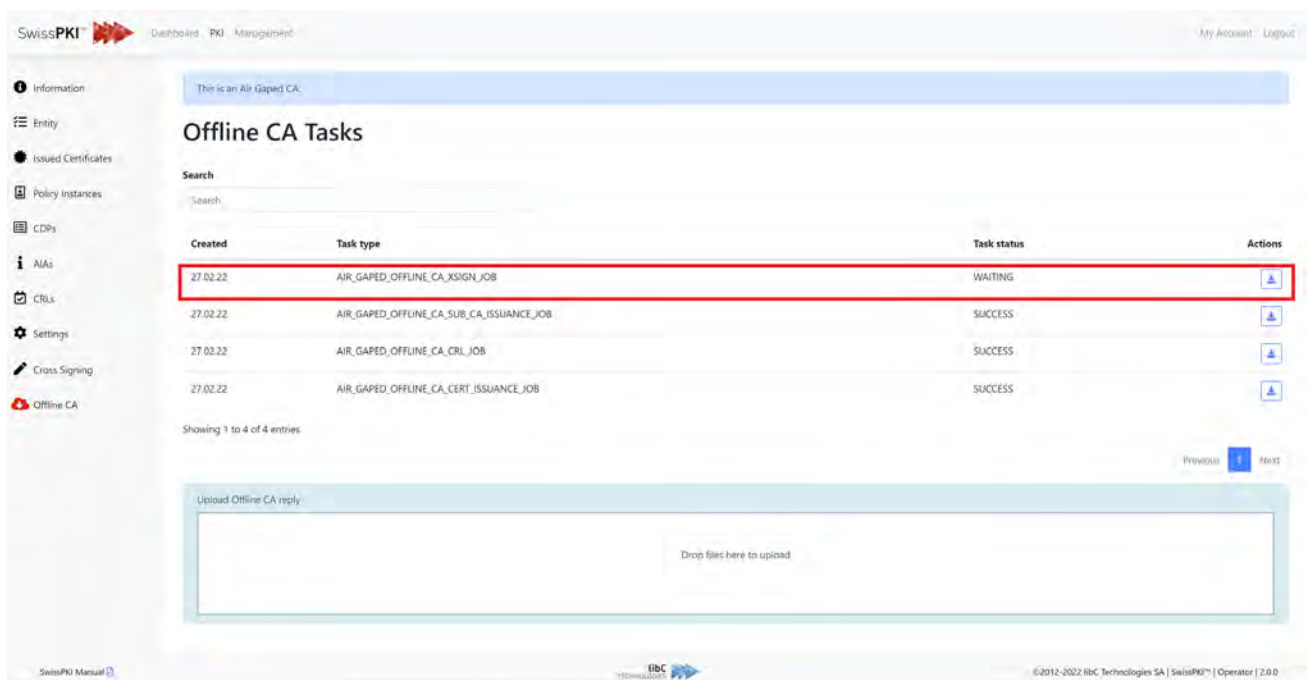
Previous 1 Next

SwissPKI Manual  libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.3.2.1.10.6 Air Gaped CA and Offline CA CSR issuance

Creating CSR request signed by the Offline CA follows the same process flow as the standard SwissPKI CSR generation.

1. Select the Air Gaped CA
2. From the 'Cross Signing' sub menu, select 'generate CSR' from the (+) menu button
3. Generating a CSR will produce a signed request for the Offline CA (the generated request file name is labelled *offline\_rca\_AirGapedOfflineCAXSignJob.p7m*).



The screenshot shows the SwissPKI web interface. The main content area is titled 'Offline CA Tasks'. Below the title is a search bar. A table lists four task entries. The first entry is highlighted with a red box. Below the table is an 'Upload Offline CA reply' section with a file upload area.

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_XSIGN_JOB	WAITING	
27.02.22	AIR_GAPED_OFFLINE_CA_SUB_CA_ISSUANCE_JOB	SUCCESS	
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	SUCCESS	
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	

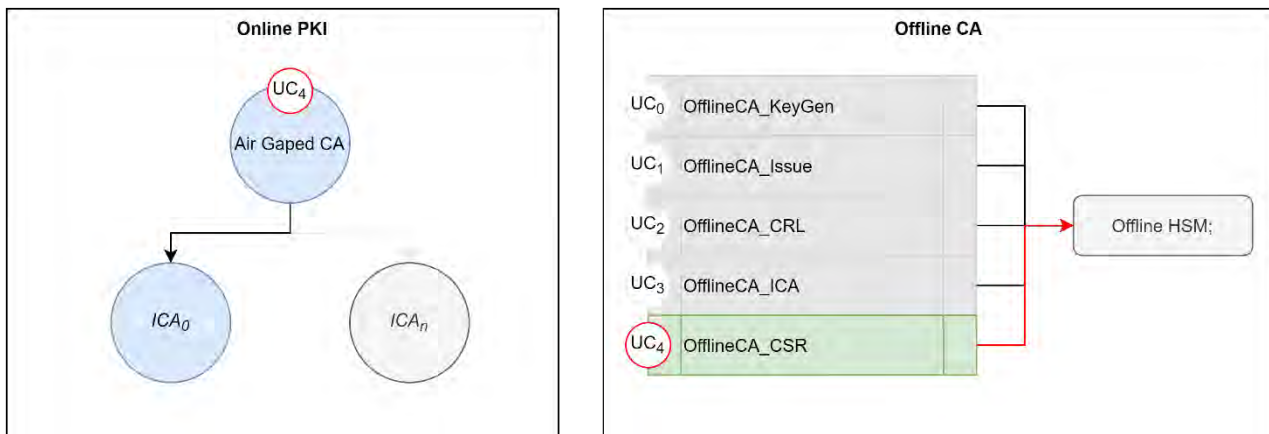
Showing 1 to 4 of 4 entries

Upload Offline CA reply

Drop files here to upload

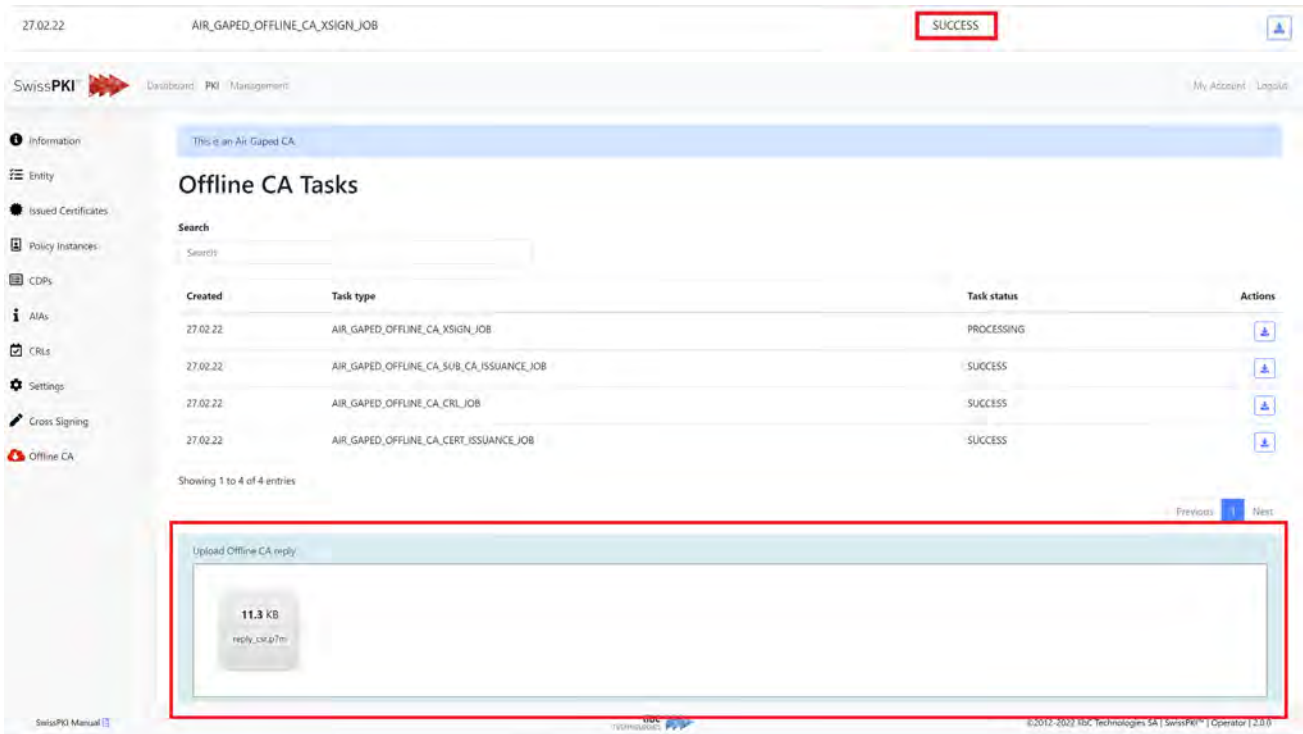


- Execute the `OfflineCA_CSR` command. This will issue the Offline CA CSR and produce a reply file for the Air Gaped CA. The sealed object on the HSM partition is used to validate the request. Bring the produced Air Gaped reply file to the Online system to finalize the procedure.



```
usage: OfflineCA_CSR
  -help                               Issue CSR using the Offline CA certificate
                                       based on the Air Gaped request and the
                                       selected private key.
  -outAirGapedReply <file>           Write Air Gaped reply to this output file
  -outCSR <file>                     Write outCSR to this output file
  -privateKeyLabel <file>           Offline CA private key label
  -provider <type>                  Luna, Primus, ARCA or Kryptus
  -request <file>                   Air Gaped CSR request in DER
  -version                           print the version information and exit
```

- Complete the Air Gaped CSR issuance process by importing the generated Offline CA reply file. When processed, the Air Gaped request in WAITING status switches to SUCCESS.

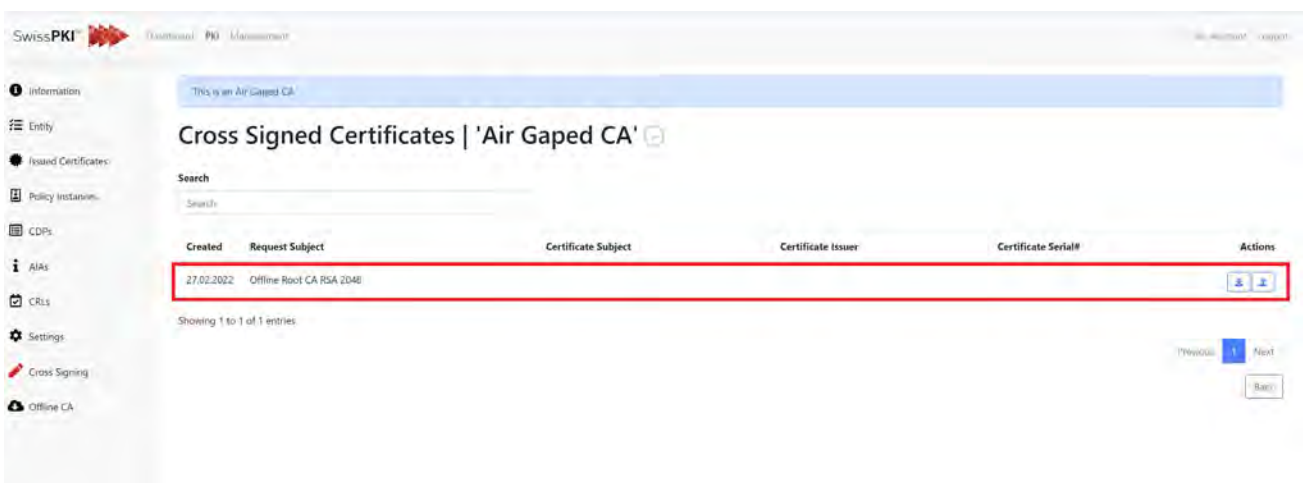


The screenshot shows the SwissPKI management interface. At the top, the date is 27.02.22 and the task ID is AIR\_GAPED\_OFFLINE\_CA\_XSIGN\_JOB, with a red box around the word 'SUCCESS'. The main content area is titled 'Offline CA Tasks' and contains a table with the following data:

Created	Task type	Task status	Actions
27.02.22	AIR_GAPED_OFFLINE_CA_XSIGN_JOB	PROCESSING	
27.02.22	AIR_GAPED_OFFLINE_CA_SUB_CA_ISSUANCE_JOB	SUCCESS	
27.02.22	AIR_GAPED_OFFLINE_CA_CRL_JOB	SUCCESS	
27.02.22	AIR_GAPED_OFFLINE_CA_CERT_ISSUANCE_JOB	SUCCESS	

Below the table, there is a section titled 'Upload Offline CA reply' which contains a file named 'reply.csr.p7m' with a size of 11.3 kB. This section is highlighted with a red box.

- The CSR is available for download from the Air Gaped 'Cross Signing' sub menu. Download the CSR and submit it to a CA for cross signing. Import the issued PKCS#7 certificate chain to complete the cross signing process and optionally switch to the new certificate chain. Switching certificate chain is performed following the standard procedure as described in section 12.3.2.1.9 *Cross Signing*.



The screenshot shows the SwissPKI management interface. The main content area is titled 'Cross Signed Certificates | Air Gaped CA'. It contains a table with the following data:

Created	Request Subject	Certificate Subject	Certificate Issuer	Certificate Serial#	Actions
27.02.2022	Offline Root CA RSA 2048				

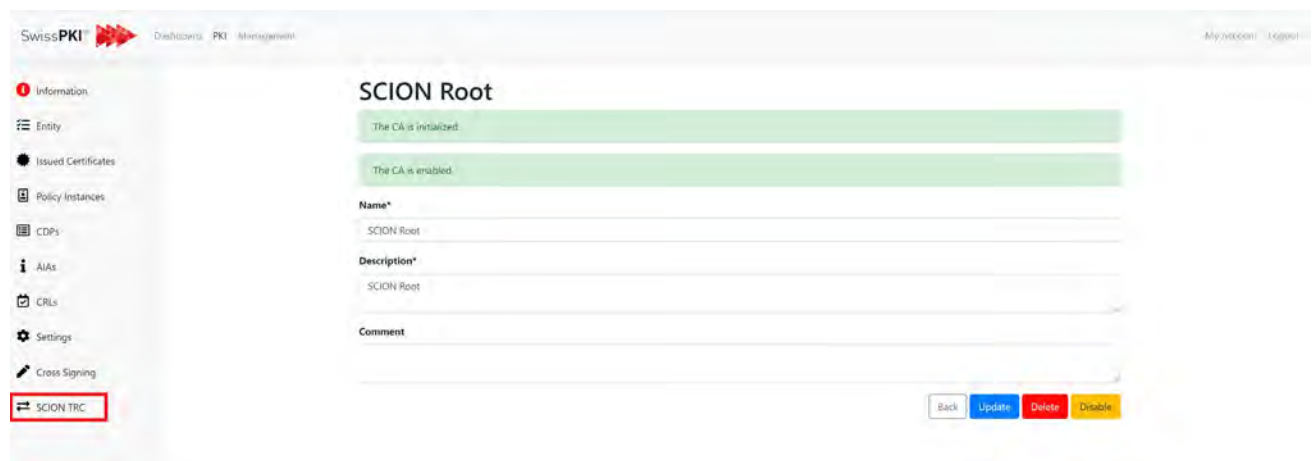
The table entry is highlighted with a red box. Below the table, there are navigation buttons for 'Previous', 'Next', and 'Back'.

### 12.3.2.1.11 TRC Signing

Using SwissPKI as a SCION Control Plane (CP) Root CA, supports Trust Root Configuration signing.

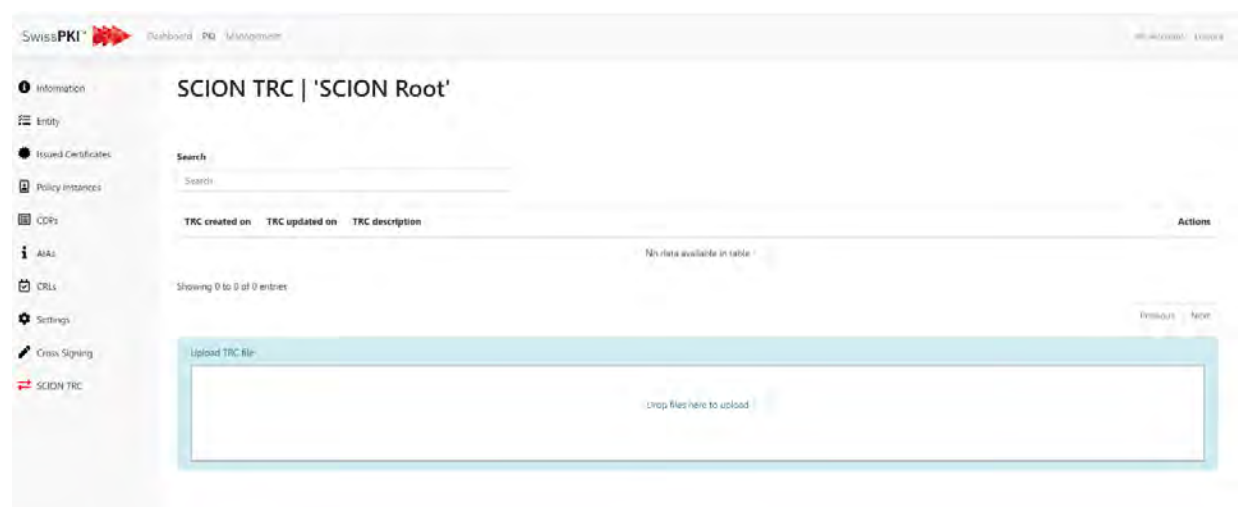
Specifications about TRC signing can be found here:

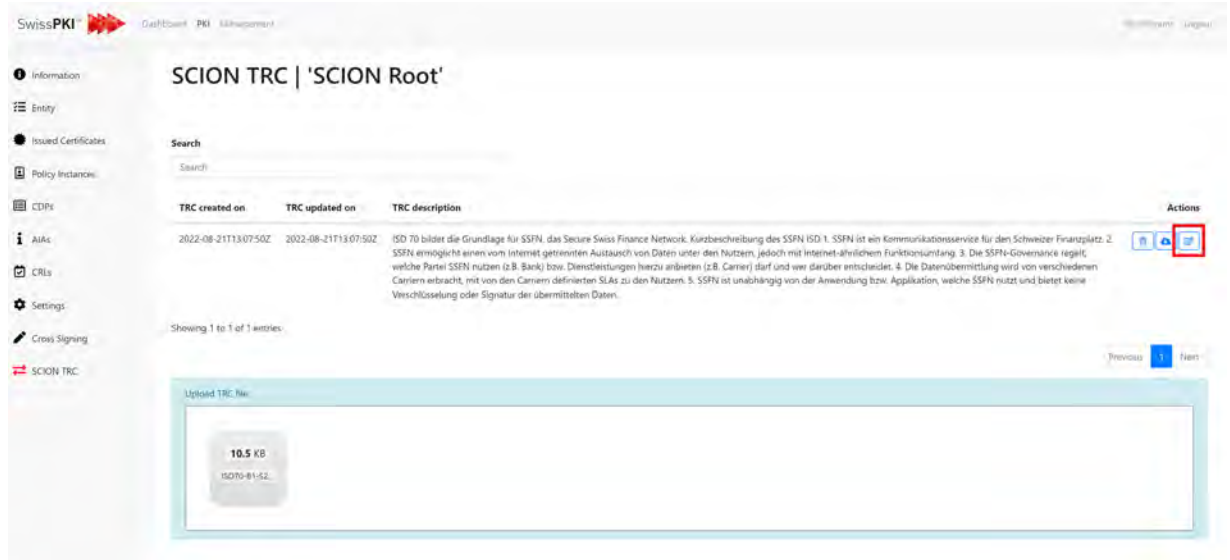
- TRC in general: <https://scion.docs.anapaya.net/en/latest/cryptography/trc.html>
- TRC-Update Process: <https://scion.docs.anapaya.net/en/latest/cryptography/trc.html#trc-update>
- TRC Format: <https://scion.docs.anapaya.net/en/latest/cryptography/trc.html#trc-format>
- Signed TRC Format: <https://scion.docs.anapaya.net/en/latest/cryptography/trc.html#signed-trc-format>



#### 12.3.2.1.11.1 Process steps

1. Upload TRC document into CAO UI

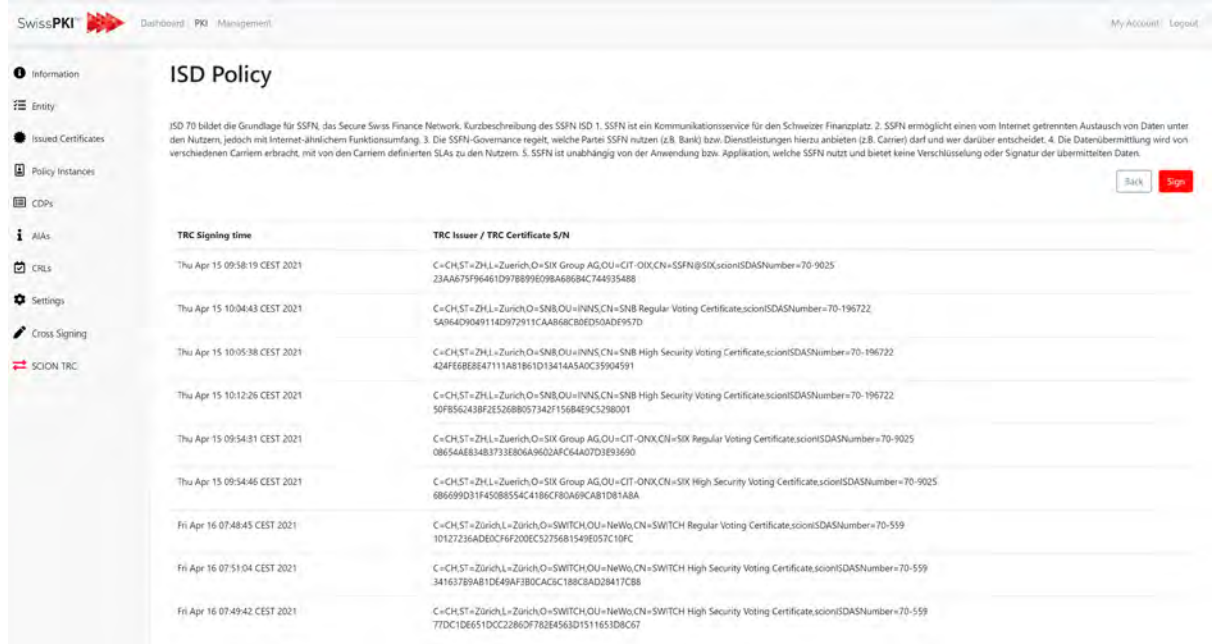




The screenshot shows the SwissPKI interface for the SCION TRC | 'SCION Root'. It includes a search bar, a table with columns for TRC created on, TRC updated on, and TRC description, and an 'Upload TRC File' section showing a 10.5 KB file named ISO70-01-02.

TRC created on	TRC updated on	TRC description
2022-08-21T13:07:50Z	2022-08-21T13:07:50Z	ISO 70 bildet die Grundlage für SSFN, das Secure Swiss Finance Network. Kurzbeschreibung des SSFN ISO 1. SSFN ist ein Kommunikationsservice für den Schweizer Finanzplatz. 2. SSFN ermöglicht einen vom Internet getrennten Austausch von Daten unter den Nutzern, jedoch mit Internet-ähnlichem Funktionsumfang. 3. Die SSFN-Governance regelt, welche Partei SSFN nutzen (z.B. Bank) bzw. Dienstleistungen hierzu anbieten (z.B. Carrier) darf und wer darüber entscheidet. 4. Die Datenübermittlung wird von verschiedenen Carriern erbracht, mit von den Carriern definierten SLAs zu den Nutzern. 5. SSFN ist unabhängig von der Anwendung bzw. Applikation, welche SSFN nutzt und bietet keine Verschlüsselung oder Signatur der übermittelten Daten.

## 2. Present the contents in the UI for approval



The screenshot shows the SwissPKI interface for the ISD Policy. It includes a table with columns for TRC Signing time and TRC Issuer / TRC Certificate S/N.

TRC Signing time	TRC Issuer / TRC Certificate S/N
Thu Apr 15 09:58:19 CEST 2021	C=CH,ST=ZH,L=Zuerich,O=SIX Group AG,OU=CIT-OSX,CN=SSFN@SIX,scionSDASNumber=70-9025 23AA675F96461D978890E09A68684C744535488
Thu Apr 15 10:04:03 CEST 2021	C=CH,ST=ZH,L=Zürich,O=SNB,OU=INNS,CH=SNB Regular Voting Certificate,scionSDASNumber=70-196722 5A864D9049114D977911CAAB68CB0ED50ADE957D
Thu Apr 15 10:05:38 CEST 2021	C=CH,ST=ZH,L=Zürich,O=SNB,OU=INNS,CH=SNB High Security Voting Certificate,scionSDASNumber=70-196722 424F68E8E47111A81B61D13414A5A0C35904591
Thu Apr 15 10:12:26 CEST 2021	C=CH,ST=ZH,L=Zürich,O=SNB,OU=INNS,CH=SNB High Security Voting Certificate,scionSDASNumber=70-196722 50F8562438F2E526880573421156B4E9C5298001
Thu Apr 15 09:54:31 CEST 2021	C=CH,ST=ZH,L=Zürich,O=SIX Group AG,OU=CIT-ONX,CN=SIX Regular Voting Certificate,scionSDASNumber=70-9025 0854A8B3483733E806A9602AFC64A07D1E93690
Thu Apr 15 09:54:46 CEST 2021	C=CH,ST=ZH,L=Zürich,O=SIX Group AG,OU=CIT-ONX,CN=SIX High Security Voting Certificate,scionSDASNumber=70-9025 686699D31F4508B554C4186CF80A69CAB1DB1ABA
Fri Apr 16 07:48:45 CEST 2021	C=CH,ST=Zürich,L=Zürich,O=SWITCH,OU=NeWo,CN=SWITCH Regular Voting Certificate,scionSDASNumber=70-559 10127236ADEDC7F6200EC52736B1549E057C10FC
Fri Apr 16 07:51:04 CEST 2021	C=CH,ST=Zürich,L=Zürich,O=SWITCH,OU=NeWo,CN=SWITCH High Security Voting Certificate,scionSDASNumber=70-559 34163789A81DE49AF3B0CAC6C188CBAD28417C88
Fri Apr 16 07:49:42 CEST 2021	C=CH,ST=Zürich,L=Zürich,O=SWITCH,OU=NeWo,CN=SWITCH High Security Voting Certificate,scionSDASNumber=70-559 77DC1DE651DCC286D782E4563D1511653DB8C67

### 3. Append signature to CMS Signed Data payload (see Signed TRC Format above) by using the Root CA's private key.

#### ISD Policy

ISD 70 bildet die Grundlage für SSFN, das Secure Swiss Finance Network. Kurzbeschreibung des SSFN ISD 1. SSFN ist ein Kommunikationsservice für den Schweizer Finanzplatz. 2. SSFN ermöglicht einen vom Internet getrennten Austausch von Daten unter den Nutzern, jedoch mit Internet-ähnlichem Funktionsumfang. 3. Die SSFN-Governance regelt, welche Partei SSFN nutzen (z.B. Bank) bzw. Dienstleistungen hierzu anbieten (z.B. Carrier) darf und wer darüber entscheidet. 4. Die Datenübermittlung wird von verschiedenen Carriern erbracht, mit von den Carriern definierten SLAs zu den Nutzern. 5. SSFN ist unabhängig von der Anwendung bzw. Applikation, welche SSFN nutzt und bietet keine Verschlüsselung oder Signatur der übermittelten Daten.

TRC Signing time	TRC Issuer / TRC Certificate S/N
Sun Aug 21 15:12:53 CEST 2022	C=CH,O=SCION,OU=Root,CN=Test CA 3789F8EDB42DB99C68C2E85B36F0575476EA569E
Thu Apr 15 09:58:19 CEST 2021	C=CH,ST=ZH,L=Zuerich,O=SIX Group AG,OU=CIT-OIX,CN=SSFN@SIX.scion.ch,OU=70-9025 23AA675F96461D978B899E098A68684C744935488

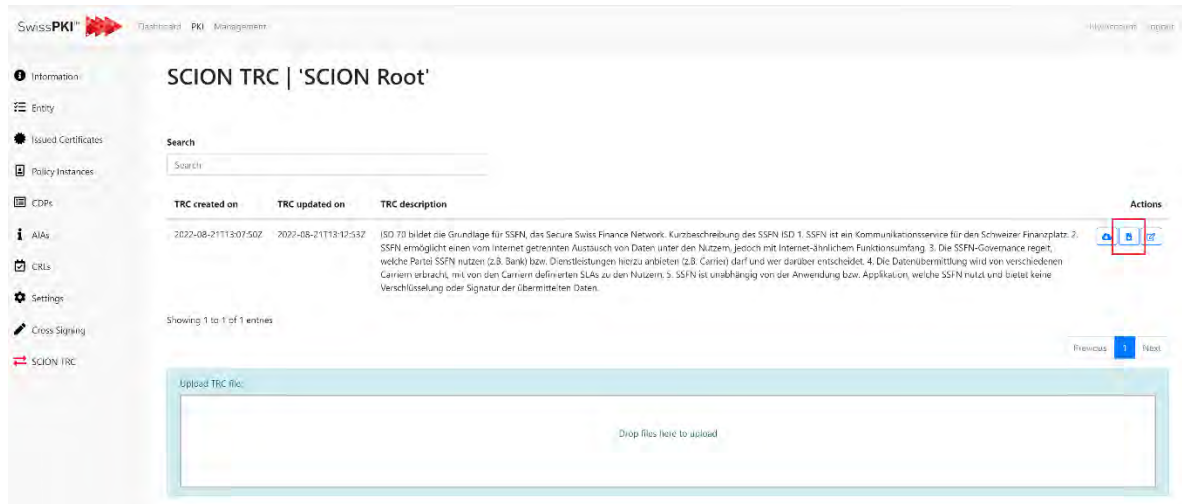
### 4. Log event



The screenshot shows the 'Events' page in the SwissPKI management interface. A search filter for 'Order UUID' is set to 'ord-123456'. The event log shows a single entry:

Created	Type	Source	Created by	Message	Order
21.08.2022 15:12:53	INFO	TRC	Admin Admin	'SIGN_TRC_JOB' handling TRC Sign Request    TRCData[description]=ISD 70 bildet die Grundlage für SSFN, das Secure Swiss Finance Network. Kurzbeschreibung des SSFN ISD 1. SSFN ist ein Kommunikationsservice für den Schweizer Finanzplatz. 2. SSFN ermöglicht einen vom Internet getrennten Austausch von Daten unter den Nutzern, jedoch mit Internet-ähnlichem Funktionsumfang. 3. Die SSFN-Governance regelt, welche Partei SSFN nutzen (z.B. Bank) bzw. Dienstleistungen hierzu anbieten (z.B. Carrier) darf und wer darüber entscheidet. 4. Die Datenübermittlung wird von verschiedenen Carriern erbracht, mit von den Carriern definierten SLAs zu den Nutzern. 5. SSFN ist unabhängig von der Anwendung bzw. Applikation, welche SSFN nutzt und bietet keine Verschlüsselung oder Signatur der übermittelten Daten. ...	

### 5. Present signed payload for download



The screenshot shows the 'SCION TRC | 'SCION Root'' page in the SwissPKI management interface. A table lists the TRC entries:

TRC created on	TRC updated on	TRC description	Actions
2022-08-21 15:12:53	2022-08-21 15:12:53	ISD 70 bildet die Grundlage für SSFN, das Secure Swiss Finance Network. Kurzbeschreibung des SSFN ISD 1. SSFN ist ein Kommunikationsservice für den Schweizer Finanzplatz. 2. SSFN ermöglicht einen vom Internet getrennten Austausch von Daten unter den Nutzern, jedoch mit Internet-ähnlichem Funktionsumfang. 3. Die SSFN-Governance regelt, welche Partei SSFN nutzen (z.B. Bank) bzw. Dienstleistungen hierzu anbieten (z.B. Carrier) darf und wer darüber entscheidet. 4. Die Datenübermittlung wird von verschiedenen Carriern erbracht, mit von den Carriern definierten SLAs zu den Nutzern. 5. SSFN ist unabhängig von der Anwendung bzw. Applikation, welche SSFN nutzt und bietet keine Verschlüsselung oder Signatur der übermittelten Daten.	<a href="#">Download</a>

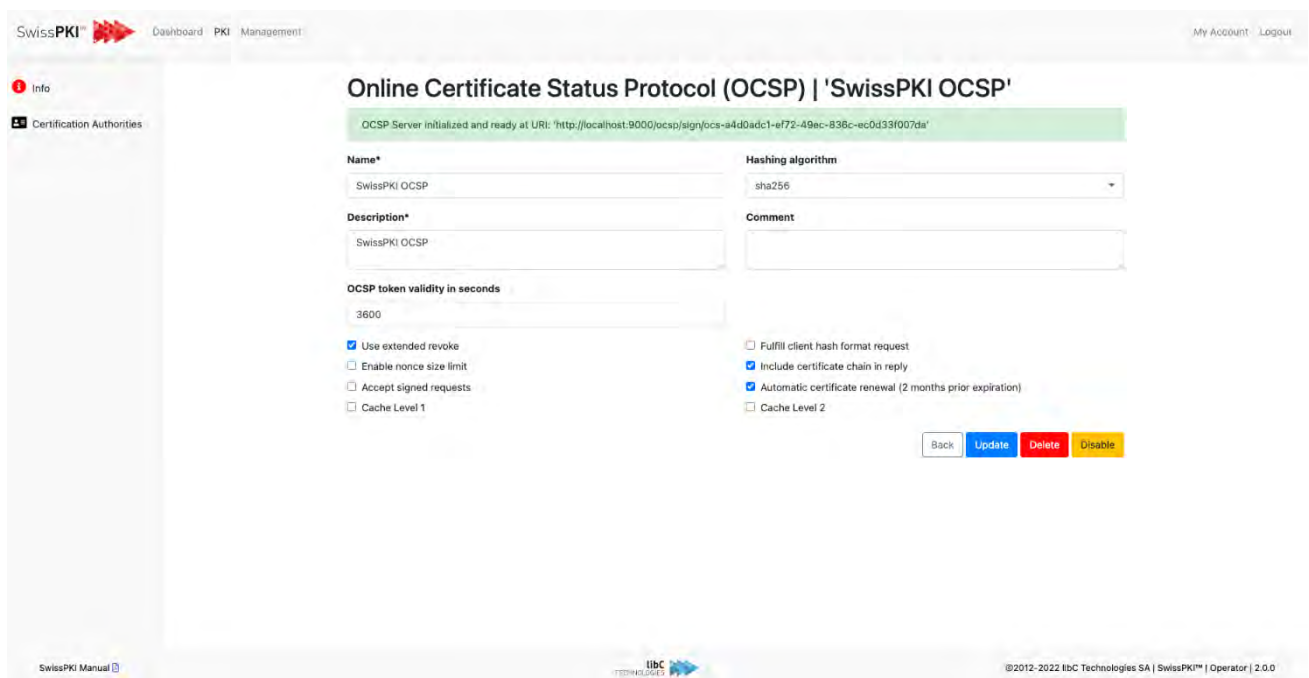
Below the table, there is an 'Upload TRC file:' section with a 'Drop files here to upload' area.

### 12.3.2.2 OCSP

To setup an OCSP Service, you initially create a Certificate Policy Template of type 'OCSP' and associate them to a Certification Authority as a 'Policy Instance' (see 12.3.2.1.4 Policy Instances).

Once initialized, the OCSP service is online <sup>22</sup>. To accept client requests, you must associate Issuing CAs with mapped OCSP Policy Instances to the OCSP service (see Certification Authorities)

The OCSP URL is used in the Certificate Policy Template URL value of the OCSP URI (see 12.3.1.2.7 Authority Information Access)



The screenshot shows the 'Online Certificate Status Protocol (OCSP) | 'SwissPKI OCSP'' configuration page. At the top, a green status bar indicates 'OCSP Server initialized and ready at URI: 'http://localhost:9000/ocsp/sign/ocs-a4d0adc1-e772-49ec-838c-ec0d33f007da''. Below this, the configuration form includes:

- Name\***: SwissPKI OCSP
- Description\***: SwissPKI OCSP
- OCSP token validity in seconds**: 3600
- Hashing algorithm**: sha256
- Comment**: (empty text area)
- Options**:
  - Use extended revoke
  - Enable nonce size limit
  - Accept signed requests
  - Cache Level 1
  - Fulfill client hash format request
  - Include certificate chain in reply
  - Automatic certificate renewal (2 months prior expiration)
  - Cache Level 2

At the bottom right of the form are buttons for 'Back', 'Update', 'Delete', and 'Disable'. The footer of the page contains 'SwissPKI Manual', the libC TECHNOLOGIES logo, and the copyright notice '©2012-2022 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

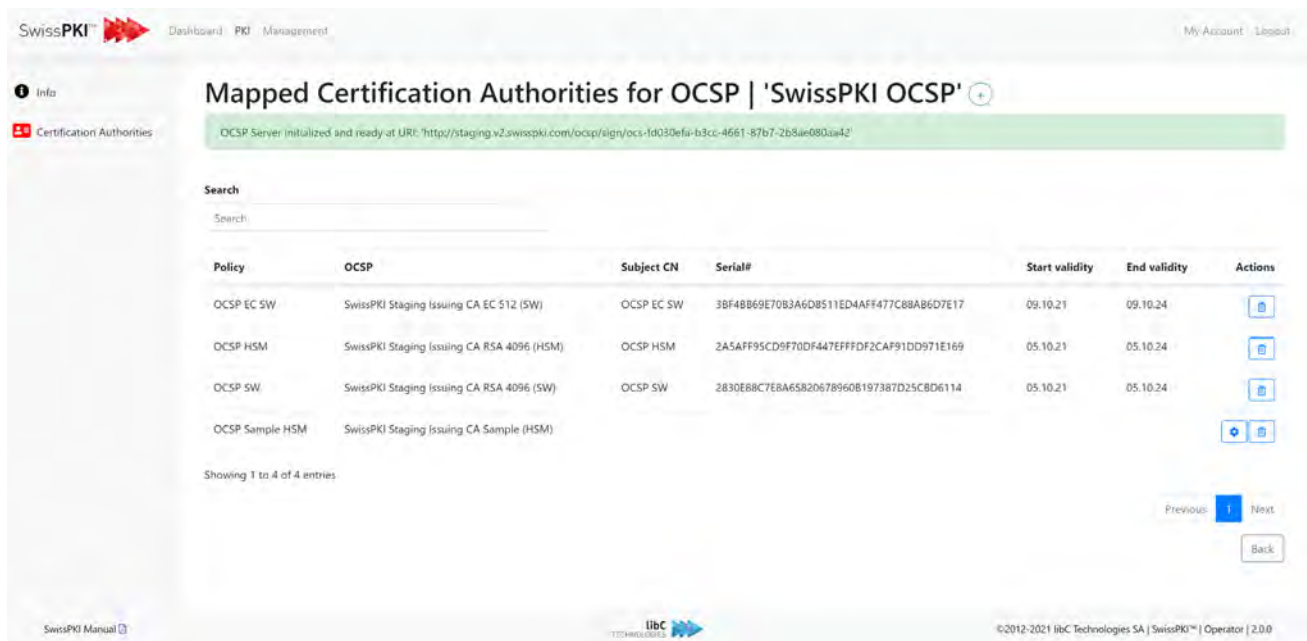
<sup>22</sup> The SwissPKI OCSP module is deployed and the *hosts.conf* contains the domain name of the deployed OCSP module.

Fields	Description
<b>Name</b>	The OCSP's logical name
<b>Hashing Algorithm</b>	The hashing algorithm used by the OCSP
<b>Description</b>	The OCSP's description
<b>Comment</b>	The OCSP's comment
<b>OCSP Token Validity in seconds</b>	The number of second an OCSP token is valid for.
<b>Use extended revoke</b>	Activate or not the use of extended revoke
<b>Fulfill client hash format request</b>	Enable / Disable the fulfillment of client has format requests
<b>Enable nonce size limit</b>	Enable / Disable nonce size limit
<b>Include certificate chain in reply</b>	Include or not the certificate chain in replies sent by the OCSP server.
<b>Accept signed request</b>	Defines is the OCSP server accepts signed requests.
<b>Automatic certificate renewal (2 months before expiration)</b>	Automatically renew 20 days before its expiration including email to system administrator. If automatic renewal is disabled, an email is sent to the system administrator prior expiration
<b>Cache Level 1</b>	If enabled, caches OCSP Responses until 10 minutes before expiration of the token at process level.
<b>Cache Level 2</b>	If enabled, caches OCSP Responses until 10 minutes before expiration of the token at multi OCSP process level (Kubernetes only, multi POD)

Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the OCSP service and revokes the certificate
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests

### 12.3.2.2.1 Certification Authorities

Display the list of associated Certification Authorities serviced by the OCSP service.



The screenshot shows the SwissPKI Management interface. The main heading is "Mapped Certification Authorities for OCSP | 'SwissPKI OCSP'". Below the heading, there is a green status bar indicating "OCSP Server initialized and ready at URL: 'http://staging.v2.swisspki.com/ocsp/sign/ocs-fd030efu-b3cc-4661-87b7-2b8ae080aa42'".

There is a search bar and a table of entries. The table has the following columns: Policy, OCSP, Subject CN, Serial#, Start validity, End validity, and Actions.

Policy	OCSP	Subject CN	Serial#	Start validity	End validity	Actions
OCSP EC SW	SwissPKI Staging Issuing CA EC 512 (SW)	OCSP EC SW	3BF48B69E70B3A6D8511ED4AFF477C88AB6D7E17	09.10.21	09.10.24	[Edit] [Delete]
OCSP HSM	SwissPKI Staging Issuing CA RSA 4096 (HSM)	OCSP HSM	2A5AFF95CD9F70DF447EFFFFD2CAF91DD971E169	05.10.21	05.10.24	[Edit] [Delete]
OCSP SW	SwissPKI Staging Issuing CA RSA 4096 (SW)	OCSP SW	2830E88C7E8A65820678960B197387D25C8D6114	05.10.21	05.10.24	[Edit] [Delete]
OCSP Sample HSM	SwissPKI Staging Issuing CA Sample (HSM)					[Add] [Delete]

Showing 1 to 4 of 4 entries

Navigation: Previous [Selected] Next [Back]

Deleting a Certification Authority from the list revokes the OCSP certificate. The OCSP service will stop replying to the client requests for the removed Certification Authority instance.



Click on the “+” sign to select Certification Authorities to add to the OCSF Service:



SwissPKI™ Dashboard PKI Management My Account Logout

Info Certification Authorities

OCSP Server initialized and ready at URI: 'http://staging.v2.swisspki.com/ocsp/sign/ocsc-fd030efa-b3cc-4661-87b7-2b8ae080aa42'

Search

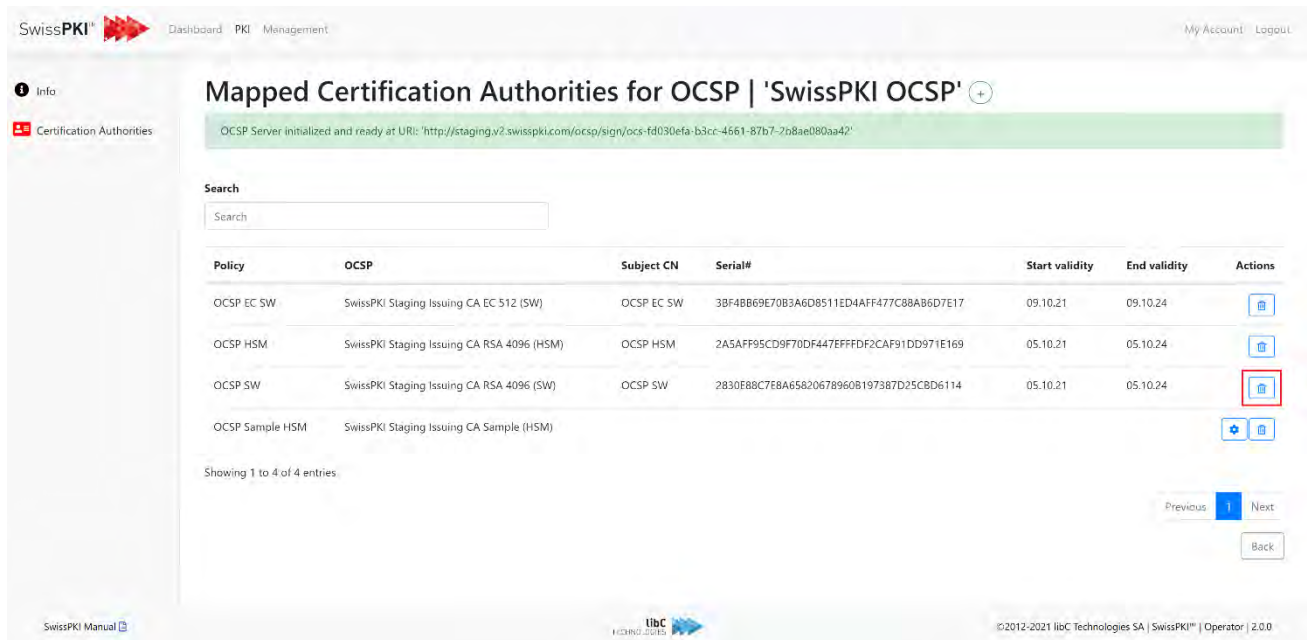
Created	Modified	Certification Authority	Name	Actions
09.10.2021	28.11.2021	SwissPKI Staging Issuing CA Sample (HSM)	OCSP Sample HSM	

Showing 1 to 1 of 1 entries

Previous 1 Next Back

SwissPKI Manual libC TECHNOLOGIES ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

Click on the “+” action to associate the OCSF Policy Instance to the OCSF Service








SwissPKI™ Dashboard PKI Management My Account Logout

Info Certification Authorities

OCSP Server initialized and ready at URI: 'http://staging.v2.swisspki.com/ocsp/sign/ocsc-fd030efa-b3cc-4661-87b7-2b8ae080aa42'

Search

Policy	OCSF	Subject CN	Serial#	Start validity	End validity	Actions
OCSP EC SW	SwissPKI Staging Issuing CA EC 512 (SW)	OCSP EC SW	3BF4BB69E70B3A6D8511ED4AFF477C88AB6D7E17	09.10.21	09.10.24	
OCSP HSM	SwissPKI Staging Issuing CA RSA 4096 (HSM)	OCSP HSM	2A5AFF95CD9F70DF447EFFFDF2CAF91DD971E169	05.10.21	05.10.24	
OCSP SW	SwissPKI Staging Issuing CA RSA 4096 (SW)	OCSP SW	2830F88C7E8A65820678960B197387D25C8D6114	05.10.21	05.10.24	
OCSP Sample HSM	SwissPKI Staging Issuing CA Sample (HSM)					 

Showing 1 to 4 of 4 entries

Previous 1 Next Back

SwissPKI Manual libC TECHNOLOGIES ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

Finalize the OCSF initialization for the Certification Authority by clicking on the “cog” action to generate the key pair and OCSF certificate. The OCSF for the selected Certification Authority is online and ready to accept client requests.

### 12.3.2.3 Time Stamping Authority

To setup a Time Stamp Service, you initially create a Certificate Policy Template of type 'Time Stamp Authority' and associate it to a Certification Authority as a 'Policy Instance' (see 12.3.2.1.4 *Policy Instances*). Creating the Time Stamp service will require a Policy Instance for generating the key pair and certificate.

The Time Stamp service can be created as a SwissPKI service or as an external Time Stamp service. The SwissPKI Time Stamp service is associated with a key pair and certificate signed by a SwissPKI Issuing CA. If you plan to use an external Time Stamp service with the Document Signer Service (see 12.3.2.4 *Document Signer Service*), then you have the option to reference an external Time Stamp server of your choice.

Once initialized, the Time Stamp service is online <sup>23</sup> and ready to accept requests.

## Time Stamp Authority (TSA) | 'SwissPKI TSA RSA (HSM)'

Time Stamp Authority initialized and ready at URI: 'http://staging.v2.swisspki.com/tsa/sign/tsa-73e9a300-b3ef-4a01-842e-8d4a7c91eb4b'

**Name\***

**Description\***

**Signature algorithm**

Include CMS Algorithm Protect Attribute  
 Rekey on renewal  
 Use NTP time source instead of local server time

**Comment**

**TimeStamp Authority Policy Id\***

Automatic certificate renewal (20 days before expiration)

<sup>23</sup> The SwissPKI TSA module is deployed and the *hosts.conf* contains the domain name of the deployed TSA module.

Fields	Description
<b>Name</b>	The TSA's name
<b>Description</b>	The TSA's description
<b>Comment</b>	The TSA's comment
<b>Signature Algorithm</b>	The signature algorithm used by the TSA
<b>TimeStamp Authority Policy ID</b>	The OID of the TSA policy
<b>Include CMS Algorithm Protect Attribute</b>	CMS Algorithm Protect Attribute
<b>Automatic Certificate Renewal</b>	Automatically renew 20 days before its expiration including email to system administrator. If automatic renewal is disabled, an email is sent to the system administrator prior expiration
<b>Rekey on renewal</b>	When enabled, generate a new TSA key pair and certificate. When disabled, the existing TSA key pair is used to issue the new certificate. Note that if rekey is disabled, then the Issuing CA must not enforce unique public key check.
<b>Use NTP time source instead of local time service</b>	Use specified NTPd time source including accuracy in ms. when time stamping. When time is outside +/- accuracy, then the time stamp is not issued. Additionally, the time stamp will not issue a token if NTPd indicates a leap second.

Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the TSA service and revokes the certificate
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests
<b>Renew</b>	Manually renew the TSA certificate

#### 12.3.2.4 Document Signer Service

The Document Signer Service (DSS) is a server side signing service that supports the eIDAS signature formats. The signature server allows you to issue advanced electronic signatures as well as qualified electronic <sup>24</sup> signatures associated with a qualified certificate.

Produce XAdES, PAdES and CAdES signatures that comply with the digital signature standard specified by the ETSI standards:

- eIDAS 2015/1506/EU and ZertES conform Document Signer Service
- XAdES (XML Advanced Electronic Signature ETSI 101 903, TS 103 171 and EN 319 132-1&2)
- CAdES (CMS Advanced Electronic Signature ETSI TS 101 733 and EN 319 122-1&2)
- ASiC (Associated Signature Container ETSI TS 102 918 and EN 319 162-1&2)
- PAdES (PDF Extended Electronic Signature ETSI TS 102 778 and EN 319 142-1&2)
- Supports detached, enveloping, and enveloped structures
- Supports B, T, LT, and LTA signature baselines

In addition to the eIDAS signature formats, the service provides also an URI for directly signing hashes using the document signer service private key.

---

<sup>24</sup> Requires SwissDSS for ZertES or eIDAS signatures with SCAL2. SwissDSS is the qualified version of the DSS module.

To setup a Document Signer Service, you initially create a Certificate Policy Template of type 'Document Signer' and associate it to a Certification Authority as a 'Policy Instance' (see 12.3.2.1.4 *Policy Instances*). Creating the Document Signer service will require a Policy Instance for generating the key pair and certificate.

Once initialized, the Time Stamp service is online <sup>25</sup> and ready to accept requests.

## Document Signer | 'SwissPKI DSS RSA (HSM)'

Document Signer enabled at URI: 'https://staging.v2.swisspki.com/dss/sign/dss-77b0a08c-1933-4df0-a7da-819799e8f37a'

Document Signer (hash only) enabled at URI: 'https://staging.v2.swisspki.com/dss/sign/hash/dss-77b0a08c-1933-4df0-a7da-819799e8f37a'

**Name\***

**Description\***

**Container\***

**Base line\***

**Time Stamp Authorities**

Automatic certificate renewal (20 days before expiration)  
 Rekey on renewal  
 Enable JWT

**Comment**

**Format\***

**Signature Algorithm\***

**Envelope\***

Allow signing with expired certificate

[Back](#) [Update](#) [Delete](#) [Disable](#) [Renew](#)

<sup>25</sup> The SwissPKI DSS module is deployed and the *hosts.conf* contains the domain name of the deployed DSS module.

Fields	Description
<b>Name</b>	The DSS logical name
<b>Certificate Policy Instance</b>	The policy instance you wish to use to create the DSS certificate
<b>Description</b>	The DSS description
<b>Comment</b>	The DSS comment
<b>Container</b>	The DSS container. Three options are available: <ul style="list-style-type: none"> <li>• None</li> <li>• ASiC-S</li> <li>• ASiC-E</li> </ul>
<b>Format</b>	The DSS format. Two options are available: <ul style="list-style-type: none"> <li>• CAdES</li> <li>• XAdES</li> </ul>
<b>Base line</b>	The DSS baseline. Four choices are available: <ul style="list-style-type: none"> <li>• Baseline-B</li> <li>• Baseline-T</li> <li>• Baseline-LT</li> <li>• Baseline-LTA</li> </ul>
<b>Signature Algorithm</b>	The signature algorithm used to sign documents: <ul style="list-style-type: none"> <li>• sha224</li> <li>• sha256</li> <li>• sha384</li> <li>• sha512</li> <li>• sha3-224</li> <li>• sha3-256</li> <li>• sha3-384</li> <li>• sha3-512</li> </ul>
<b>Time Stamp Authority</b>	The Time Stamp Authority linked to the DSS server <b>Note:</b> The TSA can be of type SwissPKI or External
<b>Envelope</b>	The DSS envelope
<b>Automatic certificate renewal (20 days before expiration)</b>	Automatically renew 20 days before its expiration including email to system administrator. If automatic renewal is

	disabled, an email is sent to the system administrator prior expiration
<b>Allow signing with expired certificate</b>	Documents are signed if the DSS certificate expired.
<b>Rekey on renewal</b>	<p>When enabled, generates a new key along with a new certificate when renewing the service.</p> <p>When disabled, only a new certificate is issued using the same (initial) private key. Note that the Issuing CA will need to be configured to allow reusing keys (disable unique public key check)</p>
<b>Enable JWT</b>	<p>When enabled, enforces authentication validation when submitting signing requests to the service.</p> <p>Supported JWT keys and hashes are respectively, HMAC, RSA, EC and X.509 and SHA256/SHA384/SHA512</p>

Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the DSS service and revokes the certificate
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests
<b>Renew</b>	Manually renew the service's certificate

#### **12.3.2.4.1 JWT configuration**

You can enforce JWT authentication on the Document Signer Service URLs by enabling the JWT configuration. Each request to the signature or hash signing URI is validated against the registered keys. You define the supported signing algorithm for the signing service by selecting one or more of the following JWT hashes and key types.

##### **12.3.2.4.1.1 Supported JWT hashes**

The Document Signer service can enforce the following signing hash algorithms when validating a JWT token

- SHA256
- SHA384
- SHA512

##### **12.3.2.4.1.2 Supported JWT keys**

The Document Signer service can enforce the following signing key types when validating a JWT token

- HMAC
- RSA
- EC
- X.509



### 12.3.2.4.1.3 Registered JWT keys

You register public keys or shared secrets with the Document Signer service by selecting 'Add JWT key'

Add JWT key
✕

**Label\***

**Supported JWT keys**

EC

**Key\***

```

-----BEGIN PUBLIC KEY-----
MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE7iSTM9iB1tw0pAAjYSQ3KxmHTwPt
LwHHY10vEduKJUC023BLnJomVHZT7BCXEFrJF145/nrhA0XtuFQYioQi9g==
-----END PUBLIC KEY-----

```

**Expires on**

Cancel
Create

Registered keys are listed in the JWT configuration tab of the Document Signer service

Certificate details | JWT configuration

Supported JWT hash algorithms: 2 selected [Update] | Supported JWT keys: 4 selected [Update]

[Add JWT key]

Search:

Created	Modified	Label	Key type	Expires on	Actions
03.08.2023	03.08.2023	my-hmac	HMAC	-	[Delete]

Showing 1 to 1 of 1 entries

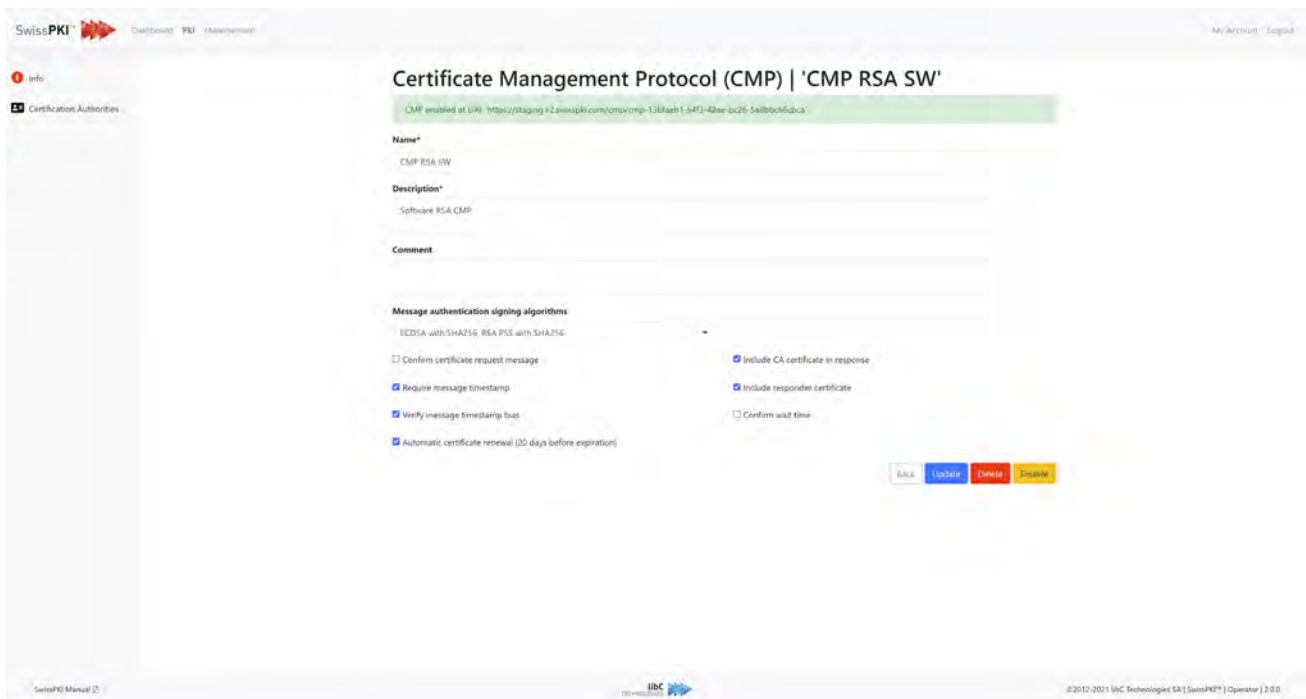
Attribute	Description
<b>Created</b>	Creation date
<b>Modified</b>	Modification date
<b>Label</b>	Free text label. The key label is printed in the Document Signer service when signing a request.
<b>Key type</b>	Key type. One of HMC, RSA, EC or X.509 Note: for RSA, EC and X.509 only upload PEM encoded public keys or X.509 certificate.
<b>Expires on</b>	Expiration date of the key. When empty, the key is usable until it gets deleted from the list. Note: for X.509 certificates, the expires on field is filled in with the end validity of the certificate
<b>Action</b>	Delete selected key

### 12.3.2.5 Certificate Management Protocol

The Certificate Management Protocol (CMP) is a service to which certification and revocation requests are sent using the SwissPKI CMP SDK.

To setup a CMP Service, you initially create Certificate Policy Template of types ‘CMP Signer’ and ‘CMP Cipher’ (see 12.2.3.7 *Certificate Management Protocol*) and associate them to a Certification Authority as a ‘Policy Instance’ (see 12.3.2.1.4 *Policy Instances*). Creating the CMP service requires two Policy Instances for generating the key pairs and certificates.

Once initialized, the CMP service is online <sup>26</sup>. To accept client requests, you must associate Issuing CAs with mapped CMP Policy Instances to the CMP service (see 12.3.2.5.1 *Certification Authorities*)



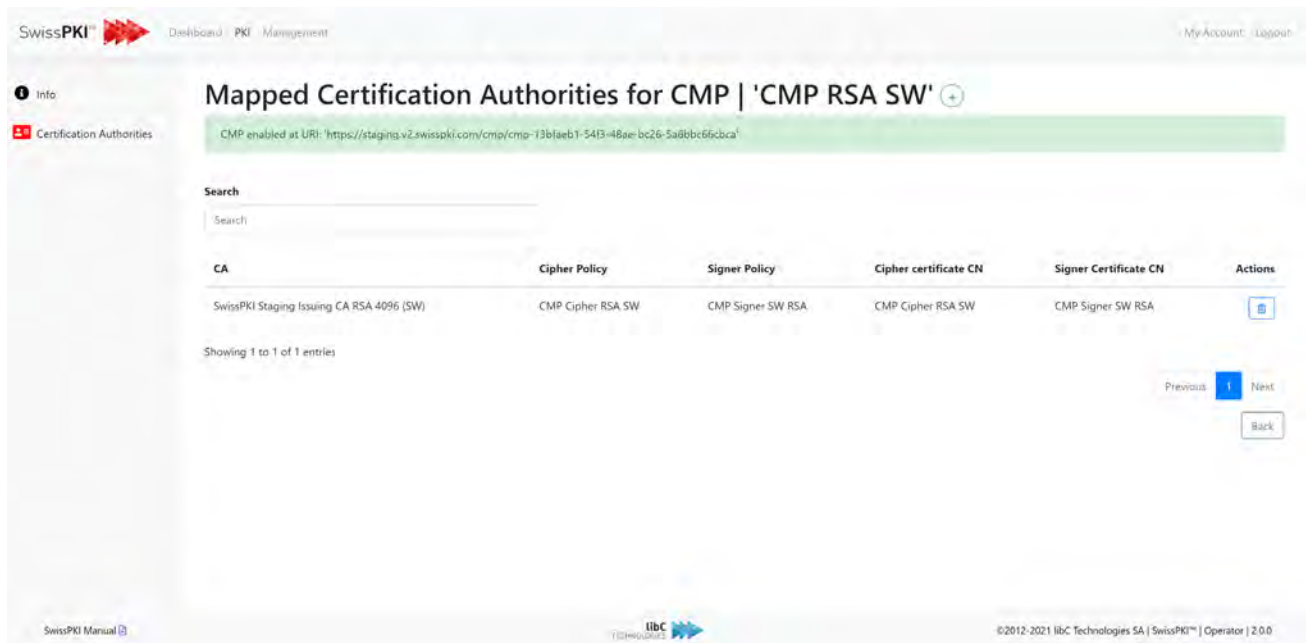
<sup>26</sup> The SwissPKI CMP module is deployed and the *hosts.conf* contains the domain name of the deployed CMP module.

Fields	Description
<b>Name</b>	The CMP's name
<b>Description</b>	The CMP's description
<b>Comment</b>	The CMP's comment
<b>Message authentication signing algorithms</b>	The signature algorithms used by the CMP to send authentication messages. Two options are available: <ul style="list-style-type: none"> <li>• RSA PSS with sha256</li> <li>• EC DSA with sha256</li> </ul>
<b>Confirm certificate request message</b>	If this option is selected, a confirmation is required for certificate request messages.
<b>Include CA certificate in response</b>	Include or not the CA certificate in responses.
<b>Require message timestamp</b>	Require or not messages timestamp
<b>Include responder certificate</b>	If this option is selected, the responder certificate is included in the CMP
<b>Verify message timestamp bias</b>	Verify or not the messages timestamp bias
<b>Confirm wait time</b>	Confirm or not the wait time
<b>Automatic certificate renewal (20 days before expiration)</b>	Automatically renew 20 days before its expiration including email to system administrator. If automatic renewal is disabled, an email is sent to the system administrator prior expiration

Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the CMP service and revokes the certificate
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests

### 12.3.2.5.1 Certification Authorities

List the associated Issuing Certification Authorities with the CMP service.



The screenshot shows the SwissPKI web interface. The main heading is "Mapped Certification Authorities for CMP | 'CMP RSA SW'". Below this, a green banner indicates "CMP enabled at URI: https://stagins.v2.swisspki.com/cmp/cmp-13b1aeb1-54f3-48ae-bc26-5a8bcb66c6ca". A search bar is present. The main content is a table with the following columns: CA, Cipher Policy, Signer Policy, Cipher certificate CN, Signer Certificate CN, and Actions.

CA	Cipher Policy	Signer Policy	Cipher certificate CN	Signer Certificate CN	Actions
SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Cipher RSA SW	CMP Signer SW RSA	CMP Cipher RSA SW	CMP Signer SW RSA	[Edit]

Showing 1 to 1 of 1 entries

Navigation: Previous, Next, Back

Deleting an Issuing Certification Authority revokes the associated certificates.

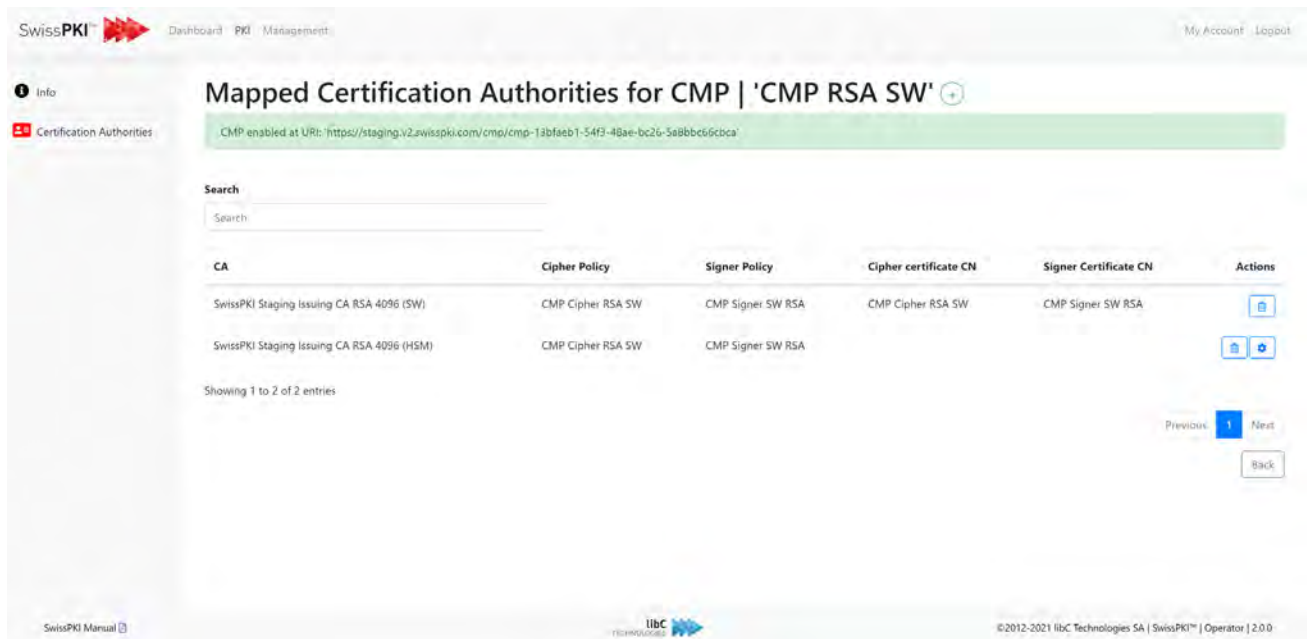
To add an Issuing Certification Authority, click on “+” sign to display the list of available Issuing Certification Authorities which can be assigned to the CMP Service. Remember to assign a CMP Signer and Cipher Policy Instances to the Issuing Certification Authority for it to appear in the list.

Click “+” sign to add the Issuing Certification Authority



The screenshot shows the SwissPKI web interface. The main heading is "Candidate Certification Authorities for CMP | 'CMP RSA SW'". Below this, there is a green bar indicating the CMP is enabled at a specific URI. The interface is divided into sections for CA, Cipher Policy, and Signer Policy. The CA section shows "SwissPKI Staging Issuing CA RSA 4096 (HSM)". The Cipher Policy section has a dropdown menu with "CMP Cipher RSA SW" selected. The Signer Policy section has a dropdown menu with "CMP Signer SW RSA" selected. There is a blue "+" button to add a new entry and a "Back" button. The footer includes "SwissPKI Manual", the libC logo, and copyright information: "©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0".

Finalize the initialization by clicking on the “cog” action to generate the key pairs and certificates.



SwissPKI Dashboard PKI Management My Account Logout

Info Certification Authorities

### Mapped Certification Authorities for CMP | 'CMP RSA SW'

CMP enabled at URI: <https://staging.v2.swisspki.com/cmp/cmp-13bfaeb1-54f3-48ae-bc26-5a8bbc66c0ca>

Search

CA	Cipher Policy	Signer Policy	Cipher certificate CN	Signer Certificate CN	Actions
SwissPKI Staging Issuing CA RSA 4096 (SW)	CMP Cipher RSA SW	CMP Signer SW RSA	CMP Cipher RSA SW	CMP Signer SW RSA	
SwissPKI Staging Issuing CA RSA 4096 (HSM)	CMP Cipher RSA SW	CMP Signer SW RSA			

Showing 1 to 2 of 2 entries

Previous 1 Next

Back

SwissPKI Manual libC TECHNOLOGIES ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

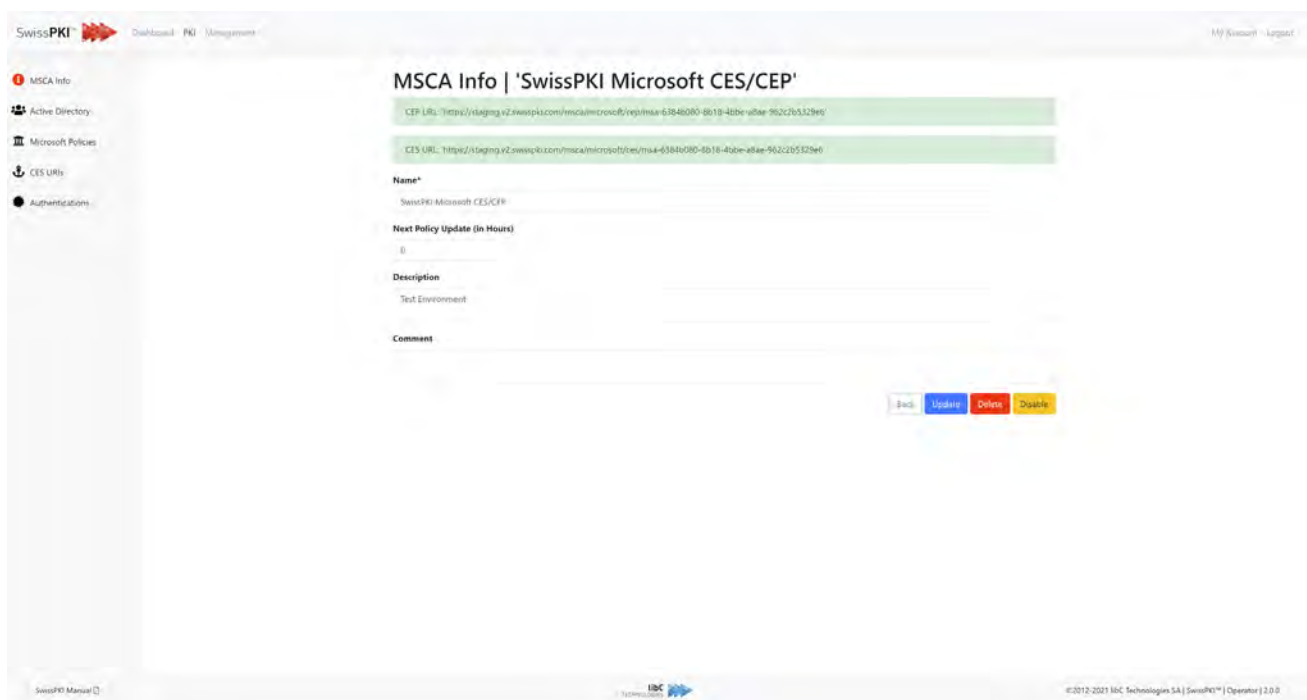
### 12.3.2.6 Microsoft CES / CEP

Microsoft CES/CEP allows you to integrate Microsoft autoenrollment with SwissPKI. For detailed setup instructions, please refer to

<https://support.swisspki.com/support/solutions/articles/44001819320-microsoft-ces-and-cep-setup>

#### 12.3.2.6.1 MSCA Info

The info pane indicates the CES and CEP URLs used by the IIS CES and CEP extensions for redirecting autoenrollment requests.

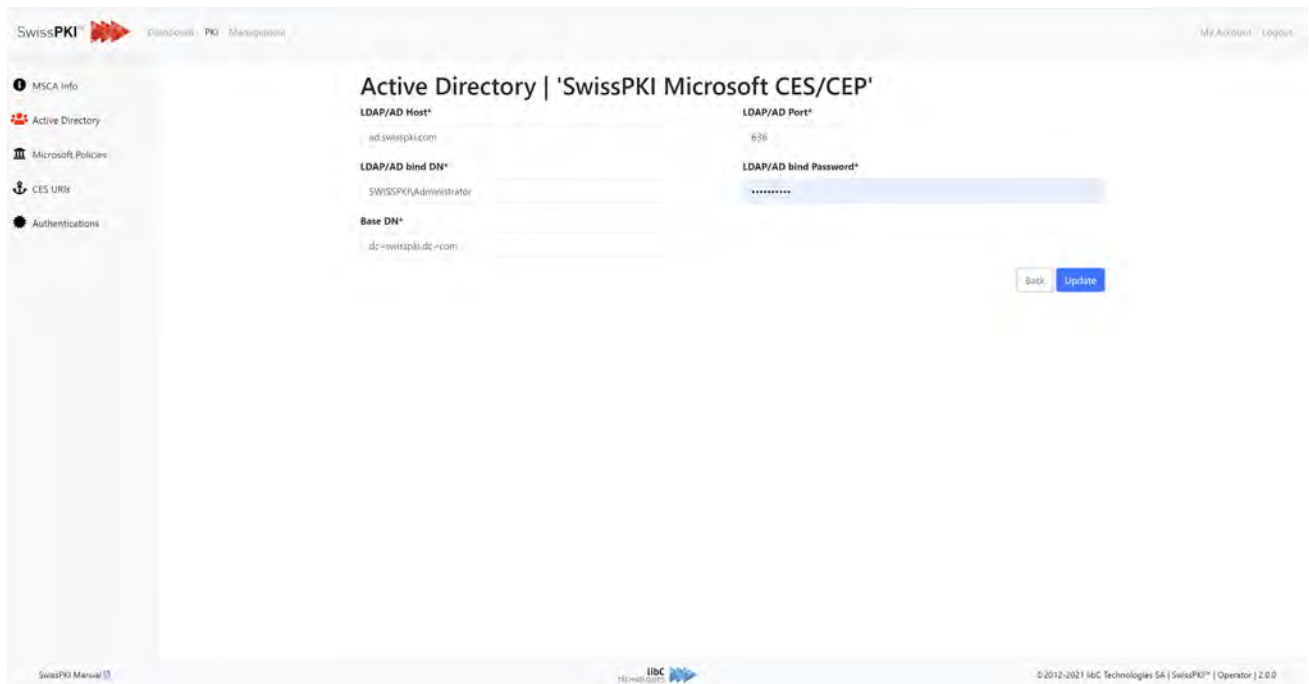


Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the MSCA service (CES and CEP), removes all linked certificate policy instances from the service
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests



### 12.3.2.6.2 Active Directory

Define the connection settings to the Active Directory. When processing registration requests, the settings defined in *12.3.1.2.26 Microsoft Policy* are executed against this server configuration.



The screenshot shows the SwissPKI web interface for configuring Active Directory. The page title is "Active Directory | 'SwissPKI Microsoft CES/CEP'". The configuration fields are as follows:

Field	Value
LDAP/AD Host*	ad.swisspk.com
LDAP/AD Port*	626
LDAP/AD bind DN*	SWISSPKIAdministrator
LDAP/AD bind Password*	*****
Base DN*	dc=swisspk,dc=com

At the bottom right of the form, there are two buttons: "Back" and "Update".

Footer information: SwissPKI Manual | libC TECHNOLOGIES | © 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0

### 12.3.2.6.3 Microsoft Policies

When assigning Microsoft policy instance (see *12.3.2.1.4 Policy Instances*) to a Certification Authority, you can add/remove specific certificate policies to the Microsoft CES/CEP service. The assigned certificate policies are made available for autoenrollment.



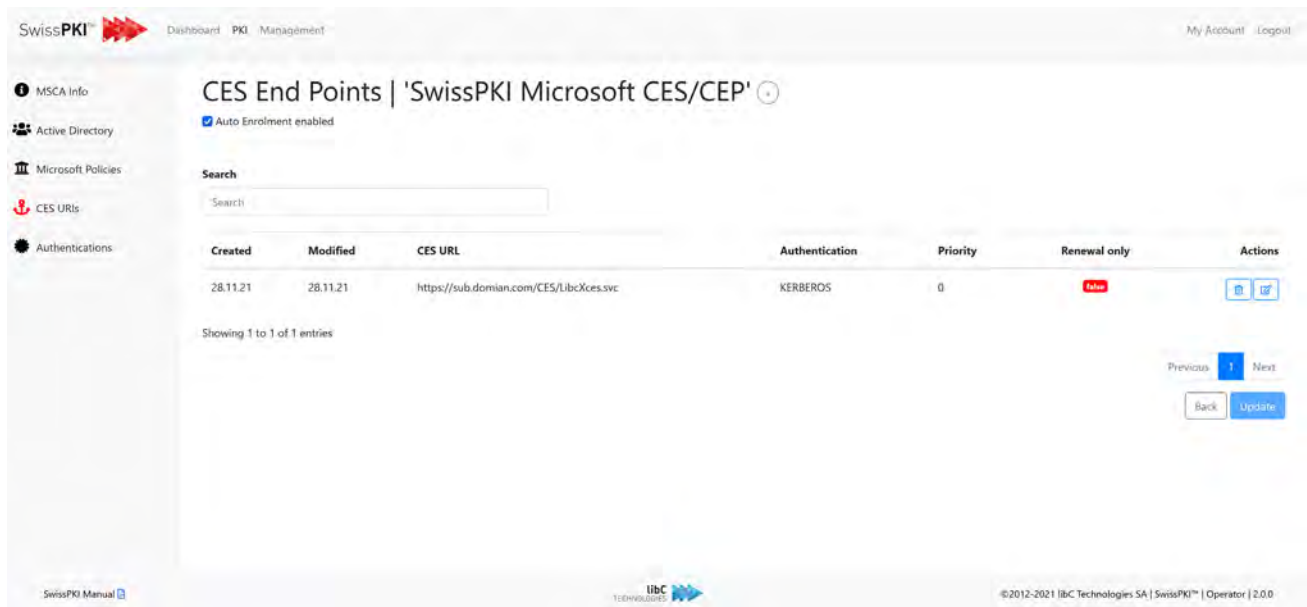
The screenshot shows the 'Microsoft Policy Mappings' page for the 'SwissPKI Microsoft CES/CEP' service. The interface includes a search bar and a table with the following data:

Created	Modified	CA Name	Template name	Policy Name	Client Name	Actions
28.11.2021	28.11.2021	SwissPKI Staging Issuing CA RSA 4096 (HSM)	Microsoft End User	Microsoft End User	Client B	

Below the table, it indicates 'Showing 1 to 1 of 1 entries'. Navigation buttons for 'Previous', 'Next', and 'Back' are visible. The footer contains the libC logo and copyright information: '© 2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.3.2.6.4 CES URIs

Certificate Enrollment Server CES is the deployed IIS CES module URL. Microsoft certificate requests are directed to the registered URLs to MSCA CES URL. You can register multiple CES URL.



SwissPKI Dashboard PKI Management My Account Logout

MSCA Info Active Directory Microsoft Policies CES URIs Authentications

CES End Points | 'SwissPKI Microsoft CES/CEP'

Auto Enrolment enabled

Search

Created	Modified	CES URL	Authentication	Priority	Renewal only	Actions
28.11.21	28.11.21	https://sub.domain.com/CES/libcCes.svc	KERBEROS	0	<input type="checkbox"/>	<input type="button" value="edit"/> <input type="button" value="delete"/>

Showing 1 to 1 of 1 entries

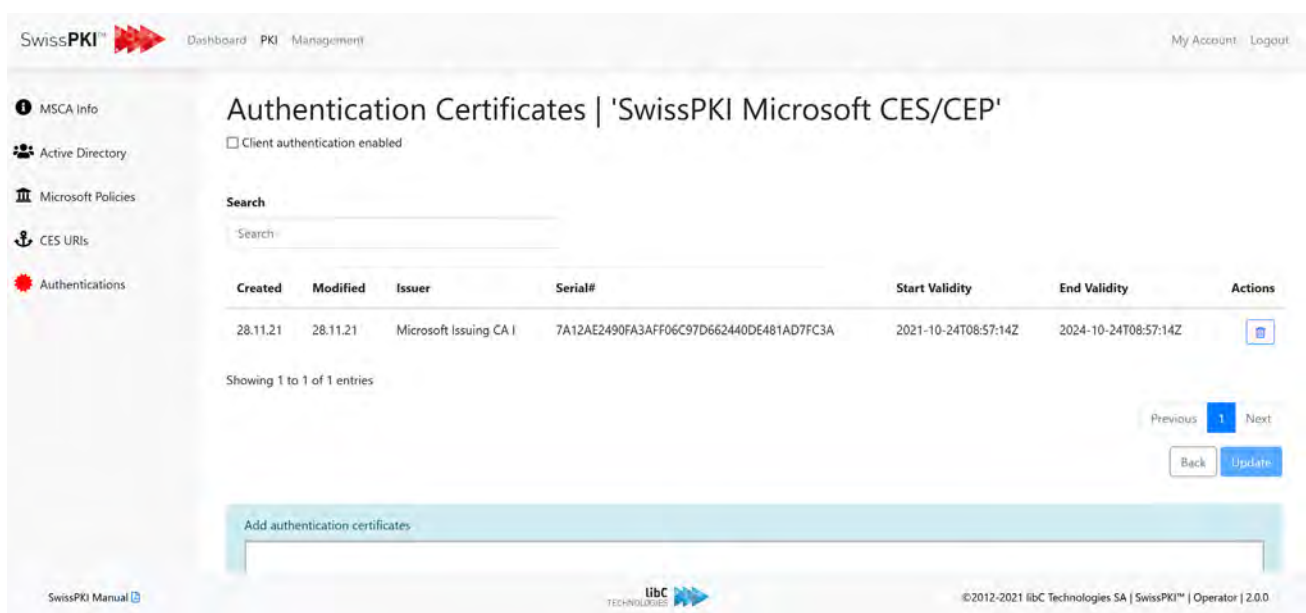
Previous Next Back Update

SwissPKI Manual libC TECHNOLOGIES ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0


### 12.3.2.6.5 Authentication

When enabled, you register the authentication certificates deployed at the IIS CES and CEP Module. Requests sent from the IIS proxy to the MSCA services (CES and CEP) can be digitally signed and SwissPKI will validate the incoming signed requests.

For each issued certificate on the IIS CES and CEP modules, you register the authentication certificates with the MSCA service.



The screenshot shows the SwissPKI web interface. The main heading is 'Authentication Certificates | 'SwissPKI Microsoft CES/CEP''. Below the heading, there is a checkbox for 'Client authentication enabled' which is currently unchecked. A search bar is present with the text 'Search'. Below the search bar is a table with the following columns: Created, Modified, Issuer, Serial#, Start Validity, End Validity, and Actions. The table contains one entry:

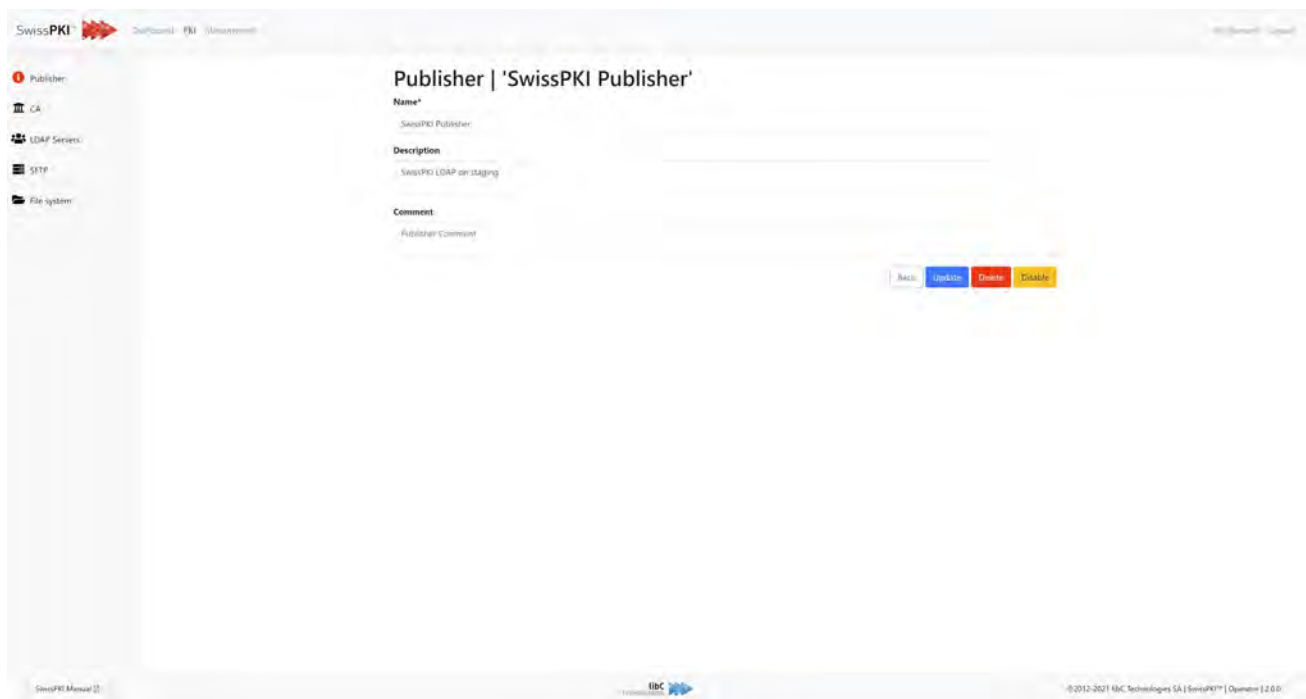
Created	Modified	Issuer	Serial#	Start Validity	End Validity	Actions
28.11.21	28.11.21	Microsoft Issuing CA I	7A12AE2490FA3AFF06C97D662440DE481AD7FC3A	2021-10-24T08:57:14Z	2024-10-24T08:57:14Z	

Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom of the table area, there are navigation buttons: 'Previous', '1', 'Next', 'Back', and 'Update'. Below the table is a light blue box with the text 'Add authentication certificates' and a large empty input field. The footer of the page includes 'SwissPKI Manual', the libC TECHNOLOGIES logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.3.2.7 Publisher

A publisher broadcasts certificates and/or CRL/ARL event to the publication end points registered with the service. When publication is enabled for client certificate issuance or when a CA is registered with the publisher, then the produced certificates and/or CRLs are written to the end points.

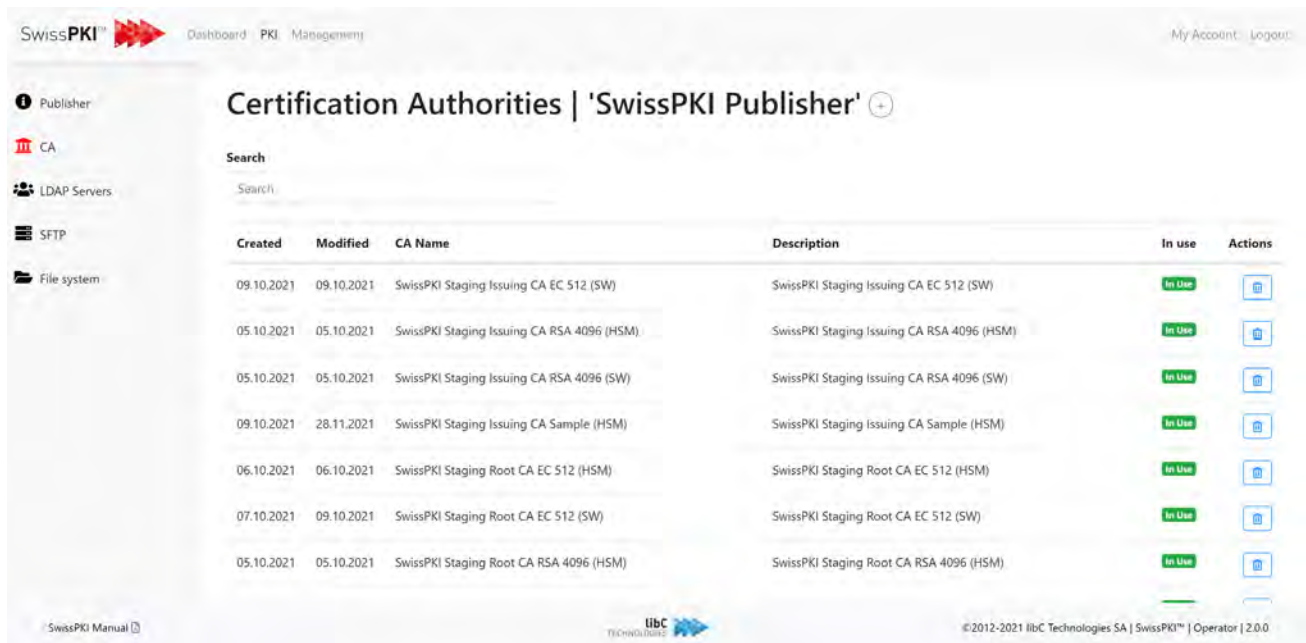
You can register Certification Authority with multiple Publisher services.










Action	Description
<b>Update</b>	Update the information fields
<b>Delete</b>	Deletes the Publisher service
<b>Disable</b>	Enable/disable the service. When disabled, the service does not process requests

### 12.3.2.7.1.1 Certification Authorities

Add or remove a Certification Authority to the Publisher Service.



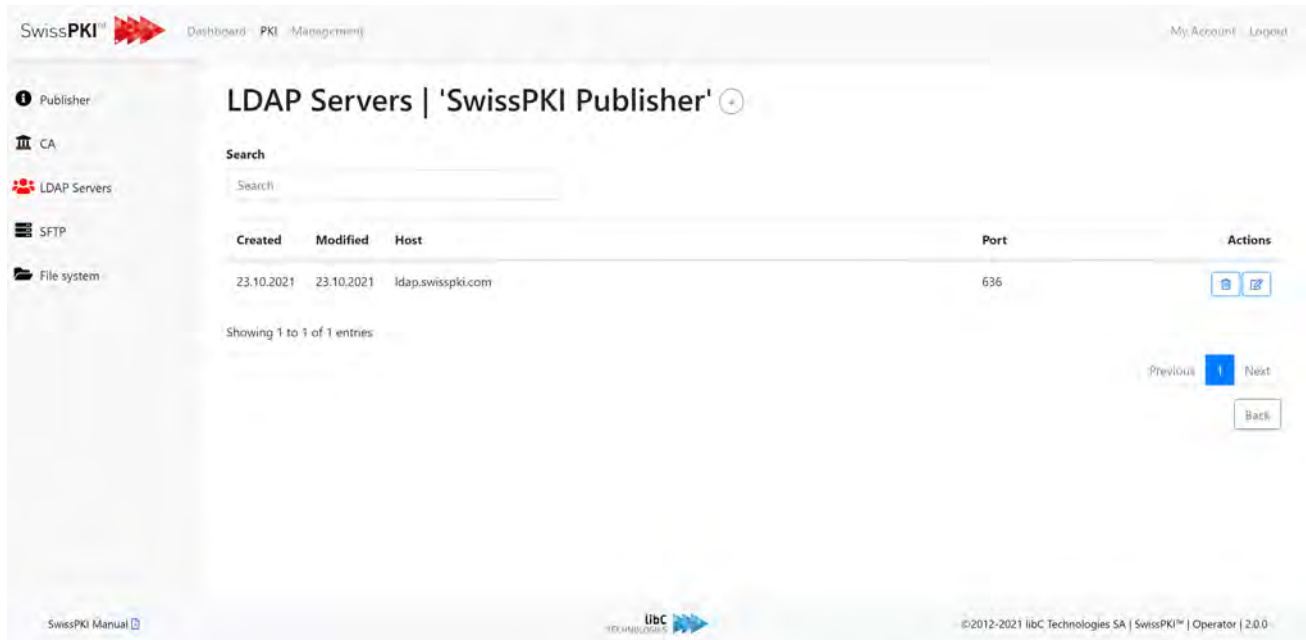
The screenshot shows the 'Certification Authorities' management page in the SwissPKI interface. The page title is 'Certification Authorities | 'SwissPKI Publisher'' with a plus icon for adding new entries. A search bar is located below the title. The main content is a table listing existing certification authorities.

Created	Modified	CA Name	Description	In use	Actions
09.10.2021	09.10.2021	SwissPKI Staging Issuing CA EC 512 (SW)	SwissPKI Staging Issuing CA EC 512 (SW)	In Use	
05.10.2021	05.10.2021	SwissPKI Staging Issuing CA RSA 4096 (HSM)	SwissPKI Staging Issuing CA RSA 4096 (HSM)	In Use	
05.10.2021	05.10.2021	SwissPKI Staging Issuing CA RSA 4096 (SW)	SwissPKI Staging Issuing CA RSA 4096 (SW)	In Use	
09.10.2021	28.11.2021	SwissPKI Staging Issuing CA Sample (HSM)	SwissPKI Staging Issuing CA Sample (HSM)	In Use	
06.10.2021	06.10.2021	SwissPKI Staging Root CA EC 512 (HSM)	SwissPKI Staging Root CA EC 512 (HSM)	In Use	
07.10.2021	09.10.2021	SwissPKI Staging Root CA EC 512 (SW)	SwissPKI Staging Root CA EC 512 (SW)	In Use	
05.10.2021	05.10.2021	SwissPKI Staging Root CA RSA 4096 (HSM)	SwissPKI Staging Root CA RSA 4096 (HSM)	In Use	

At the bottom of the page, there is a footer with the libC Technologies logo, a link to the 'SwissPKI Manual', and copyright information: '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.3.2.7.1.2 LDAP Servers

Register one or more LDAP servers to publish issued certificates and CRLs/ARLs



The screenshot shows the 'LDAP Servers | 'SwissPKI Publisher'' page in the SwissPKI web interface. The page includes a search bar, a table with columns for 'Created', 'Modified', 'Host', 'Port', and 'Actions'. A single entry is listed with 'Created' and 'Modified' dates of 23.10.2021, 'Host' as ldap.swisspki.com, and 'Port' as 636. The 'Actions' column contains icons for delete and edit. Navigation buttons for 'Previous', 'Next', and 'Back' are visible at the bottom right of the table area. The footer contains the libC Technologies logo and copyright information: ©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0.

Create or editing an LDAP publication server requires following information for publication:

## LDAP Servers

**Host\***  **Port\***

**User\***  **Password\***

**LDAP Base DN\***

**User certificate publication RDN**

Publish unique end user certificate

**Machine certificate publication RDN**

Publish unique device certificate

Fields	Description
<b>Host</b>	LDAP host
<b>Port</b>	LDAP port
<b>User</b>	User (bind DN) for logging into the server. Requires RW access
<b>Password</b>	User password
<b>LDAP Base DN</b>	The BaseDN of the DIT tree.  If the published certificate is a CA, then the certificate along with CRL/ARL is published or replaced using the CA's Subject DN path in the LDAP. The object class of the CA entry MUST be of type <i>certificationAuthority</i> if the LDAP entry is already present. If the published CA certificate does not exist, the publisher creates the entry following the SubjectDN path of the certificate. Therefore, the CA certificates MUST contain an ending SubjectDN with the BaseDN.
<b>User certificate publication RDN</b>	RDN where to publish the certificates. If empty. Published to the root of the BaseDN
<b>Publish unique end user certificate</b>	Enable/disable check box to publish unique certificate entries in the LDAP
<b>User certificate publication RDN</b>	RDN where to publish the certificates. If empty. Published to the root of the BaseDN
<b>Publish unique device certificate</b>	Enable/disable check box to publish unique certificate entries in the LDAP

#### End user certificate publication attributes

Attributes	Values
<b>objectClass</b>	top person inetOrgPerson
<b>cn</b>	Subject DN's CN is present, otherwise certificate serial number
<b>surName</b>	CN












<b>givenName</b>	if present in Subject DN
<b>mail</b>	if present in Subject DN if present in SAN RFC 822
<b>organization</b>	if present in Subject DN
<b>organizationalUnit</b>	if present in Subject DN
<b>locality</b>	if present in Subject DN
<b>uid</b>	Certificate's serial number
<b>userCertificate;binary</b>	DER certificate

Example of LDAP publication for end user with two certificates

Name	Value	Type	Size
objectClass	top	OID	3
objectClass	person	OID	6
objectClass	organizationalPerson	OID	20
objectClass	inetOrgPerson	OID	13
cn	Sample End User	Directory String	15
sn	Sample End User	Directory String	15
mail	sample@swisspki.com	IAS String	19
o	Internet Widgits Pty Ltd	Directory String	24
uid	1E073BC9E467394070D0681E13DF43207CCDE71C	Directory String	40
uid	39143133A953A3CC737C6C2BF9DA310D8D19BEA0	Directory String	40
userCertificate;binary	Sample End User	Certificate	589
userCertificate;binary	Sample End User	Certificate	588

## CA certificate and CRL/ARL

Attributes	Values
<b>objectClass</b>	top applicationProcess certificationAuthority
<b>cn</b>	Subject DN's CN is present, otherwise certificate serial number
<b>organizationalUnit</b>	if present in Subject DN
<b>locality</b>	if present in Subject DN
<b>caCertificate;binary</b>	DER certificate
<b>certificateRevocationList;binary</b>	DER CRL
<b>authorityRevocationList;binary</b>	DER ARL

Name	Value	Type	Size 
 <b>objectClass</b>	<b>top</b>	<b>OID</b>	<b>3</b>
 <b>objectClass</b>	<b>applicationProcess</b>	<b>OID</b>	<b>18</b>
 <b>objectClass</b>	<b>certificationAuthority</b>	<b>OID</b>	<b>22</b>
 cn	Root CA EC I	Directory String	12
 ou	PKI	Directory String	3
 cACertificate...	Root CA EC I	Certificate	665
 certificateRe...	com, SwissPKI, PKI, Root CA EC I	Certificate List	361
 authorityRev...	com, SwissPKI, PKI, Root CA EC I	Certificate List	374

## Machine certificate

Attributes	Values
<b>objectClass</b>	top device pkiUser
<b>cn</b>	SAN DNS and/or IPs
<b>organization</b>	if present in Subject DN
<b>organizationalUnit</b>	if present in Subject DN
<b>locality</b>	if present in Subject DN
<b>serialNumber</b>	Certificate's serial number
<b>userCertificate;binary</b>	DER certificate

Name	Value	Type	Size	
objectClass	top	OID	3	
objectClass	device	OID	6	
objectClass	pkiUser	OID	7	
cn	server.dns.org	Directory String	14	
cn	last.dns.org	Directory String	12	
cn	other.dns.org	Directory String	13	
o	Internet Widgits Pty Ltd	Directory String	24	
serialNumber	5A903A58DF472732139DB2B17B0EDEE47FFB338B	Printable String	40	
userCertificate;binary	server.dns.org	Certificate	612	

### 12.3.2.7.1.3 Unique certificate publication

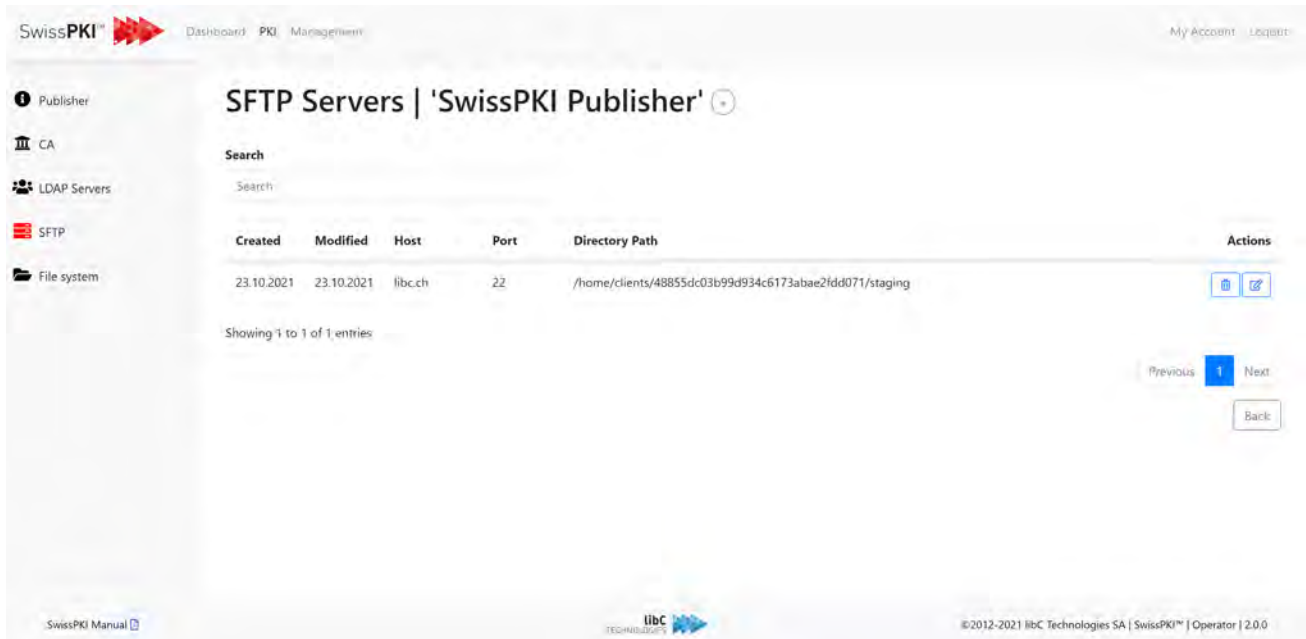
When enabled, the unique certificate publication creates a single LDAP object per published certificate. The LDAP object RDN is created with *<BaseDN>,uid='certificate serial number.'*

When disabled, the certificate publication creates LDAP object using the certificate's common name as the CN. The LDAP object RDN is created with *<BaseDN>,cn='certificate subject common name.'* An LDAP entry may have multiple certificates published for the object.



**Note:** when no certificate Subject CN is present, the certificate serial number is used.

### 12.3.2.7.1.4 SFTP Servers

Register one or more SFTP servers to publish issued certificates and CRLs/ARLs. Published certificates and CRL are published using the serial number with the extension *.cer*, *.crl* or *.arl* in DER format to the SFTP servers. CRL and ARL are prefixed with the CDP name.



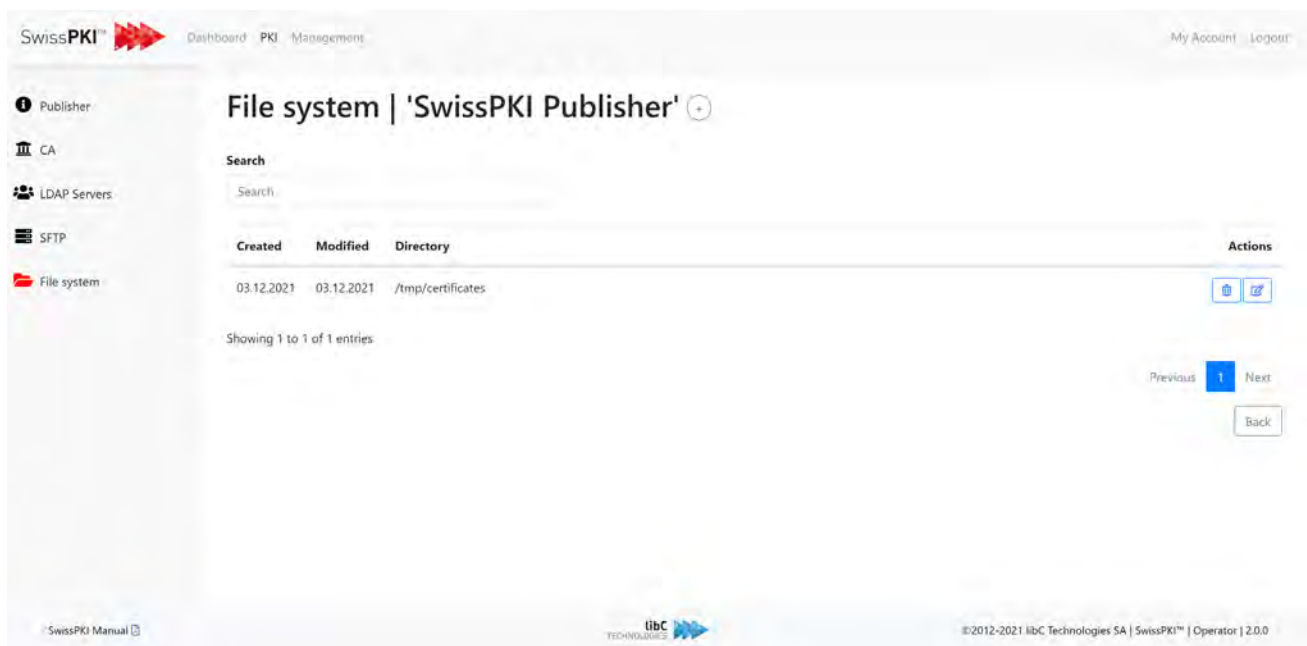
The screenshot shows the SwissPKI web interface. The top navigation bar includes 'SwissPKI', 'Dashboard', 'PKI Management', 'My Account', and 'Logout'. The left sidebar contains navigation links for 'Publisher', 'CA', 'LDAP Servers', 'SFTP', and 'File system'. The main content area is titled 'SFTP Servers | 'SwissPKI Publisher'' and features a search bar. Below the search bar is a table with the following data:

Created	Modified	Host	Port	Directory Path	Actions
23.10.2021	23.10.2021	libc.ch	22	/home/clients/48855dc03b99d934c6173abae2fdd071/staging	 

Below the table, it says 'Showing 1 to 1 of 1 entries'. At the bottom right of the table area, there are navigation buttons: 'Previous', '1' (selected), 'Next', and 'Back'. The footer of the interface includes 'SwissPKI Manual', the libC Technologies logo, and the copyright notice '©2012-2021 libC Technologies SA | SwissPKI™ | Operator | 2.0.0'.

### 12.3.2.7.1.5 File system

Register one or more file system directory on the server running the SwissPKI process to publish issued certificates and CRLs/ARLs. Published certificates and CRL are published using the serial number with the extension *.cer*, *.crl* or *.arl* in DER format to the SFTP servers. CRL and ARL are prefixed with the CDP name.



The screenshot shows the SwissPKI web interface. The main heading is "File system | 'SwissPKI Publisher'". Below this is a search bar and a table with the following columns: "Created", "Modified", "Directory", and "Actions".

Created	Modified	Directory	Actions
03.12.2021	03.12.2021	/tmp/certificates	[Delete] [Refresh]

Below the table, it says "Showing 1 to 1 of 1 entries". There are navigation buttons for "Previous", "Next", and "Back".

### 13 Auditor UI

Auditors are PKI users who have been assigned the AUDITOR role. They are authorized to access the audit log.

Please refer to section *12.2.9 Events*

Filter	Description
<b>Range (between)</b>	Allows you to display all the events that occurred in a desired date range.
<b>Source</b>	Allows you to filter the events by source. Thirteen sources are available: <ul style="list-style-type: none"> <li>• Issuance</li> <li>• TSA</li> <li>• DSS</li> <li>• OCSP</li> <li>• CMP</li> <li>• CRL</li> <li>• Admin</li> <li>• Authorization</li> <li>• Renewal</li> <li>• Recovery</li> <li>• Revocation</li> <li>• Publisher</li> <li>• Login</li> </ul>
<b>Type</b>	Allows you to filter the events by type. Three event types are available: <ul style="list-style-type: none"> <li>• Info</li> <li>• Warning</li> <li>• Error</li> </ul>
<b>Serial Number</b>	Allows you to display all events related to a given serial number
<b>Subject CN</b>	Allows you to display all events related to a subject common name.
<b>Order Id</b>	Allows you to display all events related to an Order Id

## 14 Registration UI

Please refer to the *'SwissPKI RA User Manual 2.0.pdf'*

## 15 SCION

You use the API to integrate Anapaya SCION Control Services with SwissPKI which acts as RA Operator towards the adapter

### 15.1 Protocol Adapter Responsibilities

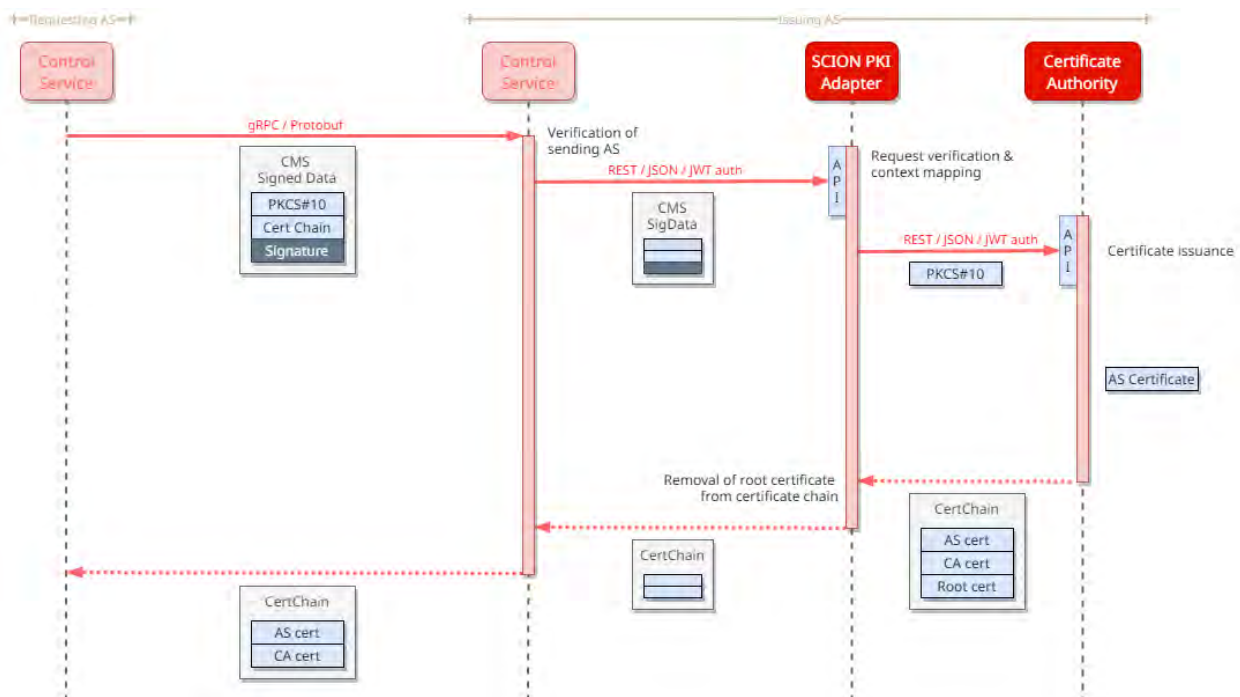
- Provision of an easily integrable interface for managing the PKI relates business processes of the SCION Control Services
- Mapping of the requests to the corresponding PKI context and its client environment.

### 15.2 Business Processes

#### 15.2.1 AS Certificate Renewal

##### 15.2.1.1 Overview

The AS certificate renewal process allows any associated Autonomous System to automatically renew its expiring AS certificate by sending a renewal request to the Control Service of the Issuing AS. The Control Service forwards the renewal request to the PKI Adapter by sending it to the certificate renewal endpoint. The PKI Adapter forwards the request to the Certificate Authority for issuance. The Certificate Authority returns the AS Certificate to the PKI Adapter, which then removes the root certificate from the certificate chain and returns the updated certificate chain to the Control Service. The Control Service then returns the updated certificate chain to the Requesting AS.





The renewal request payload is an RFC 5652 CMS Signed Data message. The signed payload of the CMS message is the PKCS#10 certificate signing request (CSR) containing the public key to be certified. The signature of the CMS Signed Data message has been applied with the private key of the current AS certificate of the requesting AS. This way, the AS confirms to be the legitimate origin of the renewal request.

When receiving the request, the PKI Adapter performs a range of verification steps to ensure the request is correctly formed and the requester is the legitimate owner of the identity information. If all these verifications succeed, it forwards the request to the Certification Authority.

As a result, it received the issued certificate and returns it back to the requester via the SCION Control Service of the Issuing AS. The newly issued AS certificate is sent back together with the certificate of the CP CA instance that issued the AS certificate.

### 15.2.1.2 Service Endpoint

Endpoint	Method	Authentication
/ra/isds/{isd-number}/ases/{as-number}/certificates/renewal	POST	Required

### 15.2.1.3 Preconditions

The following pre-conditions must be met to allow a client AS to renew its AS certificate by using this service.

- The requester has a valid AS certificate to sign the renewal request.
- The transmitting SCION Control Service has a Client ID and secret to authenticate at the service and is known by the PKI Adapter and is trusted by the AS Organisation’s PKI Client. (Authentication must be performed before calling this endpoint. The Control Service must present a valid token in its service call.)
- The requesting AS Organization is correctly configured in PKI (Realm and Client) and within SCION PKI Adapter (ISD-AS number mapped to PKI context).
- The requesting AS Organization has a vetted identity information entry in the ID-Repository.

### 15.2.1.4 Post-Conditions

As a result, the requester will receive a new “regular” AS certificate and the certificate of the CP CA instance that signed the certificate. In case of a mismatch of any precondition or any service failure, the request will receive an error report with human readable information describing the problem.

## 15.2.1.5 Process Steps

### 15.2.1.5.1 Case correlation ID

For every request entering the certificate renewal Service endpoint, the Service generates a unique case correlation identifier (UUID) which will be added to any log message and error message sent back to the requester. This allows to track and analyses error situations and service failures.

### 15.2.1.5.2 Service Request verification

The first step after the request was received is to verify that the PKI Adapter instance oversees managing it. If this verification step fails, the process will be stopped, and the request is responded with an error message.

The following request verification steps will be executed:

- Verification that the ISD Number of the request (sent as an URL parameter) is known by the PKI Adapter configuration.
- Verification that the AS Number of the request (also sent as an URL parameter) in combination with the ISD Number matches an AS known by the PKI Adapter configuration.
- In addition, the AS referred to must be in status “active.”

The verification of the ISD- and AS-Number and its status is solely done against the PKI Adapter’s configuration file.

### 15.2.1.5.3 Request Payload validation

After the verification of the request parameters, the request payload will be validated.

### 15.2.1.5.4 Payload Structure

The payload consists of an RFC 5652 CMS Signed Data message conforming to Section 5 of the RFC. It contains the following information in its structure:

- Digest Algorithm: sha-256 or sha-512
- Encapsulated Content Info:
  - eContentType: PKCS10 OID: 1.2.840.113549.1.10 or just 1.2.840.113549.1.7.1 (Data) or
  - eContent: The self-signed PKCS#10 CSR to be certified
- Certificates: Set of certificates to be used to verify the message signature. This set consists of the following certificates:
  - AS certificate whose private key was used to sign the CMS message
  - Issuing CP CA certificate whose private key was used to issue the AS certificate.
- CRLs: Empty. This field is not used.
- Signer Info: As the CMS message is signed by only the requester AS key, there must be only one Singer Info in the set.

#### 15.2.1.5.5 Signature verification

The signature certificate (AS certificate) contained on the certificates set and referred to in the Signer Info will be validated in the following way:

- It must not be expired at the time of request verification.
- It must have the characteristics of an AS certificate. (ISD-AS Number in the Subject DN, Key Usage “digitalSignature” is set, Key Usage “keyCertSign” is not set, Extended Key Usage “timestamping” is set.)
- It is SubjectDN ISD-AS Number attribute must match the ISD-AS Number of the renewal request (service endpoint call).
- Its certificate chain must validate up to a CP Root CA certificate which is either an own Root instance or a Root instance explicitly trusted by the PKI (SwissPKI Realm).
- If it is an AS certificate issued by an own Certification Authority, the revocation status is checked, and it must not be revoked.

#### 15.2.1.5.6 PKCS10 Request verification

After the successful verification of the CMS Signed Data message and its signature, the PKCS#10 CSR sent as the CMS message signed payload is verified in the following way:

- The content must be a valid PKCS#10 data structure.
- Its self-signature must validate
- The ISD-AS Number attribute of the SubjectDN must match the ISD-AS Number of the service request and the signature (AS) certificate.

#### 15.2.1.5.7 Identity validation

##### 15.2.1.5.7.1 Process Steps

- Verification of Protocol Buffer SignedMessage’s Signature
  - Policy: Signature must have been applied with a former AS certificate key of the requesting AS
- Extraction and verification of PKCS#10 CSR
  - Policy: ISD-AS Number of Subject DN in CSR must match the ISD-AS Number in the Subject DN of the AS certificate which was used to verify the signature of the Protocol Buffer Signed Message
- Put request into right context
  - Identify SwissPKI RA Operator Account for the given ISD-AS Number.
  - Identify SwissPKI realm / client environment of requesting tenant.
  - Identify SwissPKI Certification Policy with which the renewal will be performed.
- Renew certificate with SwissPKI
  - Authenticate against SwissPKI API by using configured RA Operator credentials.
  - Submit PKCS#10 CSR with required context information and receive the signed AS certificate with chain
  - Return certificate chain containing the new AS certificate

## 15.2.2 Frontend API

### 15.2.2.1 Use Cases

In the first phase, the API supports an individual use case: the renewal of AS certificates in a direct communication of the AS Control Services to the Issuing AS by using the SCION protocol stack.

### 15.2.2.2 Authentication and Authorization

The API uses JSON Web Token (JWT) for user authorization. Authentication is implemented via a Client ID and a Shared Secret. Each user of the API (Issuing AS Control Service) gets a unique Client ID and a Shared Secret generated by the PKI administrators. The Client ID used is the ID of a PKI Registration Authority Operator (RAO) user. The Shared Secret is an API Key generated for that user.

#### 15.2.2.2.1 Credentials

Element	Type / Value	Validity
Client ID	String / RAO user name	Lifetime
Shared Secret	String / API Key of RAO user	Lifetime. Can be renewed in the PKI. Renewed APKI Keys expire within 7 days.

#### 15.2.2.2.2 Token reception

The JWT Token can be fetched by calling the `/auth/token` service endpoint with the POST method and providing the required credentials in the JSON request body. See the technical API specification for further details.

The issued token will expire after the lifetime defined by the PKI Adapter. The Client will have to request a new token again by calling the `/auth/token` service path.

#### 15.2.2.2.3 Token self-generation

Alternatively, the Control Service can build and sign the JWT Token itself and send it with every request requiring authentication. To do this, the control service places its client ID into the subject claim and the issuer claim of the token and provides the token with a HMAC256 signature applied by using the shared secret. (HMAC384 and HMAC512 are also possible.)

Self-issued token must respect the lifetime allowed by the definitions of the Control Service.

## Header

Header attribute	Value(s)	Description
<b>alg</b>	"HS256", "HS384", "HS512"	Signature Algorithm.
<b>type</b>	"JWT"	Token Type. Always JSON Web Token

## Payload

Payload claims	Value(s)	Description
<b>sub</b>	"<ClientId>"	Client Identifier as used to authenticate on the /auth/token path
<b>iss</b>	"<IssuerId>"	Issuer of the token. PKI Adapter identifier, or ClientId for self-issued token.
<b>iat</b>	Timestamp (seconds)	Issued At timestamp as defined in RFC 7519, Section 4.1.6
<b>nbf</b>	Timestamp (seconds)	Not Before timestamp as defined in RFC 7519, Section 4.1.5
<b>exp</b>	(iat + tokenLifetime)	Expiration time as defined in RFC 7519, Section 4.1.4  tokenLifetime is the default lifetime of a token. The maximum value accepted by the PKI Adapter can be queried by accessing the /auth/token path with the GET method. (See below.)
<b>scope</b>	"ra" and/or "ca"	Optional. List of scopes permitted to access.

### 15.2.2.2.4 Token presentation

The Client must present the JWT token for all paths requiring authentication by sending it in the according Authorization HTTP Header.

```
Authorization: Bearer <token>
```

#### 15.2.2.2.5 Token and account verification

The PKI Adapter validates the JWT token presented in the request header in the following way:

##### Token verification

- The token must not have expired
- The token must not have a lifetime longer than permitted by the adapter's configuration. The lifetime is calculated from the `exp` and `iat` claims.

##### Account verification

- The token's subject claim must match with the identifier of a configured SCION Control Service account configuration.
- The token's signature must validate with the shared secret of the given subject.
- The token's scope contains the value of the scope required by the endpoint.

## 15.2.3 API Reference

### 15.2.3.1.1 Overview

The API is based on the CA Integration API specified by Anapaya <sup>27</sup> and has the following basic path's structure.

Path	
<b>/auth</b>	Authentication operations.
<b>/auth/token</b>	JWT token auth specific operations.
<b>/healthcheck</b>	Endpoint to evaluate the availability of the service
<b>/ra/...</b>	Operations in context of Registration Authority services. Accessed by the Control Service of the Issuing AS.
<b>/ra/isds/</b>	RA Operations on Isolation Domains.
<b>/ra/isds/{isd-number}</b>	RA Operations on a specific Isolation Domain.
<b>/ra/isds/{isd-number}/ases</b>	Operations on Autonomous Systems of an ISD.
<b>/ra/isds/{isd-number}/ases/{as-number}</b>	Operations on a specific Autonomous System of an ISD.
<b>/ra/isds/{isd-number}/ases/{as-number}/certificates/...</b>	Operations on Certificates of an Autonomous System.
<b>/ca/...</b>	Operations in context of Certification Authority services. E.g., accessed by the SSFN service management platform to enroll a new participant / tenant to the CA.  Specification of this part of the API will be subject of a later project phase.
<b>/ca</b>	Reserved

<sup>27</sup> <https://gist.github.com/Oncilla/2f7eb2a9b142a58b82596a02980d4749>

In the current stage of implementation, not all paths are provided with service endpoints. The available service paths are documented in the following sections.

### 15.2.3.1.2 Authentication

A client can authenticate itself at the `/auth/token` endpoint by using the POST method.

Path	<code>/auth/token</code>	
<b>Method</b>	POST	
<b>Request Parameters</b>	None, parameters are sent in the request body.	
<b>Request Body</b>	JSON object containing client identifier and shared secret to authenticate the client. <ul style="list-style-type: none"> <li>- Client identifier string</li> <li>- Shared secret</li> </ul> <pre>{   "client_id": "string",   "client_secret": "string" }</pre>	
<b>Responses</b>	<b>Status Code</b>	<b>Content</b>
	200	JSON object containing the access token and some meta information. <pre>{   "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ5In0",   "token_type": "Bearer",   "expires_in": 3600 }</pre>
	400	RFC 7807 Problem Details JSON object
	401	RFC 7807 Problem Details JSON object



	500	RFC 7807 Problem Details JSON object
	503	RFC 7807 Problem Details JSON object
<b>Authentication</b>	none	

In addition to Anapaya’s specification, the /auth/token endpoint also supports the GET method to let the client ask for the value of the max. allowed token lifetime in seconds for self-issued tokens. The value is presented in the expires\_in attribute of the reply.

Path	/auth/token	
<b>Method</b>	GET	
<b>Request Parameters</b>	None	
<b>Request Body</b>	-	
<b>Responses</b>	Status Code	Content
	200	JSON object containing meta information allowing the client to generate self-issued tokens.  <pre>{   "token_type": "Bearer",   "expires_in": 3600 }</pre>
	400	RFC 7807 Problem Details JSON object
	401	RFC 7807 Problem Details JSON object
	500	RFC 7807 Problem Details JSON object
	503	RFC 7807 Problem Details JSON object
<b>Authentication</b>	none	

### 15.2.3.1.3 Renewal of an AS-Certificate

Path	/ra/isds/{isd-number}/ases/{as-number}/certificates/renewal		
Method	POST		
Request Parameters	Parameter	Type	Content
	isd-number	Path	Integer. ISD Number of AS to be renewed
	as-number	Path	String, URL encoded. AS Number of AS to be renewed (URL encoded)
Request Body	<p>JSON object containing the Certificate Renewal Request as Base64 encoded RFC 5652 CMS Signed Data with the following fields:</p> <ul style="list-style-type: none"> <li>- certificates: Set of certificates to build the trust chain of the key used to sign the CMS Signed Data.</li> <li>- encapContentInfo: eContentType set to id-data, and the PKCS#10 CSR as the eContent payload.</li> </ul> <pre>{   "csr": "string" }</pre>		
Responses	Status Code	Content	
	200	<p>JSON object consisting of the following elements.</p> <ul style="list-style-type: none"> <li>- Certificate Chain containing the new AS certificate and it is CA certificate(s) encoded in a PKCS#7 data structure.</li> </ul> <pre>{   "certificate_chain": "string" }</pre>	
	400	RFC 7807 Problem Details JSON object	
	401	RFC 7807 Problem Details JSON object	

	404	RFC 7807 Problem Details JSON object
	500	RFC 7807 Problem Details JSON object
	503	RFC 7807 Problem Details JSON object
<b>Authentication</b>	JWT token	

#### 15.2.3.1.4 Healthcheck

<b>Path</b>	/healthcheck	
<b>Method</b>	GET	
<b>Request Parameters</b>	none	
<b>Responses</b>	<b>Status Code</b>	<b>Content</b>
	200	JSON object with information about the service status. Allowed values for status are “available,” “starting,” “stopping,” “unavailable.”  <pre>{   "status": "available" }</pre>
	500	RFC 7807 Problem Details JSON object
	503	RFC 7807 Problem Details JSON object
<b>Authentication</b>	none	

### 15.2.3.1.5 Error Codes

### 15.2.3.1.6 HTTP Status Codes

The following HTTP status codes can be returned in case of an error by the calling client or service failure.

Code	Description	Usage
400	Bad request	Upon incorrectly formed or missing request parameters or required request body elements.
401	Unauthorized	Send if a protected resource was called without valid authorization. Missing or expired JWT token.
404	Not Found	If a requested resource cannot be found. A resource denoted by a request parameter or request body information.
500	Internal Server Error	Unexpected server failure.
503	Service unavailable	In case the service is intentionally not reachable. (E.g., due to maintenance work.)

#### 15.2.3.1.6.1 Application Error Codes

The error responses listed above send a JSON object in the response body to inform the client about the error source(s). The object returned is a [RFC 7807](#) Problem Details Object.

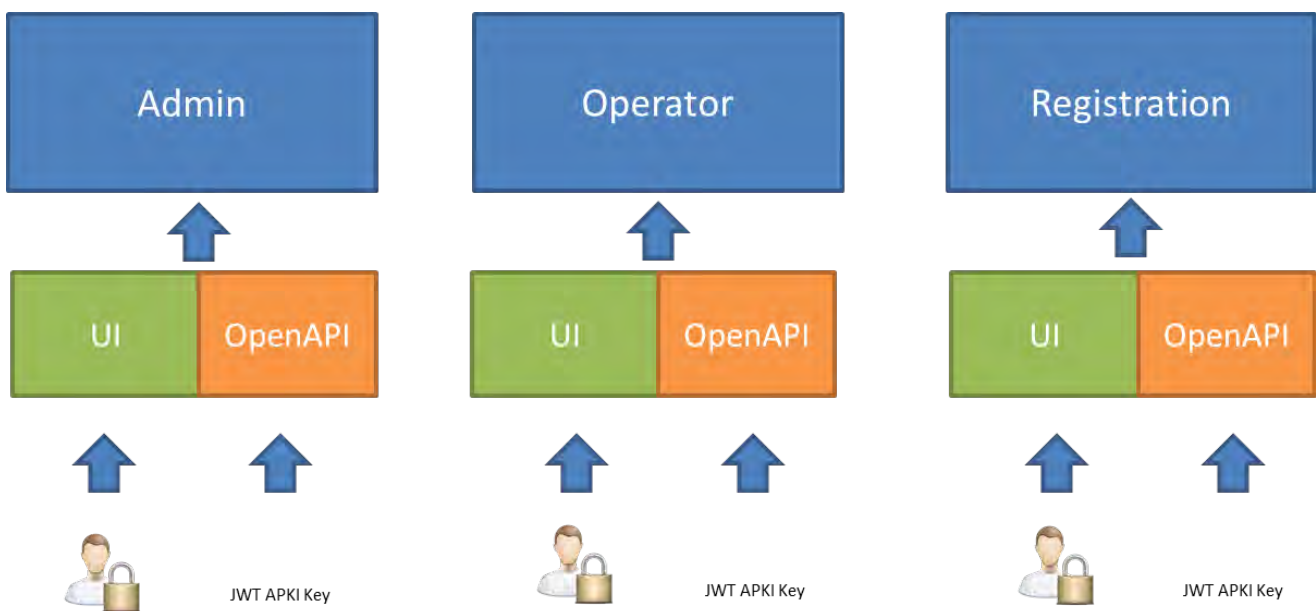
```
{
  "type": "/problem/policy-violation",
  "title": "Policy Violation",
  "status": 400,
  "detail": "The CSR sent violates the certification policy of the issuing CA. Field CN must not contain underscore characters.",
  "instance": "/problem/policy-violation#cn-malformed",
  "correlation_id": "7D99A76F-9BA5-4C15-A1C9-114A54D9B1F1"
}
```

Please note that according to RFC 7807, Problem Details Objects are returned as content type `application/problem+json`.

## 16 REST API

SwissPKI offers an OpenAPI specification for automating and integrating your PKI with your services. As a developer, you have programmatic access to the SwissPKI via web services. You use an OpenAPI generator to produce the client API for a specific programming language <sup>28</sup>.

All operations you achieve through the user interface is accessible through the API. There are three sets of APIs available:



1. PKI Administration API
2. Operator API
3. Registration API

The PKI Administration API lets you manage the global settings of the SwissPKI deployment and the its Realms, including Realm settings and associated CA Operators.

The Operator API lets you manage the PKI Entities within a Realm and their associated clients, Registration Officers, Authorizer and Auditors including certificate policy templates and certificate products.

The Registration API enables you to register, revoke and search certificates as well as authorize registration requests.

<sup>28</sup> SwissPKI OpenAPI is tested with the openapi-generator from <https://github.com/OpenAPITools/openapi-generator>

## 16.1 Roles and Permissions

For a given user and role using the client API, the same Roles and Permissions apply as the ones specified in the user interface. That is, if a given user and role is authorized to fulfill a READ operation via the Web UI, then the same operation is accessible through the generated client API. If a DELETE permission is withdrawn from a specific user and role for a specific operation, then the DELETE permission is correspondingly withdrawn from the client API operation.

To obtain an API Key, the user role must at least have the permissions *ACCOUNT\_API\_KEY* View and Create associated to its user account for the specified role. Additionally, the Update and Delete permissions enable the user to renew and/or delete its API Keys.

If the user role has no *ACCOUNT\_API\_KEY* permission enabled, it is still possible to issue an API Key to this user by a higher role if permission is granted.

Additionally, if a user is of type *SERVIE ACCOUNT*, then the user can use the API but not log in to the Web UI.

### Rules:

A PKI Administrator role can manage CA Operator API Keys if permission is granted

A CA Operator role can manage RA Officer, Authorizer and Auditor API Keys if permission is granted

A user can manage its own API Keys if permission is granted

## 16.2 API Key

To use the API, a user must obtain an API Key.

A user with multiple roles has one or more API Keys.

The API Key is an auto generated 64 bytes shared secret using digit, alpha, upper, and lowercase and is used on the client side (API) to generate a signed (HMAC-256) JW Token.

### 16.2.1 API Key Rollover

Generated API Keys are immediately available to the client and have no expiration date and time set.

Deleting an API Key prevents immediately access to the Web Services.

When an APKI Key is updated, a new API Key is generated, and the previous API Key is valid for another 7 days. The user has maximum 7 days to replace the API Key on its deployment (client configuration)

## 16.3 Authentication

Generate a JW Token (JWT) and signing it with the API Key using HMAC256 as 'text/plain'. By default, a JW Token is valid for 8 hours.

### 16.3.1 JWT Generation

The JWT must include:

Claim	Value
<b>iss</b>	SwissPKI
<b>aud</b>	REST API
<b>sub</b>	<UserName> of the SwissPKI account
<b>iat</b>	Normalized UTC date/time
<b>nbf</b>	Normalized UTC date/time
<b>exp</b>	Normalized UTC date/time

### 16.3.2 HTTP Request

Using HTTP requests to access the SwissPKI web services, include in each request the following HTTP header, where encoded JWT is the signed encoded token:

```
Authorization: Bearer <encoded JWT>
```

Using generated Java client API with the `openapi-generator`, set the encoded JWT as follow:

```
HttpBearerAuth bearerAuth =
(HttpBearerAuth)defaultClient.getAuthentication("bearerAuth");
bearerAuth.setBearerToken("encoded JWT").
```

Each service request MUST include the JWT token. The PKI web services do not return a usable session cookie.

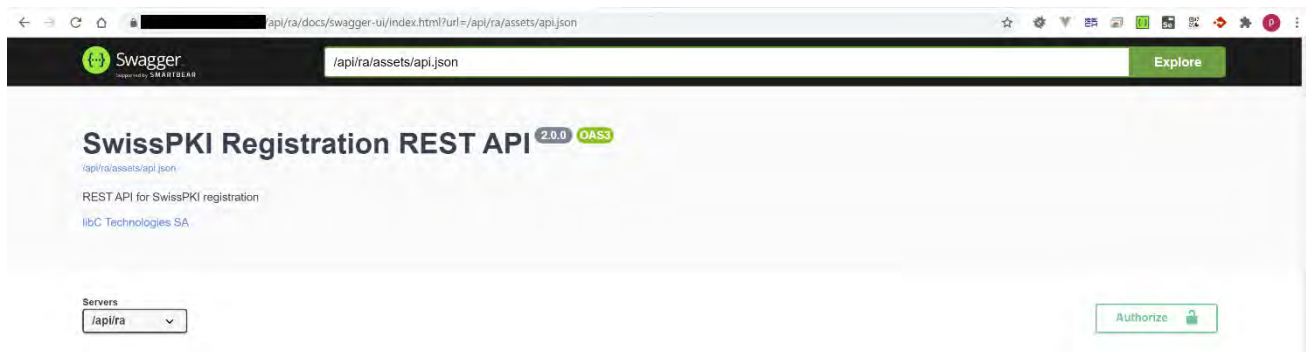
An SSL protected helper method is available to you for generating your JW Token:

```
GET /pki/api/v2/jwt/:userName/:key
```

Where `:userName` is your user account and `:key` your user account API Key which is available from the Web UI under 'My Account' menu.

### 16.3.3 OpenAPI v3 specification

A Swagger UI is packaged with each API module to allow your development team or your end consumers to visualize and interact with the API's resources without having any of the implementation logic in place. The specification is automatically generated with the visual documentation making it easy for back end implementation and client side consumption.



Specification	URL Location
Administration API	https://<deployed DNS/IP>/api/admin/
Operator API	https://<deployed DNS/IP>/api/operator/
Registration API	https://<deployed DNS/IP>/api/ra/



## 17 Migrating SwissPKI v1 to SwissPKI v2

The migration procedure from SwissPKI v1 to v2 involves dumping the database from the SwissPKI version 1 (MariaDB) and loading the dump to the SwissPKI version 2 (PostgreSQL).

### 17.1 Requirements

The SwissPKI version 1 deployment must be the latest revision 1.3.2839.

The SwissPKI version 2 deployment must be revision 2.0.213 or higher

### 17.2 Procedure

The steps to migrate from the latest SwissPKI version 1 to SwissPKI version 2 are:

1. Prepare the migration
  - a. Install SwissPKI version 2 and configure all deployment settings.
  - b. Start SwissPKI version 2 but do not perform the initial setup (see *9 Initializing SwissPKI*). This will generate a SwissPKI version 2 database schema with all necessary empty database tables.
  - c. Take SwissPKI version 1 offline by
    - i. disabling access to the services for issuing certificates. Services such as OSCP and TSA can stay online.
    - ii. disabling automatic CRL generation
    - iii. generating manually a CRL for each CA
    - iv. Backing up the SwissPKI version 1 MariaDB database
2. Obtain the Dump & Load command line scripts (ZIP file) from [support@swisspki.com](mailto:support@swisspki.com) if you plan to execute the migration from a Linux machine (unzipping and running the command lines from a shell) or download the Docker images from [nexus.libc.ch](https://nexus.libc.ch) if you plan to run the migration from a Kubernetes POD. Contact [support@swisspki.com](mailto:support@swisspki.com) to obtain access to the Dump & Load Docker images.

3. Execute the *'dump'* script on the SwissPKI version 1 MariaDB database
  - a. Navigate to the *'export\_db'* folder
  - b. Configure the following settings in the *'conf.xml'* file and verify that you have TCP access to the MariaDB server instance:

Key	Description
<b>dataPath</b>	Where to dump the data (ends with <i>"/</i> )
<b>driver</b>	The fully qualified class name of the driver to use. Example: <i>org.mariadb.jdbc.Driver</i>  For SSL configuration, please refer to <a href="https://mariadb.com/docs/connect/programming-languages/java/tls/">https://mariadb.com/docs/connect/programming-languages/java/tls/</a>
<b>url</b>	The jdbc string used to connect to the database.
<b>user</b>	Database user
<b>password</b>	Database password

- c. Run the *'export\_db.sh'* script.
- d. Logs are written in the *'all.log'* file.

4. Execute the 'load' script to import the dumped data from SwissPKI version 1 to the SwissPKI version 2 database (PostgreSQL)
  - a. Navigate to the 'import\_db' folder.
  - b. Configure the following settings in the 'conf.xml' file:

Key	Description
<b>dataPath</b>	Path to stored data (ends with "/")
<b>driver</b>	The fully qualified class name of the driver to use. Example: <i>org.postgresql.Driver</i>  For SSL configuration, please refer to <a href="https://jdbc.postgresql.org/documentation/head/ssl-client.html">https://jdbc.postgresql.org/documentation/head/ssl-client.html</a>
<b>url</b>	The jdbc string used to connect to the database.
<b>user</b>	Database user
<b>password</b>	Database password
<b>pbeSaltV1<sup>29</sup></b>	SwissPKI version 1 salt. (value of <i>swisspki.pbe.salt</i> from the <i>swisspki.conf</i> file)
<b>serverKeyV1</b>	SwissPKI version 1 secret key. (value of <i>play.http.secret.key</i> from <i>application.conf</i> file)
<b>pbeSaltV2</b>	SwissPKI version 2 salt. (see <i>swiss.pki.secret.salt</i> in the SwissPKI v2 deployment manual)
<b>serverKeyV2</b>	SwissPKI version 2 secret key. (see <i>swiss.pki.secret.key</i> in the SwissPKI v2 deployment manual)

- c. Run the 'import\_db.sh' script.
- d. Logs are written in the 'all.log' file.

---

<sup>29</sup> The AES cipher operations in SwissPKI version 2 using the AES session key derived from the salt and secret has changed to support the latest AES256 GCM cipher/decipher mode. The data ciphered in SwissPKI version 1 are using an AES256/CBC/PKCS5 cipher/decipher mode. The data migrated from SwissPKI version 1 to SwissPKI version are deciphered and re-ciphered during the migration process using the new cipher mode.

5. Take SwissPKI version 2 online by changing the DNS entries to point to the new SwissPKI version 2 deployment
6. Take SwissPKI v1 offline.

### 17.2.1 Changes in TOTP length

Username/password with TOTP login in SwissPKI version 2 uses 6 digits TOTP tokens whereas SwissPKI version 1 uses 8 digits TOTP tokens.

If you wish to continue using 8 digits TOTP tokens in SwissPKI version 2, then enable *allowV1Codes* in *authentication.conf* (please refer to the SwissPKI version 2 deployment manual).

If you want your users to use 6 digits TOTP codes, then users can login with a scratch code and generate a new TOTP from My Account -> TOTP when logged in in the Administrator, Operator or RA Web UI. An email with a new TOTP and scratch codes is sent to the user.

## 17.2.2 Changes in Notification Tags

Although most of the notification tags in the v1 are available in the v2, some of them have been removed. The table below lists all the removed tags. In the case some of the following tags are used, consider updating the notification messages.

Class	Tag	Comment / Change
<b>Certificate</b>	.generatedKeyRefId	Removed in version 2.
	.subjectCN	Change to 'subject.'
<b>CertificateOrderAuthorization</b>	.createdBy.email	Will be replaced by 'createdBy.userName.' Consider updating the notification message.
	.createdBy.firstName	
	.createdBy.language	
	.createdBy.lastname	
	.createdBy.phone	
	.createdBy.status	
	.createdBy.timemodified	
	.createdBy.userTitle	
	.createdBy.userType	
	.authorizedBy.email	Will be replaced by 'authorizedBy.userName.' Consider updating the notification message.
	.authorizedBy.firstName	
	.authorizedBy.language	
	.authorizedBy.lastname	
	.authorizedBy.phone	
	.authorizedBy.status	
	.authorizedBy.timemodified	
	.authorizedBy.userTitle	
	.authorizedBy.userType	
	.certificateRevocationInfoAuthorization.revocationReason	Removed in version 2.

	.certificateRevocationInfoAuthorization.timemodified	
<b>CertificateOrder</b>	.issuedBy.email	Will be replaced by 'issuedBy.userName.' Consider updating the notification message.
	.issuedBy.firstName	
	.issuedBy.userName	
	.issuedBy.language	
	.issuedBy.lastname	
	.issuedBy.phone	
	.issuedBy.status	
	.issuedBy.timemodified	
	.issuedBy.userTitle	
	.issuedBy.userType	
	.microsoftCertificateOrderRequest.tokenType	
.microsoftCertificateOrderRequest.username		
.policyGroupRulesACMEMapping		
<b>CertificateRenewalRule</b>	.notificationTemplate.description	Removed in version 2.
	.notificationTemplate.timemodified	
	.notificationTemplate.type	
<b>CertificationAuthority</b>	.community.comment	Removed in version 2.
	.community.description	
	.community.name	
	.community.timemodified	
<b>RenewalRule</b>	.notificationTemplate.description	Removed in version 2.
	.notificationTemplate.timemodified	
	.notificationTemplate.type	
<b>PrimusConfiguration</b>	.timemodified	Not Available in version 2
	.user	
	.host	

<b>PrimusConfigurat ionElement</b>	.lastStatusCheck	
	.port	
	.statusMessage	
	.timemodified	

### 17.3 Microsoft Policy Mappings

The architecture for the Microsoft CEP/CES has changed from SwissPKI version 1 to SwissPKI version 2. It is now possible to map one Issuing CA to multiple Microsoft AD domains and assign selected certificate policy templates for each Microsoft AD Domain.

Microsoft CES/CEP certificate policy templates must be mapped manually.

1. Log into the Operator UI and select to PKI tab.
2. Select a Microsoft CA > Microsoft Policies.
3. Add all wanted Microsoft policies using the '+' button.
4. Repeat from step 2 for all Microsoft CAs.

Please contact [support@swisspki.com](mailto:support@swisspki.com) or Professional Services [info@libc.ch](mailto:info@libc.ch) for the migration scripts and migration documentation.