Virtual appliance quickguide and admin manual

Company	KeyTalk
Original author	MR van der Sman
Last edited by	MR van der Sman
Creation date	09 May 2017
Last update date	05 May 2024
Product	KeyTalk virtual appliance: certificate and key lifecycle management and enrollment (CLM/CKMS/KMS), PKI Management
Data classification	Public
Software/firmware version	7.4.8
Manual version	7.4.8.1

Contents

1. Quickguide Introduction and download 6

2. Quickguide Resource requirements 6

3. Quickguide Trusted secure client-server communication 7

4. Quickguide high level network setup non-HA 8

5. Quickguide high level network setup High Availability 9

6. Quickguide local system admin login.... 10

7. Quickguide manually changing IP addresses & default gateway 11

8. Quickguide: Getting started.... 12

Step 1: Change your local system admin and CLI root password 12

Step 2: Ensure your KeyTalk license is valid, when it's not, upload one as provided to you:.... 12

Step 3: Ensure all daemons are running 13

Step 4: Configure your network.... 13

Step 5: Generate your unique private Certificate Authority tree and keys.... 14

Step 6: Add one or more certificate templates in TEMPLATES.... 15

Step 8: Set seat hardware recognition/pre-shared-secret settings.... 15

Step 9: Connect a Registration Authority source to your TEMPLATE 17

Step 10: Create KeyTalk agent configuration file.... 18

Step 11: (Periodically) Backup your KeyTalk configuration settings.... 19

9. Quickguide: Frequently encountered setup issues 20

10. Introduction: KeyTalk certificate & key management and enrolment.... 22

11. Setting up your KeyTalk private Certificate Authority.... 23

11.1 HSM – Hardware Security module.... 27

11.1.1 Thales Cloud Data Protection on Demand Cloud HSM configuration 28

11.1.2 Thales Luna 7 HSM configuration with and without PED 29

11.1.3 Utimaco CryptoServer LAN and cloud HSM.... 30

12. Issuing certificates to end-points.... 32

12.1 KeyTalk TEMPLATE settings: certificate Registration Authority basic settings.... 33

12.2 KeyTalk TEMPLATE settings: your default certificate source, meta-data, and certificate policies 34

12.3 KeyTalk TEMPLATE settings: set optional TPM or Virtual Smart Card support 37

12.4 KeyTalk TEMPLATE settings: set optional LDAP address book and write certificates to AD 37

12.5 KeyTalk TEMPLATE settings: request and deploy S/MIME certificates to 3rd parties.... 38

12.6 Certificate Revocation List (CRL) and Distribution Point (CDP).... 41

12.7 KeyTalk TEMPLATE settings: trusted device slot assignment to end-points and API ... 43
12.8 Certificate renewals versus new certificate requests ... 44
12.9 About Common Name (CN) and Subject Alternative Name (SAN) in certificates ... 44
12.10 Supported CA's / TSPs ... 46
12.10.1 KeyTalk Private CA ... 46
12.10.2 DigiCert: DigiCert, GeoTrust, QuoVadis, RapidSSL, Thawte ... 47
12.10.3 EJBCA ... 51
12.10.4 GlobalSign: GCC and ATLAS ... 52
GlobalSign GCC ... 53
GlobalSign ATLAS ... 56
13. Registration Authority connectors: Authentication Modules ... 58
13.1 Internal KeyTalk Db, MySQL based ... 59
13.1.1 Enrollment for Machine Certificates ... 61
13.2 Active Directory / LDAP / Kerberos ... 62
13.2.1 Initial configuration steps: ... 63
13.2.2 Advanced configuration steps: LDAPS and Global Catalog ... 64
13.2.3 Advanced configuration steps: Password change support, attribute lookup and pre-enrolment ... 66
13.2.4 Advanced configuration steps: LDAP/AD clustering, Load Balancing ... 71
13.2.5 Advanced configuration steps: Kerberos based authentication ... 71
13.2.6 Replacing AD user/hostname with machinename value or static value in the certificate CN ... 72
13.2.7 Overwriting default certificate template values with Active Directory Attribute values ... 73
13.3 Azure Active Directory (AAD) ... 74
13.3.1 Add and configure a KeyTalk Authentication Azure Module connection ... 74
13.3.2 Configure Azure Active Directory for use with KeyTalk ... 75
13.3.3 Configure Azure Active Directory to enable attribute mapping to certificate fields ... 79
13.4 Radius: One Time Password tokens, Username/password, AKA SIM ... 82
13.5 MySQL ... 84
13.6 Other Identity Provider based authentication solutions ... 85
14. Certificate renewal automation parameters ... 86
14.1 Validity replacement % (alternative to validity duration in absolute time) ... 86
14.2 Replacement based on duration in absolute time (the default, and alternative to validity %) ... 87
14.3 KeyTalk client configuration file creation and advanced settings (Citrix/RD/Multi-User) ... 87
14.4 Warning when failing to renew automatically ... 88
15. End-point user/device administration and hardware recognition ... 89

16. Certificate management : SEATS ... 92

16.1 Certificate and key roll-over, encrypting stored private keys ... 92

16.1.1 HSM and MySQL Db encryption key ... 93

16.2 Historic certificate lookup ... 94

16.3 4-eye download principle for private keys ... 96

16.4 Revocation of certificates ... 97

16.5 Certificate pre-enrolment / enrolment ... 98

16.6 Certificate propagation: S/MIME email address book directory ... 99

16.7 Certificate propagation: AD AltSecurityIdentities ... 99

16.8 KeyTalk's LDAP S/MIME secure email address book directory ... 100

16.8.1 REST API and KeyTalk client LDAP S/MIME address-book client auto configure ... 102

16.9 Importing existing and new certificates ... 102

17 Certificate reporting and management meta-data ... 104

17.1 Dashboard reporting ... 104

17.2 Legacy reporting ... 105

18 User browser based self-service portal ... 107

19 Network architecture: KeyTalk in a network and HA ... 109

19.1 High Availability setup ... 110

19.2 High Availability and firmware upgrades ... 110

20 KeyTalk webUI management roles and strong authentication ... 112

20.1 Enable external CA-source trust for strong authentication ... 114

20.2 Active Directory for Management Access ... 114

21 Log files and reports ... 116

22 KeyTalk agents, clientless and automated/manual enrolment ... 117

22.1 Why a client/API footprint? ... 117

22.2 Client enrolment, clientless experience ... 117

22.3 Apple end-point device support iPhone/iPad/Mac ... 118

22.4 Mobile Device Management (MDM) support ... 119

22.4.1 MobileIron Core ... 119

22.4.2 Office 365 Intune ... 122

21.4.2.1 BETA Disclaimer ... 122

22.4.2.2 Introduction ... 122

22.4.2.3 High level flow and requirements PKCS import PFX (ie S/MIME certificates) ... 122

22.4.2.4 KeyTalk Office 365 Intune S/MIME deployment connection configuration steps ... 123

22.4.2.4.1 Set up the PFX Certificate Connector ... 123

22.4.2.4.2 Set up Azure Enterprise Application rights 125
22.4.2.4.3 Finalize KeyTalk configuration for Intune and push S/MIME PFX to Intune 131
22.4.2.4.4 Validating if Intune received the PFX certificate 132
22.4.2.5 KeyTalk Office 365 Intune SCEP deployment connection configuration steps........ 133
22.4.2.5.1 Configure your KeyTalk certificate TEMPLATE 133
22.4.2.5.2 Connect your SCEP certificate template to an Azure Module RA connector 133
22.4.2.5.3 Whitelist your Firewall rules for inbound SCEP ports 133
22.4.2.5.4 Obtain your SCEP for Intune Registration Authority certificates 133
22.4.2.5.5 Configure KeyTalk as a SCEP server. 134
22.4.2.5.6 Test the KeyTalk SCEP server with intune 134
22.4.2.5.7 Configure Office 365 for SCEP using Intune 135
22.4.3 VMware Workspace One UEM / AirWatch. 139
22.5 Manual certificate (and key) enrolment 140
22.5.1 Process flow and logic 140
22.5.2 KeyTalk CKMS InternalDB RA auth module and custom certificate meta data 140
22.5.3 Enrolled devices and users. 141
22.5.4 Managing certificate reporting meta data, key and CSR, email/SMS warnings 142
22.5.5 CSR manual processing, and signed certificate manual upload 143
22.5.6 Downloading current and historic/archived certificates and keys 143
23 (Windows) Outlook automated email disclaimer configuration 145
24 Configuring OWA/EO for S/MIME 147
25 Changing docker IP 147
26 KeyTalk contact details and 3rd line support 148

ANNEX A: Deploying KeyTalk virtual appliance in AWS. 149
ANNEX B: Importing KeyTalk virtual appliance in Azure 152
ANNEX C: Sample S/MIME use-case configuration 157
ANNEX D: Sample Third party S/MIME use-case configuration 172
ANNEX E: Sample end-user machine certificate use-case configuration. 186
ANNEX F: Sample (web)server certificate use-case configuration 195
ANNEX G: Migrating from KeyTalk 5.8.14 to KeyTalk 6.x 202

1. Quickguide Introduction and download

The KeyTalk Certificate and Key Lifecycle Management solution (CKMS) / KeyTalk PKI management solution, automates user client, device, server and IoT certificate & key management and enrolment, based on 1 or multiple defined certificate templates, and a connected commonly used Registration Authority (RA).
To deploy in your network use minimally 1 KeyTalk virtual application server.

It's used to create, process, renew (ie life-cycle manage): Server certificates, client certificates, and IoT certificates. Automation is attained using KeyTalk supported native third party API integration, REST API, ACME and SCEP protocols

For production High Availability purposes an external MySQL Db (single or cluster) can be added. This MySQL Db is used to store configuration settings, store issued certificates and key-pairs (in an AES 256 encrypted format), as a central repository for the front-end KeyTalk virtual application server(s). Allowing you to setup a 3rd party LoadBalancer and multiple KeyTalk application servers for scalability purposes.

Download your preferred KeyTalk virtual appliance here: https://keytalk.com/support
You can obtain the KeyTalk client/app here: https://keytalk.com/support

Note: The KeyTalk client/app is not required when using MDM, or LoadBalancers, or ACME protocol.

2. Quickguide Resource requirements

KeyTalk's virtual front-end application appliance is based upon Ubuntu 22.04 LTS and requires:

- 100 GB diskspace, preferred premium SSD
- 4 CPUs
- 16 GB memory
- 1 IPv4 and/or IPv6 address (default assigned through DHCP)

KeyTalk's optional external MySQL Db can be any customer preferred edition: Cluster CGE, Enterprise, Standard or Community edition, version 8 is required.
The database itself is provided by KeyTalk as a generate script, and will grow with each user by roughly 15 kB (kilobyte) per stored certificate. The MySQL Db should have at least:

- Minimally 1 GB diskspace per 100.000 certificates
- 4 CPU's
- 16 GB memory
- 1 IPv4 and/or IPv6 address
- 15.000 IOPS

A virtual appliance version of the MySQL Db is also available. Note that this is a single instance and not clustered:
htpps://downloads.keytalk.com/downloads/server/OVF_KeyTalkv-7.4.3_dbserver_UbuntuServer-22.04.02.zip

Whenever the KeyTalk firmware is upgraded, the KeyTalk server will upgrade the connected MySQL Db as well to match needed database changes.

The following list is a quick overview of used ports to define firewall rules prior to setup for the KeyTalk virtual application server:

REQUIRED Ports:

Port	Type		Function
80	TCP	Inbound	Download certificates when offering S/MIME cert downloads to third parties using the internal KeyTalk Certificate Authority.
	TCP	Inbound	When KeyTalk server is used as its Private CA CRL Distribution Point (CDP) the CRL is accessed over port 80 (optional)
	TCP	Outbound	*.ubuntu.com (Ubuntu security package updates, when not using internal repo)
443	TCP	Inbound	KeyTalk's secure KeyTalk client communication using KeyTalk's Private CA based client-server certificate
	TCP	Outbound	pypi.python.org (Python package updates)
443	TCP	Outbound	KeyTalk firmware update server https://downloads.keytalk.com
53	TCP	Outbound	DNS
123	TCP	Outbound	NTP
3000	**TCP**	**Inbound**	**KeyTalk management interface**

OPTIONAL Ports:

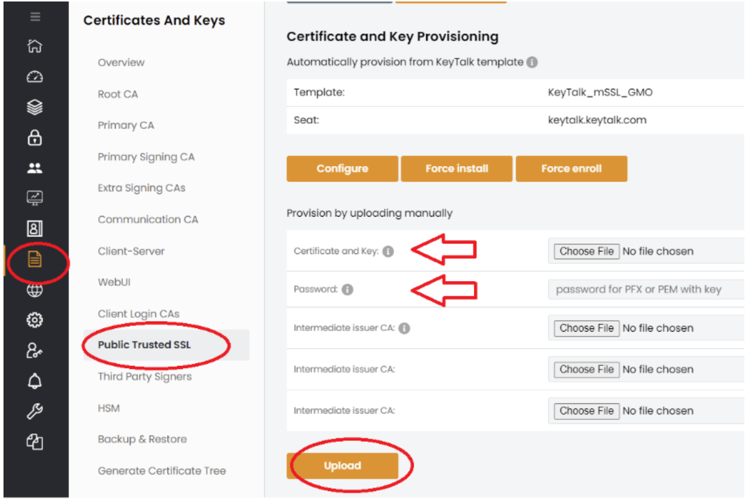
Port	Type		Function
80	TCP	Inbound	Inbound Intune SCEP requests over HTTP
	TCP	Outbound	NDES to MS-CA over HTTP
	TCP	Outbound	GlobalSign and TRUSTZONE CA platform communication : https://system.globalsign.com
443	TCP	Outbound	DigiCert QuoVadis CA platform: https://tlws.quovadisglobal.com/
	TCP	Outbound	DigiCert CertCentral CA platform US: https://certcentral.digicert.com
	TCP	Outbound	DigiCert CertCentral CA platform EU: https://certcentral.digicert.eu
	TCP	Outbound	NDES to MS-CA over HTTPS
	TCP	Outbound	Azure Integration: https://graph.microsoft.com Azure Global Service https://graph.microsoft.us Azure US government L4 https://dod-graph.microsoft.us Azure US Government L5 https://graph.microsoft.de Azure Germany https://microsoftgraph.chinacloudapi.cn Azure China
88	TCP	Outbound	Kerberos token authentication
389	TCP	Outbound	AD/LDAP non-secure
514	UDP	Outbound	Syslog server
636	TCP	Outbound	AD/LDAP secure
1812	TCP	Outbound	RADIUS authentication communication
3000	TCP	Outbound	KeyTalk end-user self-service portal (mTLS authentication only)
3306	TCP	Outbound	MySQL communication over TLS 1.3
8443	TCP	Outbound	GlobalSign Atlas HVCA certificate requests: https://emea.api.hvca.globalsign.com:8443

3. Quickguide Trusted secure client-server communication

KeyTalk’s default end-point communication protocol makes use of TLS 1.2.
As TLS enforcement is very strict, it is necessary to ensure that the FQDN and/or IP address(es) of the KeyTalk application server matches with the SubjectAlternativeName value set in the certificate configured for the KeyTalk application server (sub-menu tab “client-server” as found under “certificates and keys” main menu tab!!)

When using the KeyTalk end-point REST API to integrate KeyTalk functionality into your own app/software, it is highly recommended to enforce similar strict matching criteria and for example not disregard a SAN mismatch.

When deploying certificates to Apple end-points, a globally trusted SSL certificate is required for KeyTalk CKMS, given Apple’s App Transport Security (ATS) policy as used by the KeyTalk apps for MacOSX and iOS. Enroll/install this trusted certificate and key for the proper FQDN under :



Certificate and Key Provisioning

Automatically provision from KeyTalk template

Template: KeyTalk_mSSL_GMO

Seat: keytalk.keytalk.com

Provision by uploading manually

Certificate and Key: No file chosen

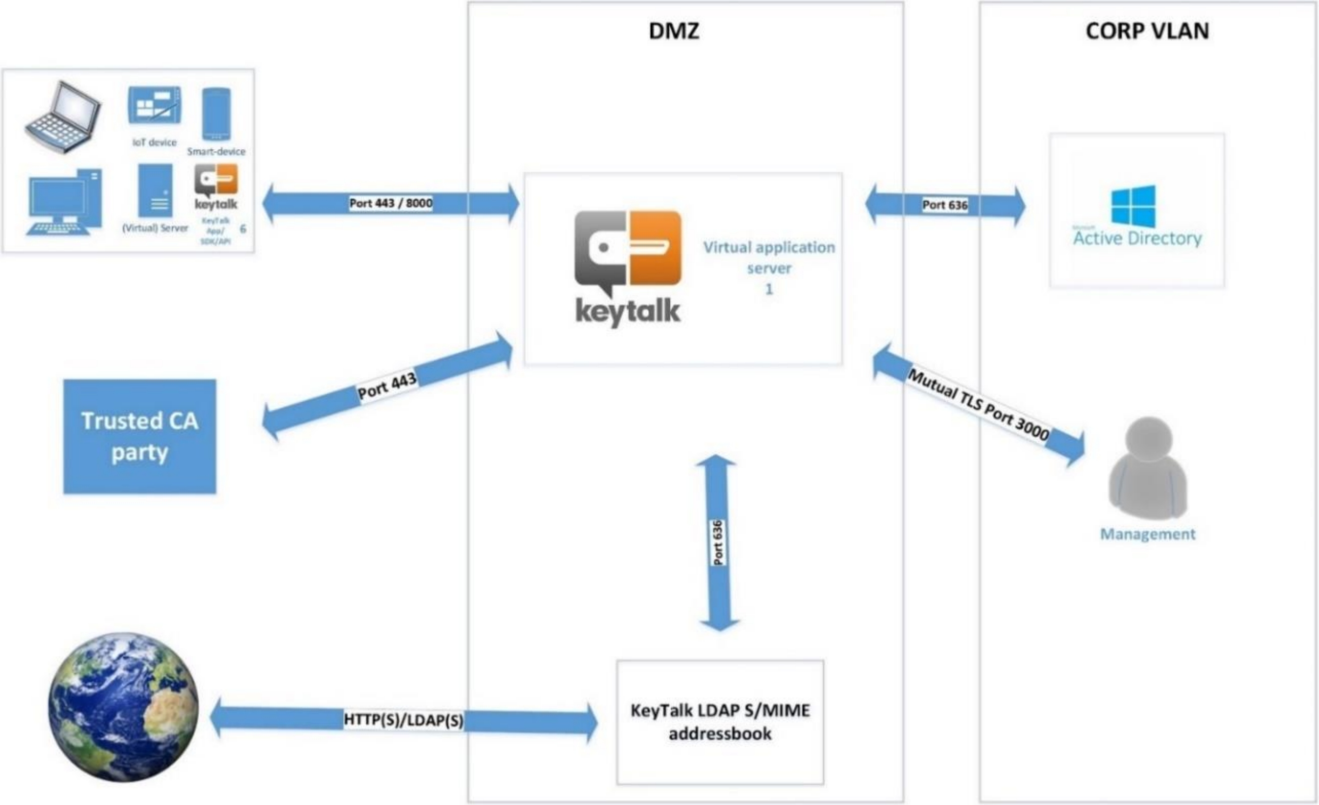
Password: password for PFX or PEM with key

Intermediate issuer CA: No file chosen

Intermediate issuer CA: No file chosen

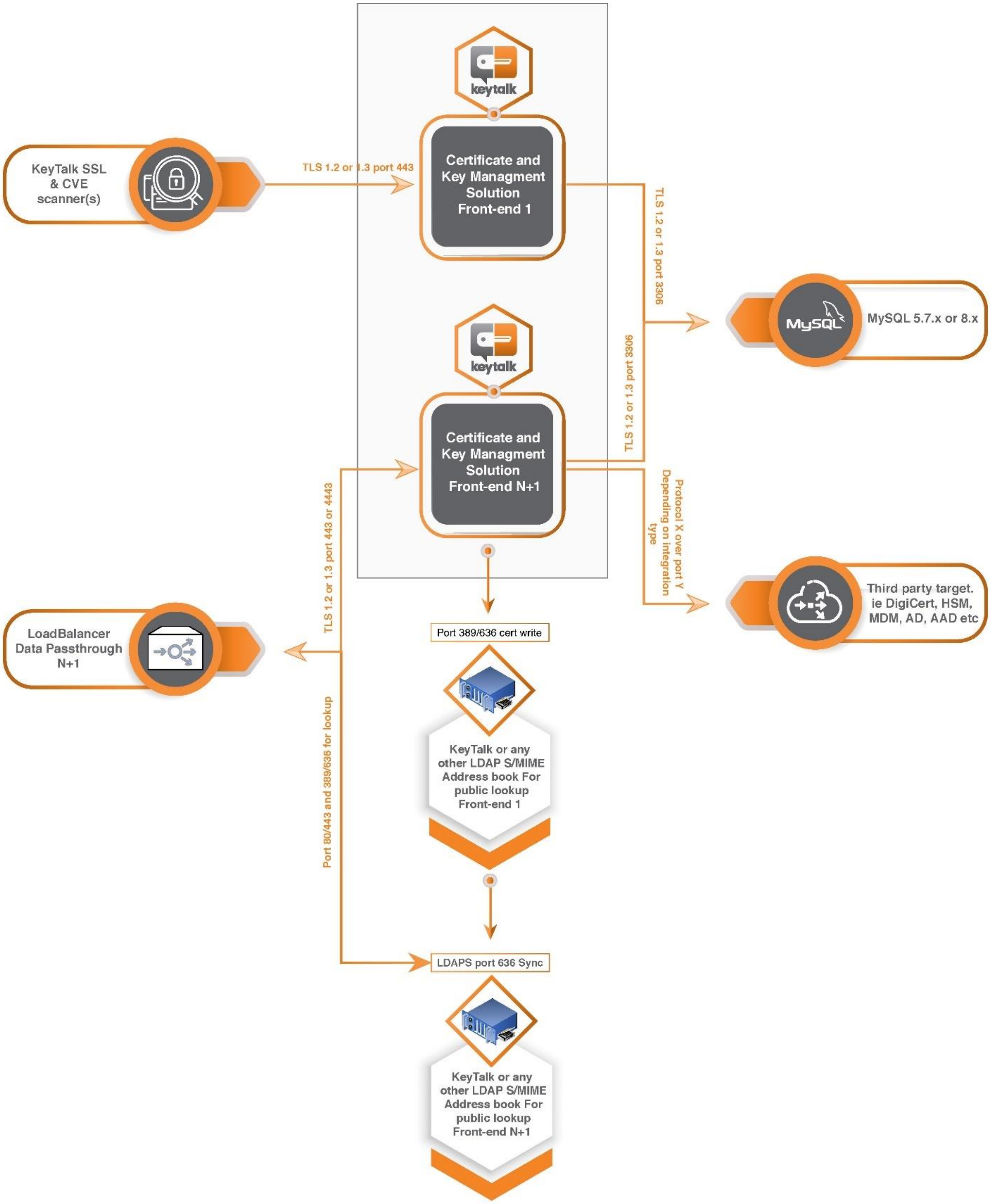
Intermediate issuer CA: No file chosen

4. Quickguide high level network setup non-HA



NOTE: The LDAP S/MIME secure email address-book, Trusted CA party (such as DigiCert) and AD are not required, though typically used in an S/MIME or trusted server world-facing scenario

5. Quickguide high level network setup High Availability


KeyTalk SSL & CVE scanner(s)
TLS 1.2 or 1.3 port 443
Certificate and Key Managment Solution Front-end 1
keytalk
TLS 1.2 or 1.3 port 3306
MySQL
MySQL 5.7.x or 8.x
TLS 1.2 or 1.3 port 3306
keytalk
Certificate and Key Managment Solution Front-end N+1
Protocol X over port Y Depending on integration type
Third party target. ie DigiCert, HSM, MDM, AD, AAD etc
TLS 1.2 or 1.3 port 443 or 4443
LoadBalancer Data Passthrough N+1
Port 389/636 cert write
KeyTalk or any other LDAP S/MIME Address book For public lookup Front-end 1
Port 80/443 and 389/636 for lookup
LDAPS port 636 Sync
KeyTalk or any other LDAP S/MIME Address book For public lookup Front-end N+1

NOTE: The LDAP S/MIME secure email address-book, Trusted CA Party and AD are not required, though typically used in an S/MIME or trusted server world-facing scenario

6. Quickguide local system admin login

To login as KeyTalk admin from the hypervisor Command Line Interface (CLI) or SSH use:
Username: keytalk
Password: change!

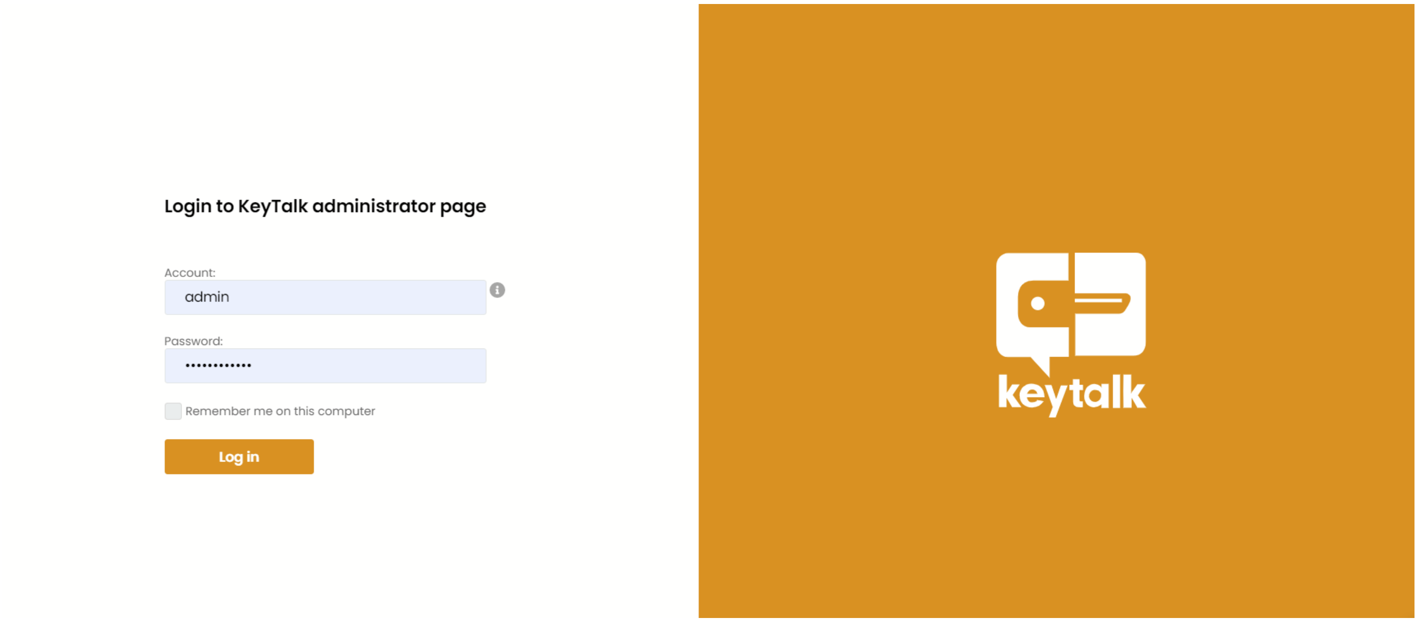
To see what IP address was assigned from your DHCP server type the command: ip a

Since ROOT is disabled on both system- and SSH-level, when entering CLI commands use “sudo”

Login into the Management Graphical User Interface from a browser using:
https://<myipaddress>:3000
Username: admin
Password: change!

NOTE: As KeyTalk ships with a self-signed factory default SSL certificate and keypair, your browser will give a security trust warning until you replace it with a trusted certificate with a proper SAN. Replace the KeyTalk port 443 and port 3000 certificate as soon as possible to ensure proper TLS trust.

NOTE: Strong mTLS authentication for the management GUI is supported under the menu item “Admin”, when logged in under the “admin” account, and requires a proper matching trusted client certificate subject meta data as issued under the KeyTalk Signing CA, or other trusted CA as configured under “Certificates and Keys” “Client login CAs”.



Login to KeyTalk administrator page

Account: admin

Password: ••••••••••••

☐ Remember me on this computer

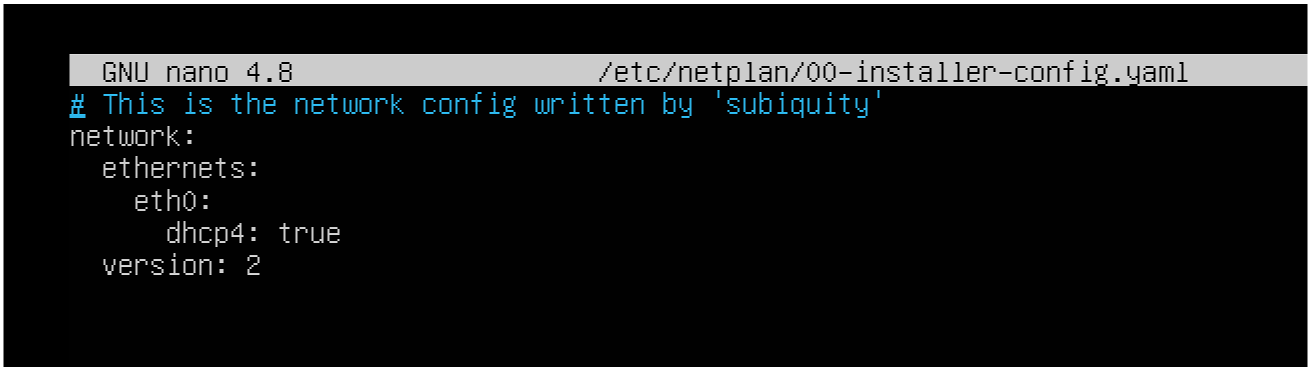
7. Quickguide manually changing IP addresses & default gateway

When you do not make use of a DHCP server, you need to configure the IP address(es) and basic networking manually.

Launch the following command from your KeyTalk virtual appliance command line console:

```
sudo nano /etc/netplan/00-installer-config.yaml
```

You will see:

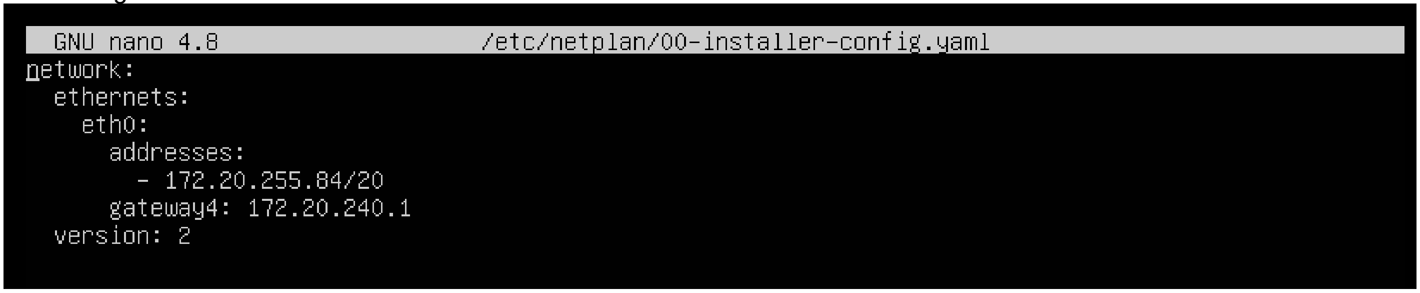


```
  GNU nano 4.8                    /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    eth0:
      dhcp4: true
  version: 2
```

Change the config file to:

```
network:
   ethernets:
     eth0:
        addresses:
           - <my_ipv4_address>/<subnet-prefix>
        gateway4: <my_ipv4_gateway_address>
   version 2
```

For example, my IPv4 address is: 172.20.255.84
The config will look like



```
  GNU nano 4.8                    /etc/netplan/00-installer-config.yaml
network:
  ethernets:
    eth0:
      addresses:
        - 172.20.255.84/20
      gateway4: 172.20.240.1
  version: 2
```

Save your settings using: <CTRL X> and selecting YES to save the modifications.

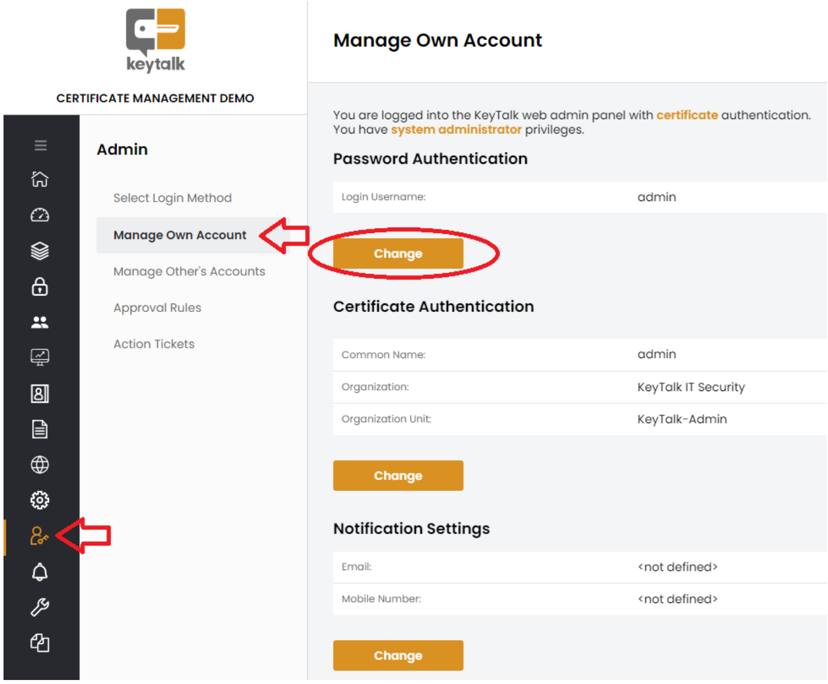
Now effectuate the network change by typing the following command:

```
sudo netplan apply
```

Use the command ifconfig to check if your IP change was properly effectuated
For further references see: https://netplan.io/examples/#using-dhcp-and-static-addressing

8. Quickguide: Getting started

Step 1: **Change your local system admin and CLI root password**



Manage Own Account

You are logged into the KeyTalk web admin panel with certificate authentication.
You have system administrator privileges.

Password Authentication

Login Username:	admin

Certificate Authentication

Common Name:	admin
Organization:	KeyTalk IT Security
Organization Unit:	KeyTalk-Admin

Notification Settings

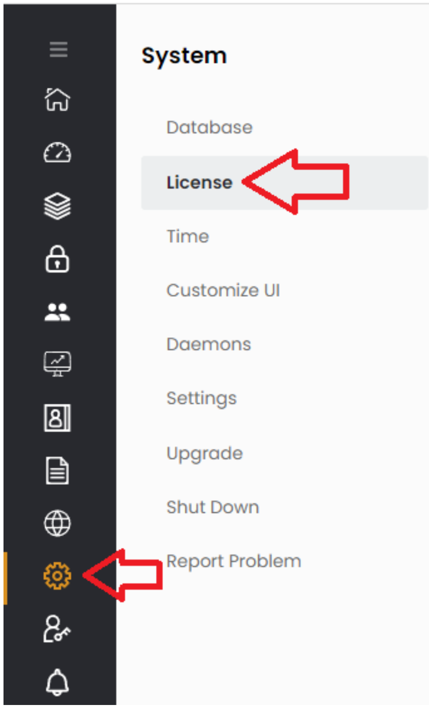
Email:	<not defined>
Mobile Number:	<not defined>

From the GUI you can change your KeyTalk local system admin password.
From your SSH/Hypervisor CLI you can change your CLI management password using:

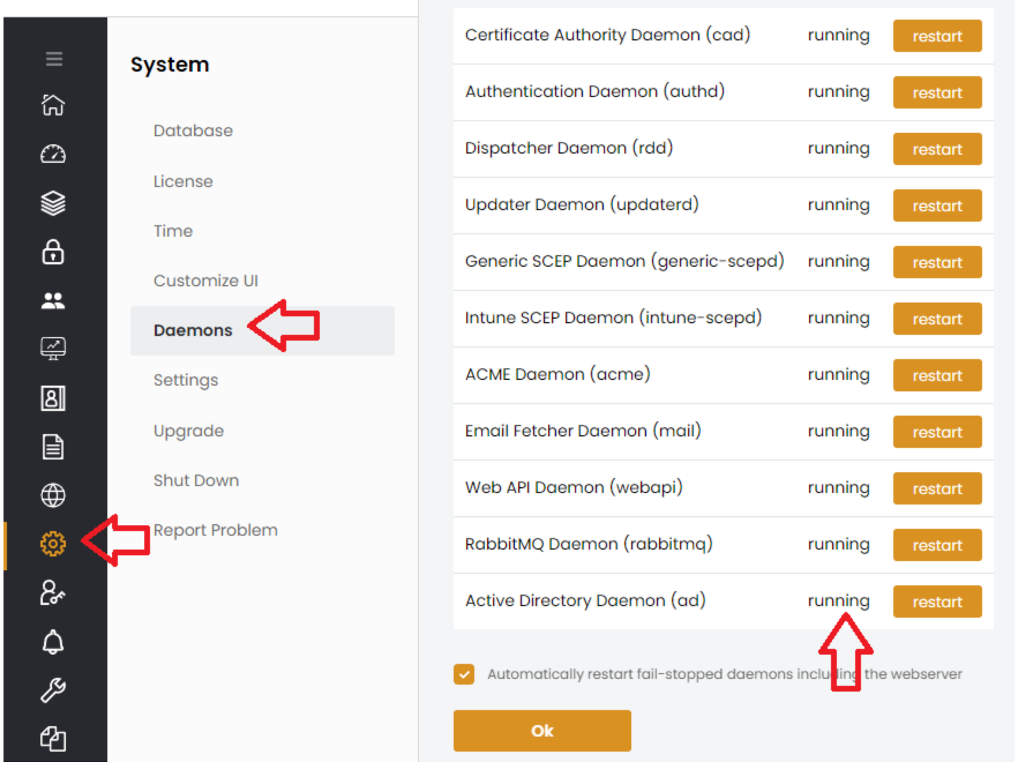
```
passwd
```

Ensure you remember the set console password and/or set SSH key as there is no way to reset a lost console password or key, other than to reinstall from scratch !!

Step 2: **Ensure your KeyTalk license is valid, when it's not, upload one as provided to you:**



Step 3: Ensure all daemons are running

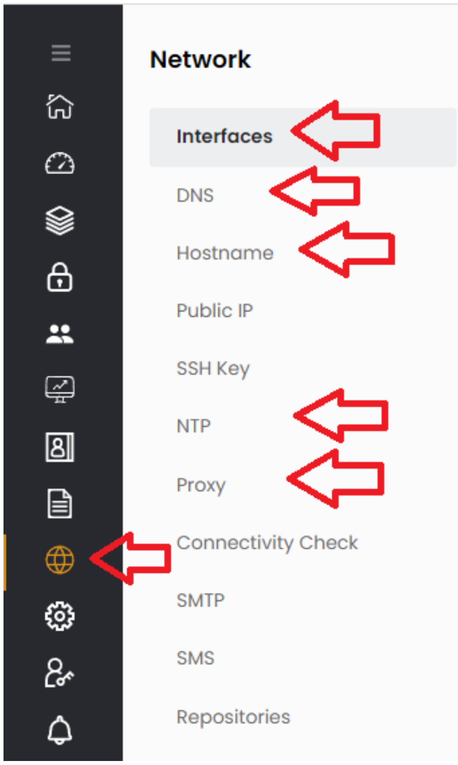


Certificate Authority Daemon (cad)	running	restart
Authentication Daemon (authd)	running	restart
Dispatcher Daemon (rdd)	running	restart
Updater Daemon (updaterd)	running	restart
Generic SCEP Daemon (generic-scepd)	running	restart
Intune SCEP Daemon (intune-scepd)	running	restart
ACME Daemon (acme)	running	restart
Email Fetcher Daemon (mail)	running	restart
Web API Daemon (webapi)	running	restart
RabbitMQ Daemon (rabbitmq)	running	restart
Active Directory Daemon (ad)	running	restart

☑ Automatically restart fail-stopped daemons including the webserver

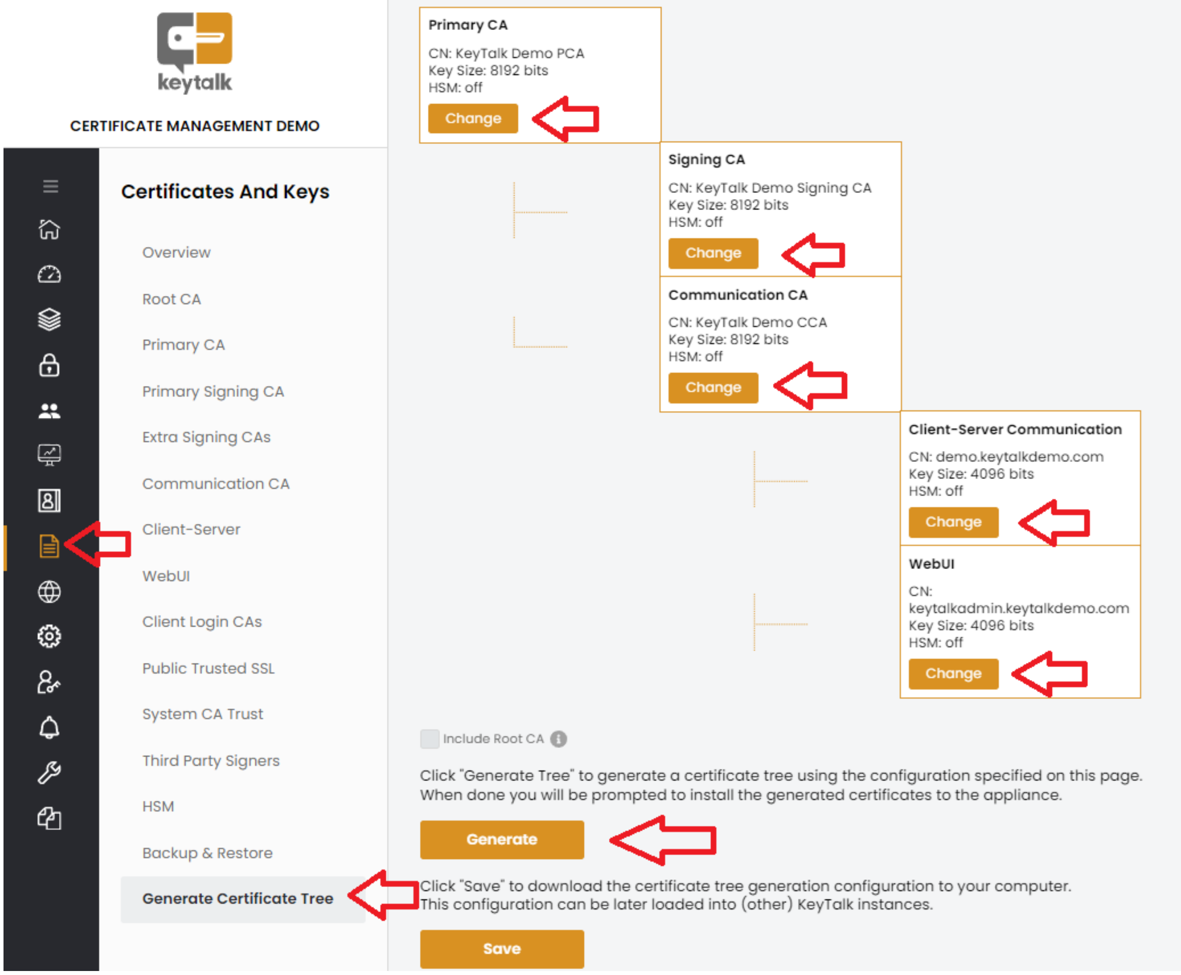
NOTE: The SCEP, Intune SCEP and ACME Daemon do not need to run, unless you are making use of SCEP or ACME protocol to issue certificates.

Step 4: Configure your network



NOTE: NTP is ALWAYS in UTC format and should never run with a local Offset

Step 5: Generate your unique private Certificate Authority tree and keys



Primary CA
CN: KeyTalk Demo PCA
Key Size: 8192 bits
HSM: off

Signing CA
CN: KeyTalk Demo Signing CA
Key Size: 8192 bits
HSM: off

Communication CA
CN: KeyTalk Demo CCA
Key Size: 8192 bits
HSM: off

Client-Server Communication
CN: demo.keytalkdemo.com
Key Size: 4096 bits
HSM: off

WebUI
CN: keytalkadmin.keytalkdemo.com
Key Size: 4096 bits
HSM: off

□ Include Root CA

Click "Generate Tree" to generate a certificate tree using the configuration specified on this page. When done you will be prompted to install the generated certificates to the appliance.

Click "Save" to download the certificate tree generation configuration to your computer. This configuration can be later loaded into (other) KeyTalk instances.

NOTE: Ensure that at least the CLIENT-SERVER certificate is configured with a **proper DNS resolvable Subject Alternative Name !!**
If the SAN value is incorrect, then the KeyTalk client will fail to connect due to TLS being properly enforced.

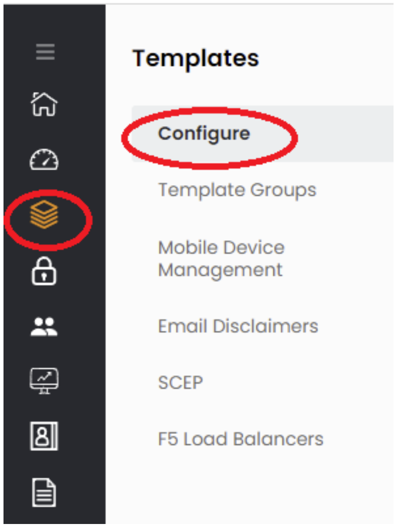
Example: When your client-Server certificate is communicating with keytalk.keytalk.com with IPv4 address 78.237.33.242 or using a LoadBalancer on IPv4 address 331.22.80.170, the SAN value should look like: DNS:keytalk.keytalk.com,IP:78.237.33.242,IP: 331.22.80.170 when both the FQDN and the IPv4 address are being used to connect to the KeyTalk server.

It is good practice **NOT** to use an IP address in the SAN field, and instead use proper DNS propagation, resulting in only having the FQDN show in the SAN. However, in Proof of Concept environments commonly we see solely IP addresses being used.

After pressing **GENERATE**, the CA-tree will be created. Once ready select **INSTALL** to effectuate your unique private Certificate Authority.

SAVE your CA-tree for offline backup purposes, so your private keys are always available to only you

Step 6: Add one or more certificate templates in TEMPLATES



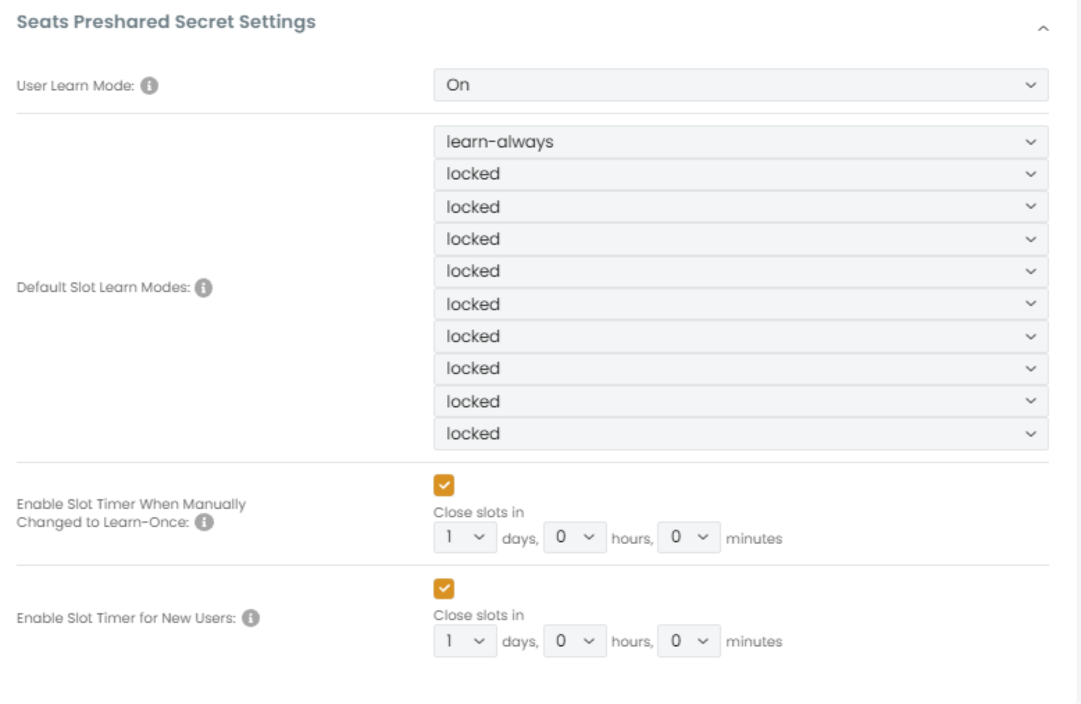
A KeyTalk TEMPLATE is a certificate template, policy template, and link to a used CA source. It is the core of the KeyTalk server functionality. Without it, you cannot issue nor manage certificates!

Default certificate fields and (extended) key-usage) can be optionally overwritten with mapped metadata coming from the connected and configured Registration Authority (such as your AD), provided the target CA allows for it. Trusted Certificate Service Providers will often enforce predefined certificate profiles (example DigiCert or GlobalSign), private CAs often will not.

Note For developers using the KeyTalk REST API:
The client hardware component numbers and tooltip defined in HWSIG are an implementation for use with the KeyTalk agents only. It only accepts numeric values.
Developers are free to use the numeric information in the HWSIG field in any way they see fit, or even disregard it, and generate their own Hardware Signature hash, usable with the SEATS administration within KeyTalk.

Step 8: Set seat hardware recognition/pre-shared-secret settings

Set user hardware recognition (extra Factor Authentication based on a static pre-shared secret) per KeyTalk Certificate Template. One slot typically equals 1 device of an end-user/end-point



Seats Preshared Secret Settings

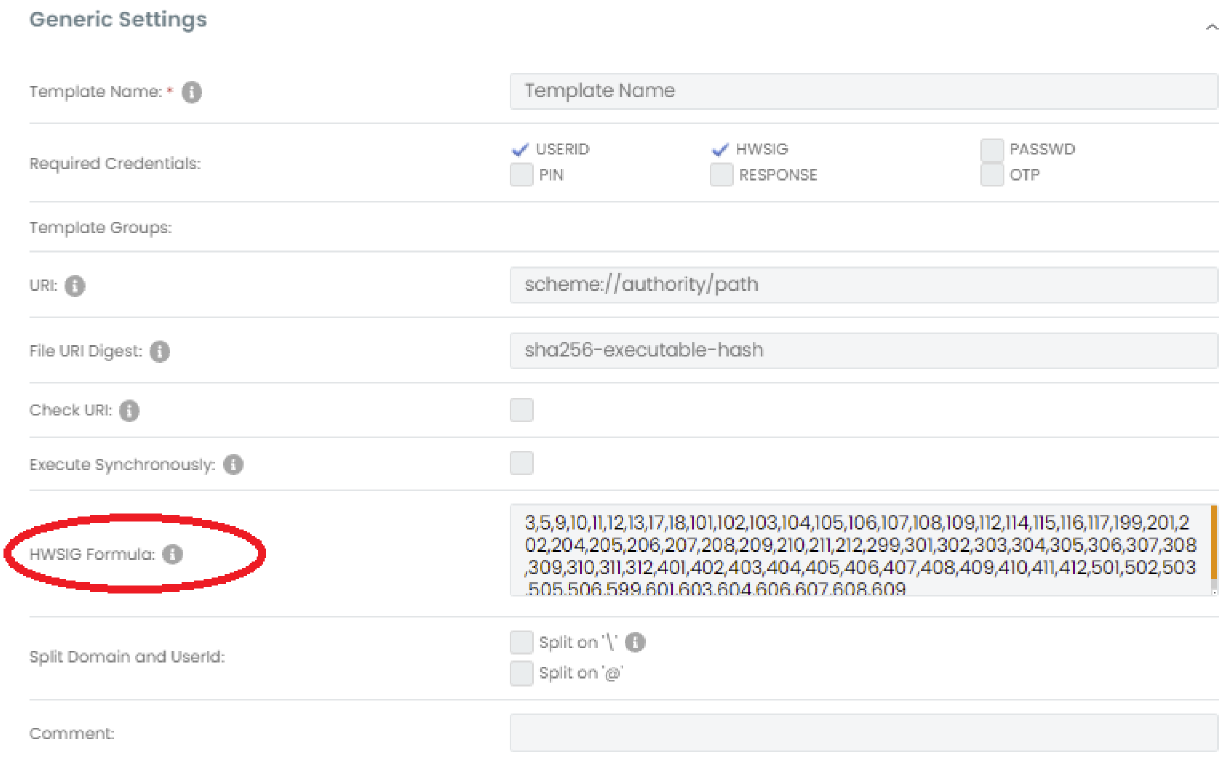
User Learn Mode: On

Default Slot Learn Modes:
learn-always
locked
locked
locked
locked
locked
locked
locked
locked
locked

Enable Slot Timer When Manually Changed to Learn-Once: ☑
Close slots in 1 days, 0 hours, 0 minutes

Enable Slot Timer for New Users: ☑
Close slots in 1 days, 0 hours, 0 minutes

KeyTalk's SEATS preshared secret settings allows for added Multi-Factor- Authentication based on Hardware/software recognition of a user-device as defined in the HWSIG field of the TEMPLATE.



Generic Settings

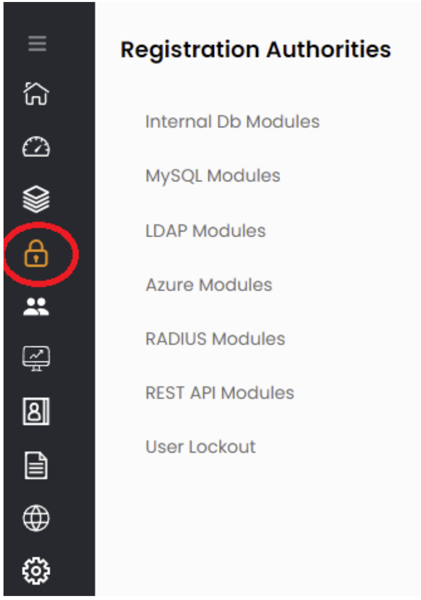
Template Name: *	Template Name
Required Credentials:	☑ USERID ☑ HWSIG □ PASSWD □ PIN □ RESPONSE □ OTP
Template Groups:	
URI:	scheme://authority/path
File URI Digest:	sha256-executable-hash
Check URI:	□
Execute Synchronously:	□
HWSIG Formula:	3,5,9,10,11,12,13,17,18,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,599,601,603,604,606,607,608,609
Split Domain and UserId:	□ Split on '\' □ Split on '@'
Comment:	

The HWSIG can always be learned automatically for an initial unknown, but first time positively authenticated user (on), or manually (off), or scheduled ie automatic learning for a period of time only. Each slot represents 1 stored device SHA2 hash, whereby each hash effectively becomes a static pre-shared secret for that particular device.

The timers allow for the automated closing of a Learn-Once slot. Ie when a device hasn't been learned within the timer period, it will close automatically, thus reducing the risk of a malicious third party successfully using a phishing attack to register their own device.

Learn-Always is for most customer recommended to be used for PoC and production purposes, as it allows any device with a positive authentication to obtain a certificate (and key). Only customers who have a need to add extra protection to prevent certificates from being issued to non-authorized devices should use this feature.

Step 9: Connect a Registration Authority source to your TEMPLATE



Registration Authorities

- Internal Db Modules
- MySQL Modules
- LDAP Modules
- Azure Modules
- RADIUS Modules
- REST API Modules
- User Lockout

KeyTalk enables the Admin to connect a configured TEMPLATE to a Registration Authority which is typically an Identity Provider source.
Commonly LDAP (including (Azure Entra ID and local AD), RADIUS (including common OTP tokens), and MySQL are used by the majority of current day customers.

Other Identity Provider solutions can be supported. Request sales@keytalk.com to add your preferred Identity Provider(s) or protocols to the solution based on your business-case if you are missing one.

KeyTalk strongly advises to start with a basic connector and test/check the connection using our inbuilt internal database RA, to ensure the connection and KeyTalk solution in general works, before using the more advanced settings such as secure connections over LDAPS or EAP, or even certificate field mapping, which allows TEMPLATE certificate template default values to be overwritten with user unique information from your Identity Provider source)

For most LDAP / Active Directories as of Server 2012R2, the BIND setup process looks like:
1) Edit the LDAP/AD server as defined under the LDAP authentication module
2) BIND DN: $(userid)@mydomain.local
3) BIND password: $(password)
4) Allow empty password: UNCHECK! (this prevents potential anonymous BIND abuse)
5) BASE DN: dc=mydomain,dc=local

This should result in a positive certificate request when the BIND is successful over either LDAP or LDAPS protocol, based on a user's AD username and its corresponding password.

Using LDAPS is recommended from a secure communication point of view.
When using LDAPS ensure first that the BIND works using LDAP communication. Once this works, enable LDAPS, which requires an ldap server url update to ldaps:// AND change port to 636 AND have the LDAPS intermediate certificate and root uploaded to the KeyTalk authentication module in PEM or DER format (just the certificate NOT the private key).

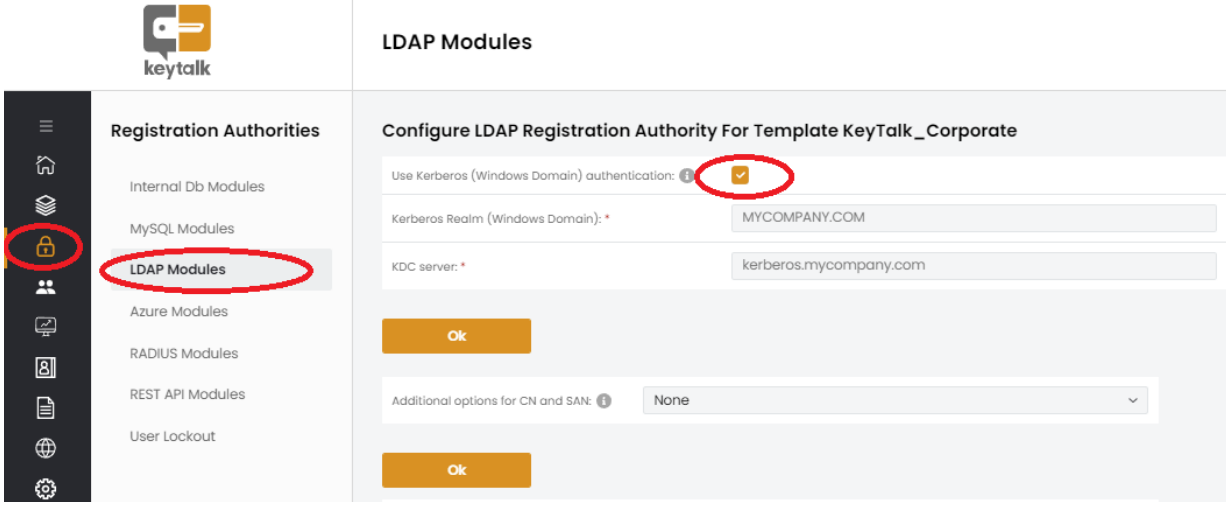
Ensure the used IP and/or FQDN are correct in the AD SSL certificate and ensure its SHA2, otherwise TLS will fail.

Certificate Field mapping is a powerful tool, allowing for default certificate template settings (ie KeyTalk TEMPLATE settings) to be overwritten with unique values coming from a user's account.

For example, attributes in an Active Directory or field names in a MySQL Db.
This allows for unique SAN values per unique Common Name, or even name@domain.local to be mapped to "name@domain.com" as an email address, or "displayname" as the certificate CN value.

Note that the search string likely requires sAMAccountName or another attribute depending on how you setup your AD.

Once the LDAP / AD connector works, you can optionally activate Kerberos based authentication.



LDAP Modules

Configure LDAP Registration Authority For Template KeyTalk_Corporate

Use Kerberos (Windows Domain) authentication: ☑

Kerberos Realm (Windows Domain): * MYCOMPANY.COM

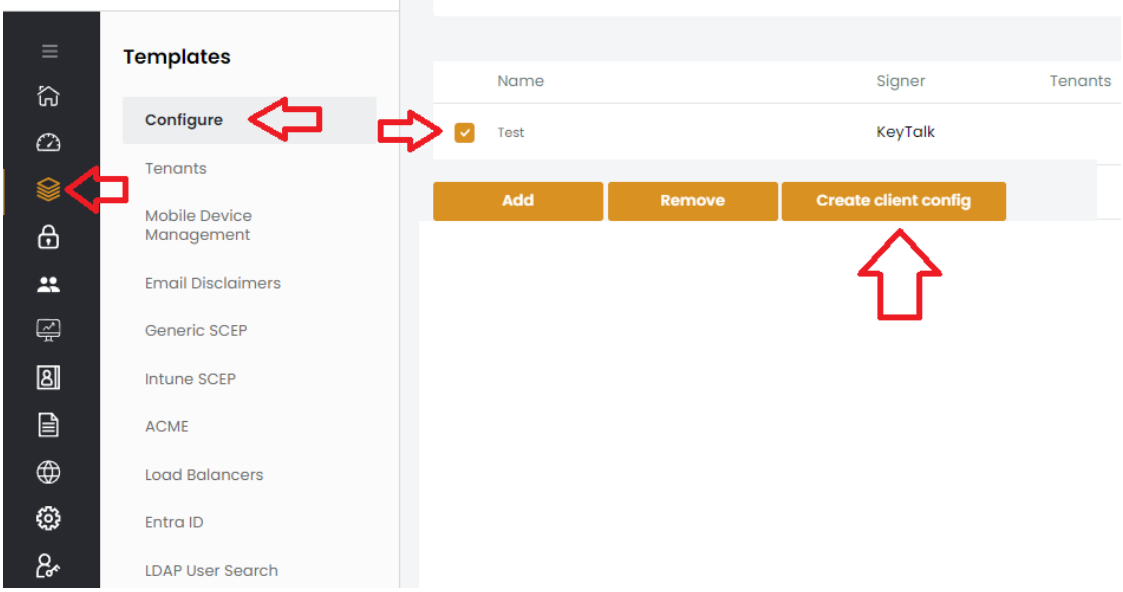
KDC server: * kerberos.mycompany.com

Ok

Additional options for CN and SAN: None

Ok

Step 10: Create KeyTalk agent configuration file



	Name	Signer	Tenants
☑	Test	KeyTalk	

Select the certificate TEMPLATE you wish to have the KeyTalk clients connects to, and select CREATE CLIENT CONFIG.

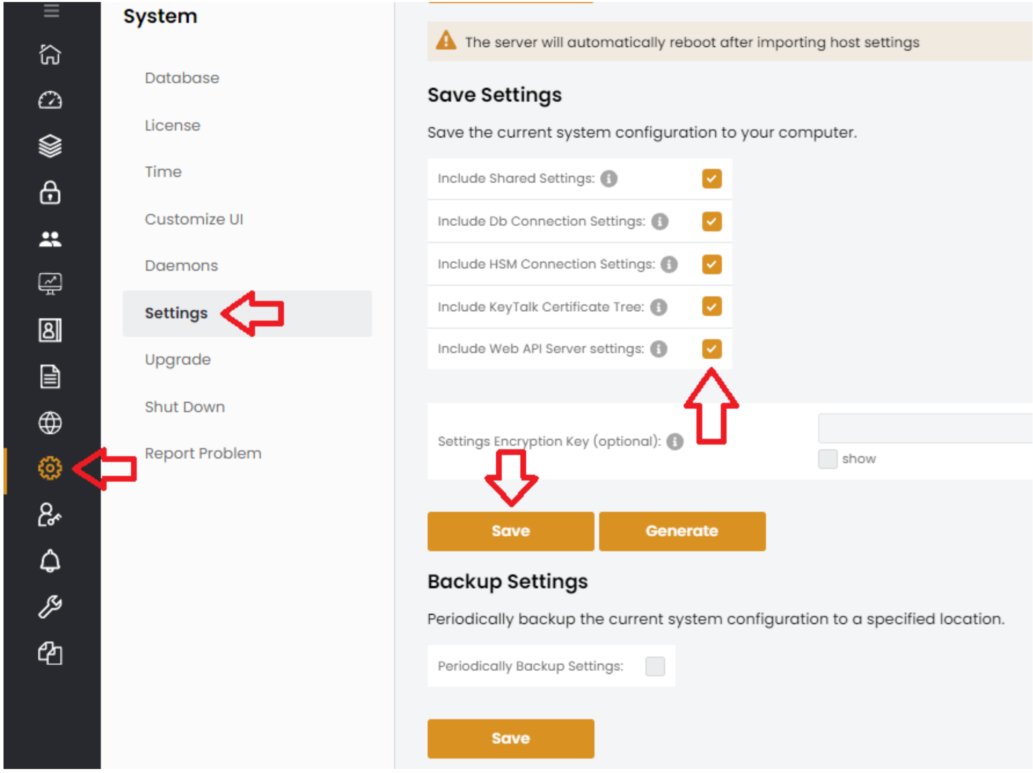
This will result in a .rccd (real client configuration data) file and needs to be imported into a KeyTalk agent, or be made part of the installer, or some of it content could be statically used with the REST API, in order to obtain a certificate (and private key) from your KeyTalk environment.

The configuration file can also be pushed with the Windows installer, or even be embedded in your company specific DMG, APK or IPA installer. Kindly contact KeyTalk support if this is needed.

More information on the contents of the .rccd file can be found here (chapter 2.1 page 6):
https://downloads.keytalk.com/downloads/documents/KeyTalkApi.pdf

When solely using non-KeyTalk proprietary communication protocols, such as SCEP or ACME or native MDM integration, the Client Config file is not required.

Step 11: **(Periodically) Backup your KeyTalk configuration settings**



The server will automatically reboot after importing host settings

Save Settings

Save the current system configuration to your computer.

Include Shared Settings: ☑
Include Db Connection Settings: ☑
Include HSM Connection Settings: ☑
Include KeyTalk Certificate Tree: ☑
Include Web API Server settings: ☑
Settings Encryption Key (optional): □ show

Backup Settings

Periodically backup the current system configuration to a specified location.

Periodically Backup Settings: □

Select what needs to be included in the native KeyTalk settings and configuration backup file. Shared settings include all stored certificates, private keys, and certificate template information.

AES-256-GCM encryption can be applied to the backup-file by applying a key (input or generated).

NOTE 1: **If you forget the encryption key, there is no way to restore the backup!!**

NOTE 2: **When AES-256 CBC HMAC-SHA256 is used to encrypt private keys as stored under KeyTalk SEATS (configured under SYSTEM -> Settings -> Database) then the backup will contain the encrypted private keys, which can only be decrypted when configuring the correct AES-256 CBC HMAC-SHA256 key in your KeyTalk instance.**
You can choose to include or exclude the AES-256 CBC HMAC-SHA256 as part of the backup.

NOTE 3: **The content of the encrypted backup can be decrypted using the following application on a Linux system:**
https://downloads.keytalk.com/downloads/tools/aesencrypt-5.7.1.tgz

Periodic backups are supported simply by adding a checkmark and selecting the appropriate details you wish periodically backed-up to a remote server supporting SSH authentication.

For secure remote connections, KeyTalk server has its own SSH key, generated and configured under https://<IP/FQDN>:3000/network?sshkey

9. Quickguide: Frequently encountered setup issues

a) ***I can ping the KeyTalk admin IP address, but am unable to login as a KeyTalk admin!!***
Very likely you are trying to login just after the KeyTalk application server started. Wait 1 minute to have the KeyTalk Daemon start properly and retry. Also don’t forget to use :3000 to login to the management GUI.

b) ***KeyTalk is up and running, but my KeyTalk agents are unable to authenticate or connect!!***
When the client cannot connect, very likely the KeyTalk server cannot be reached, so ensure the proper IP or FQDN are used to connect, AND ensure the IP and/or FQDN are part of the KeyTalk client-server certificate SAN value. If not part of the SAN the TLS handshake will fail.
To check your configured client-server certificate SAN, en see of the connection is at all possible, open a browser and type: https://<mydomain_or_myIPaddress>/rcdp/2.4.0/hello

Your end-point/user authentication credentials could also be invalid (such as username/password), or you are enforcing hardware recognition whereby the device used is not being trusted by the KeyTalk system.

To resolve the invalid credentials, ensure the used KeyTalk TEMPLATE is connected to an Authentication Module, and that this Authentication Module is properly configured.
To resolve the hardware recognition, in DEVID USERS a device slot can be set to learn-once or to learn-always for the particular username. Alternatively, in TEMPLATES set learn-mode to ON and set the device slot to learn-always or to learn-once

Alternatively, your KeyTalk license may have expired, or after uploading a valid license the CA Daemon did not get started. To resolve this issue: upload a valid KeyTalk license.

c) ***KeyTalk is up and running, but my KeyTalk agents are unable to obtain a certificate from an external CA***
Likely your trusted CA provider, such as GlobalSign or DigiCert, doesn’t accept your server or client certificate request. This usually happens because your requesting IP isn’t whitelisted, or an email address needs to be included for ePKI certificates, so ensure that for example the used username equals the email address, or map the SAN email value do your Active Directory attribute “mail”

Trusted CA providers also require your company and domainname to be vetted. When you request a certificate for a non-vetted domainname, either for your servers or ePKI, the certificate request will likely fail. To resolve, ensure your used domainname has been vetted with your trusted CA provider for the certificate product you are using.

d) ***My Active Directory BIND settings are 100% correct, yet KeyTalk refuses to connect to it!!***
First ensure that the KeyTalk server is allowed to talk LDAP over the set port to your AD/LDAP.

KeyTalk makes use of a BIND to your AD, based on the user’s username/password. Make sure your BIND works properly without LDAPS, once the BIND is successful, activate LDAPS.

When the BIND fails using LDAPS your LDAPS certificate on your AD and/or LoadBalancer is likely considered invalid. Possibly it expired, or does not contain the proper FQDN or IP in the SAN, OR your LDAPS certificate does not have an issuing CA, or it makes use of SHA1 which got deprecated some time ago.

To resolve: ensure your LDAPS certificate is valid, has a parent CA, and contains the proper FQDNs or IP in the SAN, and uses SHA2
Some customers tend to only make use of internal IP address, resulting in a failed LDAPS connection. In these cases, you could add the IP address to the SAN entry of your LDAPS certificate. Alternatively, you could update KeyTalk’s “Local DNS Lookup Database” as found under NETWORK -> DNS

e) ***Server certificates issued by KeyTalk to Apache or IIS servers do not bind to the server!!***
The most common cause of this problem is the fact that the certificate template (ie the TEMPLATE) does not have the Enhanced Key Usage “Server Authentication” activated.

f) ***My server/client certificate as issued by KeyTalk isn't trusted in my browser!!***
The browser does not trust the KeyTalk or Trusted CSPs CA tree. To resolve this issue import the CA tree in the client's browser to resolve this issue. When using the KeyTalk client/app the KeyTalk CA trust is automatically added to your local OS.
Alternatively, in May 2017 several browsers were updated to REQUIRE a server certificate's FQDN to be part of the DNS name entry in the certificate's Subject Alternative Name, ie the CommonName value is disregarded. To resolve this issue, map your preferred SAN entry to an Active Directory Attribute using KeyTalk's certificate field mapping feature.

g) ***I'm using a LoadBalancer for KeyTalk and/or LDAP/RADIUS etc, and connections fail!!***
KeyTalk enforces strict TLS requirements when using LDAPS/HTTPS. Ensure your LoadBalancer SSL certificate has the correct FQDN/SAN entries and resolve properly on your used DNS, similarly your target authentication solution requires proper FQDN/SAN entries AND requires its TLS certificate to be issued under a parent CA. Also note that SHA1 has been deprecated and will not be accepted for an LDAPS based connection.

h) ***I've found versions of OpenSource code that are not up to date within the KeyTalk virtual appliance!***
KeyTalk CKMS runs OS security patch fetching hourly and applies these within 24 hours of the update being fetched.

Further more the KeyTalk server makes use of many OpenSource solutions, for example OpenSSL and lighttpd.
Sometimes vulnerabilities are found, and a new version is released to mitigate the vulnerability.

The KeyTalk virtual appliance however does not make use of all functionalities of these OpenSource solutions. Therefor it's very possible that a vulnerability might exist is an OpenSource component, but because the feature is not being used nor made usable, there is no immediate need to have it updated.

Ofcourse security and good practice are important to us. So should you find an operational flaw in our server that poses a cybersecurity risk, kindly do let us know. We typically respond within 1 hour to emails.

To report an issue kindly email us at: support@keytalk.com

i) ***My local SysAdmin settings did not sync from the KeyTalk application server to the KeyTalk MySQL cluster***
This is expected behavior. Local SysAdmin settings are not synched across the cluster, only other roles settings are synched in order to prevent a lockout situation due to accidental mistakes.

j) ***KeyTalk is a great product, but it lacks functionality that I'm in need of!!***
Should you have a need for specific functionality, then it's very likely that other companies do too.
To resolve your found issue, let us know and based on your business case KeyTalk IT Security can add the functionality to improve our product and meet your and other future customer needs. Contact us at support@keytalk.com or sales@keytalk.com

k) **I locked myself out of the KeyTalk webbased management, how can I restore the login to default username/password?**
Should a System Administrator ever be locked out of a KeyTalk application server on a GUI level, then login using the KeyTalk CLI admin and run:
/usr/local/bin/keytalk/www/reset-admin-passwd

Should you be locked out of the KeyTalk virtual appliance on a CLI/SSH level, then there is no way to reset the virtual appliance. If this happens, you should retrieve a backup, or restore from a snap-shot.

10. Introduction: KeyTalk certificate & key management and enrolment

KeyTalk is a Certificate Authority vendor neutral PKI certificate&key management and secure enrolment solution. Often referred to as a CLM : Certificate Life Cycle Management solution, or PKI Management Solution or CKMS.

The KeyTalk solution can manage and (re-)enroll certificates and keys securely to end-points under:

- ✓ Its own OpenSSL based private CA
- ✓ A third-party private CA (MS ADCS, EJBCA)
- ✓ Any integrated Qualified Trusted Service Provider (DigiCert, GlobalSign, TrustZone, QuoVadis)

KeyTalk's additional products also enable:

- ✓ Certificate discovery (Requires the KeyTalk SSL Scanner virtual appliance)
- ✓ S/MIME secure email LDAP address-book S/MIME lookup over LDAP(S) and HTTP(S) (Requires the KeyTalk LDAP virtual appliance)

A customer using the KeyTalk solution does not need to choose which of the supported CA's are used.
KeyTalk enables parallel use of all of them for the customer's intended user and/or server and/or IoT, ie the KeyTalk solution is multi-tenant. Kindly ensure you inform us which of the CA platforms you need supported

Because of KeyTalk's unique vendor neutral position, a customer can also change with the flip of a pulldown-menu item the primary CA source for an entire end-point group. Allowing a customer to seamlessly transition from one CA vendor to another without a major impact on the managed end-point community.

Enrolment of certificates and keys is done to client end-points independent upon the network domain these client end-points are in.

Currently KeyTalk supports user end-points: Windows 7-11, MacOSX, various flavors of Linux, iOS 9+, and Android 4.1+.
Server end-points which are supported are: Windows server 2012R2, 2016, 2019, IIS 7-10, including IBM WebSphere, as well as various Linux flavors with Apache and TomCat.

On top of that, KeyTalk supports various LoadBalancers and Mobile Device Management Solutions.

Contrary to our competition, KeyTalk does not rely on Microsoft network domain based "clientless" enrolment, or encrypted email based PFX certificate enrolment, or solely Active Directory, as it restricts the target devices customers wish their certificates to be enrolled to.
Nor does KeyTalk rely solely on 3rd party Mobile Device Management solutions to enroll these certificates, as this would restrict the target audience and end-point devices as well.

Instead KeyTalk primarily relies on its REST API, or KeyTalk app to enroll certificates and keys to user devices, servers and IoT end-points, whereby the KeyTalk solution connects to a customer's existing Identity Provider(s) such as Active Directory, LDAP, Radius (including Vasco, RSA, Gemalto tokens), MySQL, as the customer's trusted Registration Authority. Additionally KeyTalk optionally leverages the authentication with trusted end-point device recognition / preshared secret (a SHA2 hash calculated overall several unique software and hardware components).

The KeyTalk solution contains many features which have grown into our product based on customer demand. As a result, KeyTalk CKMS not only connects to various private CA solutions and (Qualified) Trusted Certificate Service Provider solutions but also has the capability to connect to 3rd party HSMs as well as third party Key Management Systems.

When a customer has a need to connect another 3rd party product not yet covered by us, or needs a feature improvement, we at KeyTalk will gladly discuss your business case and add the needed connectors and/or features to ensure your and our success. Kindly contact us sales@keytalk.com

11. Setting up your KeyTalk private Certificate Authority

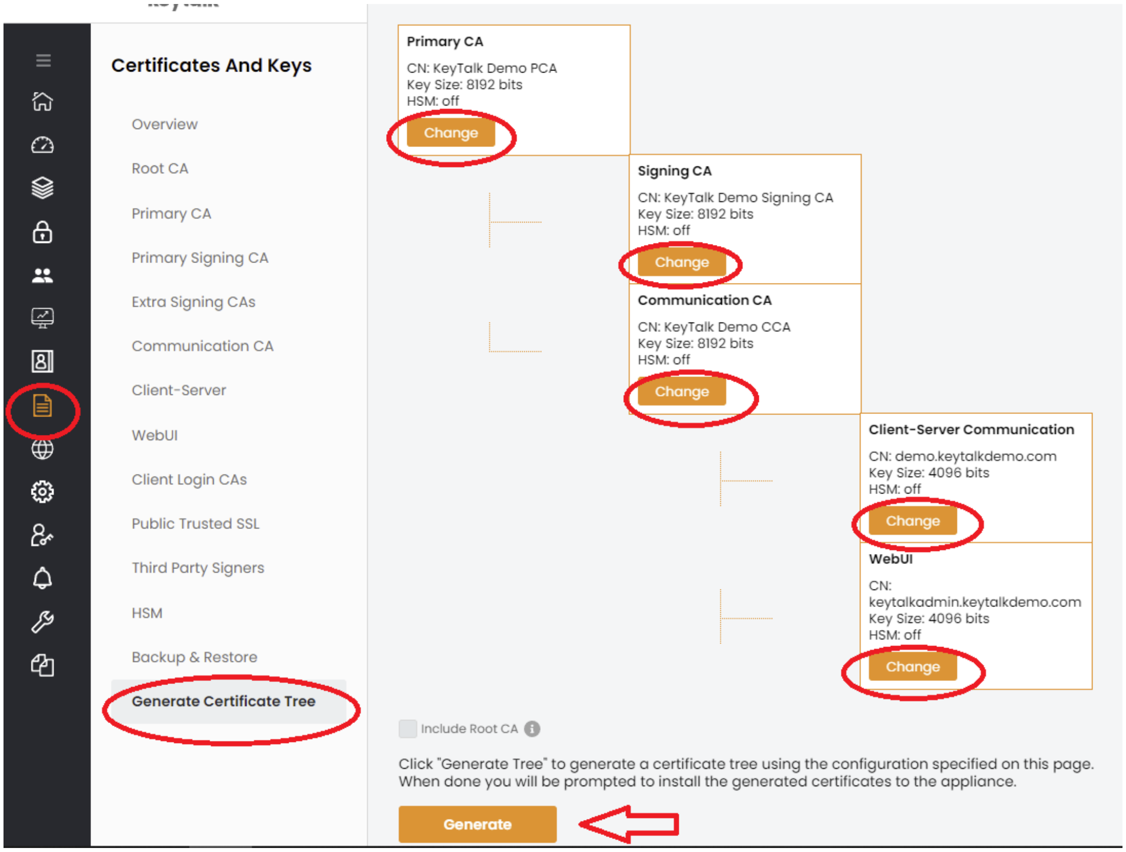
As a default, the KeyTalk solution uses its own OpenSSL based private Certificate Authority for port 443 TLS based secure communication purposes between the virtual KeyTalk application server cluster and the client end-points it serves with end-point-certificates coming from any configured Certificate Authority source, including the KeyTalk private CA. Uploading a publicly trusted certificate and key incl the intermediate CA, overrides the use of the KeyTalk private CA based certificate.

KeyTalk does not enforce the use of an HSM, and stores the KeyTalk private CA key-pairs on the KeyTalk virtual appliance(s). The CA-tree details are not shared through the connected MySQL Db in case KeyTalk is setup as a High Availability cluster.

Supported HSMs can be optionally configured under CERTIFICATES AND KEYS -> HSM , allowing the private keys to be securely generated and stored in a single HSM slot/partition or multiple slots/partitions.

A KeyTalk System Administrator can upload his/her own generated CA-tree, or like most customers, generate their own KeyTalk private CA, including multiple sub-CA's, within minutes using the following steps with our wizard:

1) Login into the KeyTalk Sys Admin webUI and go to: CERTIFCATES AND KEYS -> GENERATE CERTIFICATE TREE



Primary CA
CN: KeyTalk Demo PCA
Key Size: 8192 bits
HSM: off

Signing CA
CN: KeyTalk Demo Signing CA
Key Size: 8192 bits
HSM: off

Communication CA
CN: KeyTalk Demo CCA
Key Size: 8192 bits
HSM: off

Client-Server Communication
CN: demo.keytalkdemo.com
Key Size: 4096 bits
HSM: off

WebUI
CN: keytalkadmin.keytalkdemo.com
Key Size: 4096 bits
HSM: off

☐ Include Root CA

Click "Generate Tree" to generate a certificate tree using the configuration specified on this page. When done you will be prompted to install the generated certificates to the appliance.

KeyTalk uses as its practical root the PrimaryCA, however some customers want to generate the KeyTalk CA under an existing root, in which case you can choose to select: Include Root CA, which first needs to be uploaded in PEM format including the private key under the RootCA-tab, or point the KeyTalk server to an HSM containing the required private key of the root and upload the root certificate in pem format.
When pointing to an HSM in case an existing root needs to be used ensure that the root and its key are referred to as rcacert.pem and rcakey.pem on the HSM and then upload the cert part only to KeyTalk. Ref: See CopyObject call in PKCS11 spec https://www.cryptsoft.com/pkcs11doc/STANDARD/pkcs-11v2-11r1.pdf
To prevent naming convention confusion, KeyTalk therefor uses as its default top CA the naming reference Primary CA even when in most cases its actually the root for practical usage.
Only when an additional Root CA is used will KeyTalk include a Root CA for its private CA

Creating multiple Extra Singing CAs, requires port 80 to be open to distribute these sub-CAs to end-points.

2) CHANGE each of the 5 main parts. Whereby it is recommended to make use of an 8192 bit RSA key at least for the Primary, Signing and Communication CA when you intend on also issuing 8192 end-point certificates. Never choose a key size bigger than its parent, like for example the optional existing Root certificate.

   SHA1 has only been made possible for customers who need legacy support for SHA1, for modern environments SHA1 should otherwise **NOT** be used in production environments!

   End-point devices that obtain their client or server certificate under the KeyTalk private CA, are issued under the Signing CA with its parent Primary CA, so ensure Common Name naming convention wise it makes sense to an end-user or network admin as they will see the Primary and Signing CA in their trusted certificates list.

   The KeyTalk Communication CA is used to generate the KeyTalk client/app (including the REST API) Client-Server Communication certificate and key-pair, as well as the KeyTalk management WebUI certificate and key-pair.
   This KeyTalk Communication CA is also readable to anyone checking their trusted certificates on a device, so again ensure the Common Name makes sense as this is the first description people will read should they know where to look.

   When generating the CA-tree, ensure that the Time to Live meets your demands. Most customers choose a 10-year validity period, (KeyTalk's default) but nothing is stopping you to change this validity period to any other value. Note that end-point 32 bit Operating Systems cannot deal with validities over 18 year.
   Just be mindful of the impact of this choice, such as for example when the KeyTalk private CA requires to be renewed and the new CA has to have its trust pushed to all end-points.

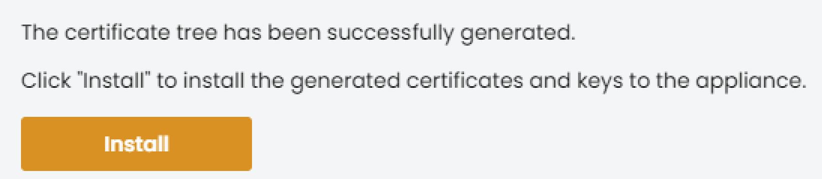
   The minimally required values in the private CA are starred red and to most Sys Admins will make sense. To understand what some of the other values exactly mean, kindly refer to https://www.ietf.org/rfc/rfc5280.txt

3) Review your CA generation settings, preferably have another person review them, and when you agree they are correct, select GENERATE TREE

Client-Server Communication
CA: demo.keytalk.com
Key Size: 2048 bits

Generating tree, this can take several minutes...

4) Once generated select INSTALL to effectuate your new private CA tree



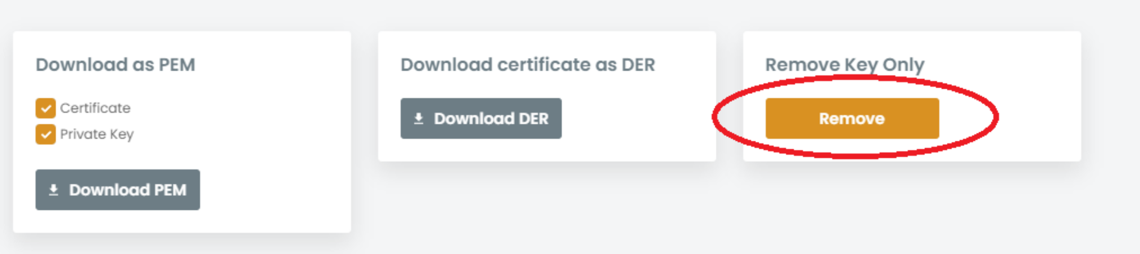
The certificate tree has been successfully generated.

Click "Install" to install the generated certificates and keys to the appliance.

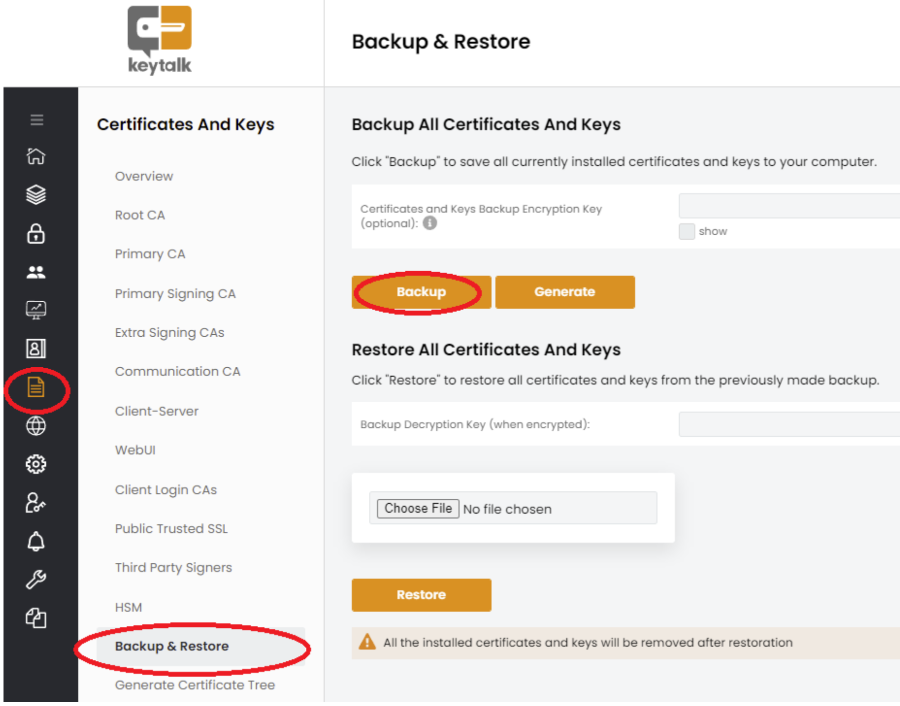
5) When not using an HSM: It is recommended to make multiple backup copies of your Root/PrimaryCA private key and keep these copies securely offline, and then DESTROY the private key of your Root/PrimaryCA

   To backup your private key, its good practice to have a second person observe this process (ie 4-eye principle), and go to: CERTIFICATES AND KEYS -> Root or PrimaryCA -> Download as PEM and incl both cert and priv key.

   To destroy your PrimaryCA and optional RootCA private keys, select CERTIFICATES AND KEYS -> PrimaryCA -> REMOVE KEY ONLY



6) Backup your CA-tree for future usage and ensure the backup is stored on a secure location accessible only to those authorized. Note that when using an HSM this backup will **NOT** contain your CA private keys.



Backup All Certificates And Keys

Click "Backup" to save all currently installed certificates and keys to your computer.

Certificates and Keys Backup Encryption Key (optional):

□ show

Restore All Certificates And Keys

Click "Restore" to restore all certificates and keys from the previously made backup.

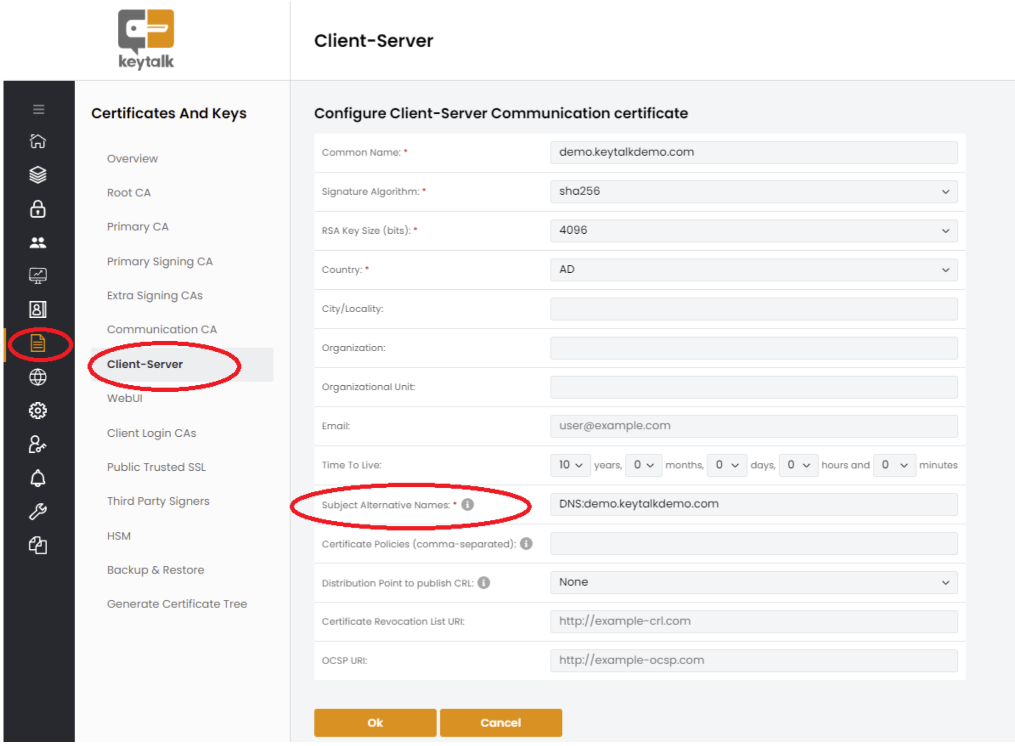
Backup Decryption Key (when encrypted):

No file chosen

All the installed certificates and keys will be removed after restoration

IMPORTANT: The KeyTalk agents and REST API enforce TLS communication. As a result, it is required to have a proper FQDN defined in the SAN value of the KeyTalk Client-Server Communication certificate. (or in the public trust certificate under Trusted Mobile SSL)
Failing to have a proper SAN value will result in TLS communication failure.

Should there be a need to update your KeyTalk Client-Server Communication certificate, simply select REGENERATE and add the proper details. This will result in a new KeyTalk Client-Server Communication certificate signed by the existing KeyTalk Communication CA. After creation don't forget about your other KeyTalk application servers should you run a KeyTalk cluster.
Alternatively upload a Public Trusted SSL certificate with appropriate SAN DNS value



Client-Server

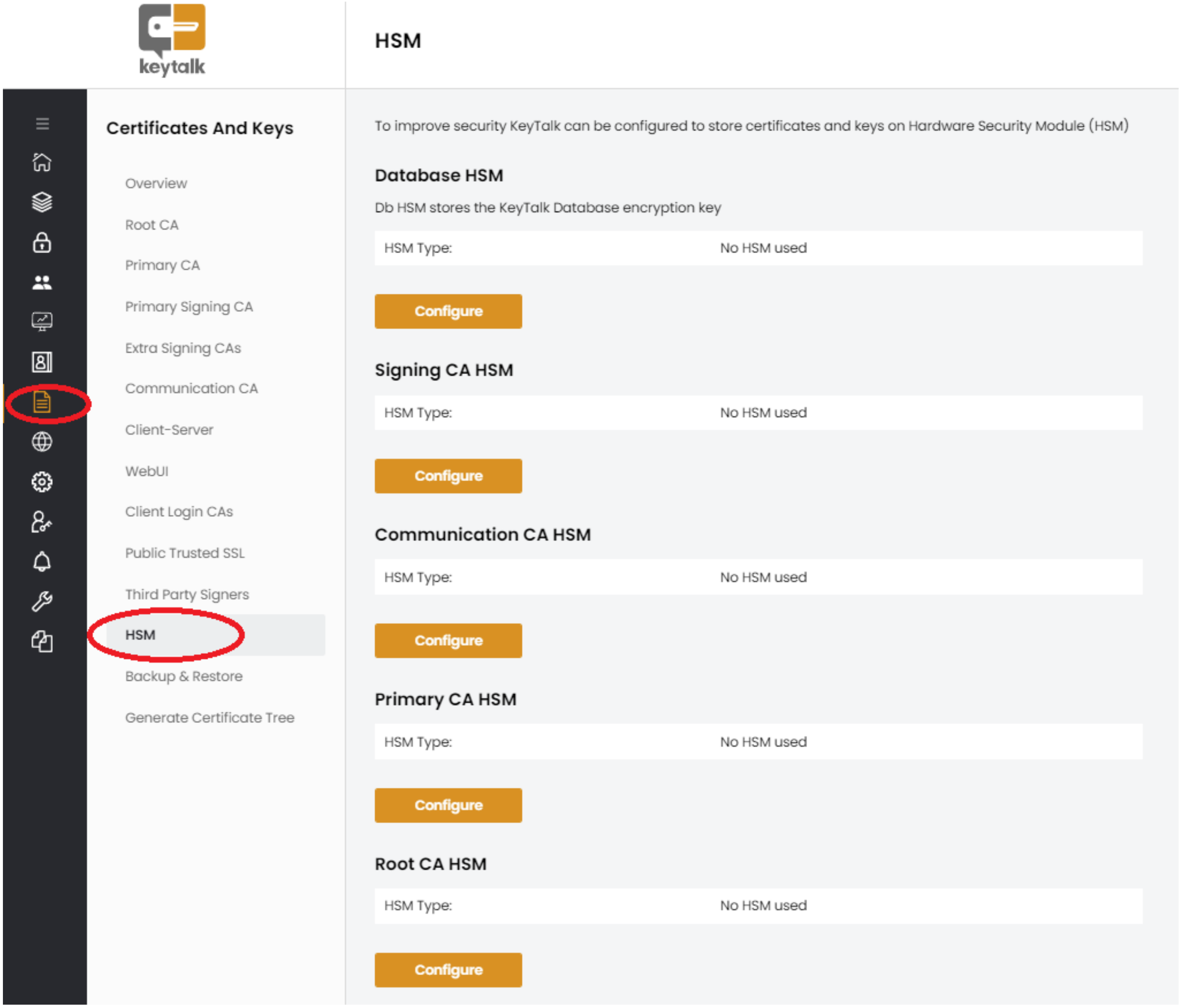
Configure Client-Server Communication certificate

Common Name: *	demo.keytalkdemo.com
Signature Algorithm: *	sha256
RSA Key Size (bits): *	4096
Country: *	AD
City/Locality:	
Organization:	
Organizational Unit:	
Email:	user@example.com
Time To Live:	10 years, 0 months, 0 days, 0 hours and 0 minutes
Subject Alternative Names: *	DNS:demo.keytalkdemo.com
Certificate Policies (comma-separated):	
Distribution Point to publish CRL:	None
Certificate Revocation List URI:	http://example-crl.com
OCSP URI:	http://example-ocsp.com

Example: When your client-Server certificate is meant for https://keytalk.keytalk.com with IPv4 address 178.237.33.242 or using a LoadBalancer on IPv4 address 31.22.80.170, the SAN value could look like: DNS:keytalk.keytalk.com,IP:178.237.33.242,IP: 31.22.80.170

It is good practice **NOT** to use an IP address in the SAN field, and instead use FQDN values that properly DNS resolve. However, in Proof of Concept environments commonly we see solely IP addresses being used hence the example.

11.1 HSM – Hardware Security module



HSM

To improve security KeyTalk can be configured to store certificates and keys on Hardware Security Module (HSM)

Database HSM

Db HSM stores the KeyTalk Database encryption key

HSM Type: No HSM used

Signing CA HSM

HSM Type: No HSM used

Communication CA HSM

HSM Type: No HSM used

Primary CA HSM

HSM Type: No HSM used

Root CA HSM

HSM Type: No HSM used

KeyTalk currently supports PKCS#11 based HSM connections to any integrated HSM, and contains several specific (usually algorithm related) integrations for target types of HSMs.
The current supported HSMs:

- Thales SafeNet Luna HSMs.
- Thales SafeNet Data Protection on Demand (DPoD) cloud HSM
- Utimaco Network HSMs
- Utimaco Cloud HSMs

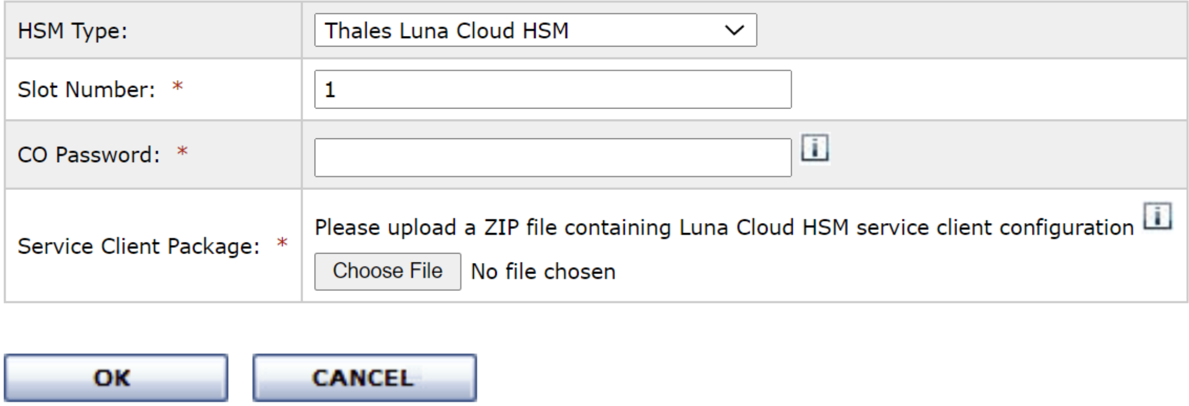
Kindly do let us know when you have another brand/type of HSM you need supported by contacting us.

HSMs are configured per KeyTalk virtual appliance instance. Their settings and passwords are **NOT** shared using the KeyTalk MySQL Db.

The HSM is used to store and make use of CA-tree signing keys, and/or KeyTalk MySQL Db AES encryption key.

11.1.1 Thales Cloud Data Protection on Demand Cloud HSM configuration

Perform these steps for each KeyTalk virtual appliance instance after configuring your DPoD HSM https://thalesdocs.com/dpod/TEMPLATEs/hsmod_TEMPLATEs/hsmod_windows_client/index.html



HSM Type:	Thales Luna Cloud HSM
Slot Number: *	1
CO Password: *	
Service Client Package: *	Please upload a ZIP file containing Luna Cloud HSM service client configuration Choose File No file chosen

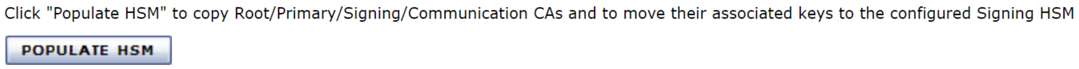
Step 1: Upload your “Luna HSM TEMPLATE client configuration“ file, often referred to as the DPoD or Data Protection On Demand file, as downloaded from your HSMoD TEMPLATE administration page for use with the KeyTalk MySQL Db encryption and/or for use with the KeyTalk private CA-tree

Step 2: Enter your slot/partition password

Step 3: Select OK

The following steps apply when configuring the HSM for KeyTalk internal CA purposes

Step 4: Populate your HSM with the original non-HSM generated CA-tree material by pressing: Populate:

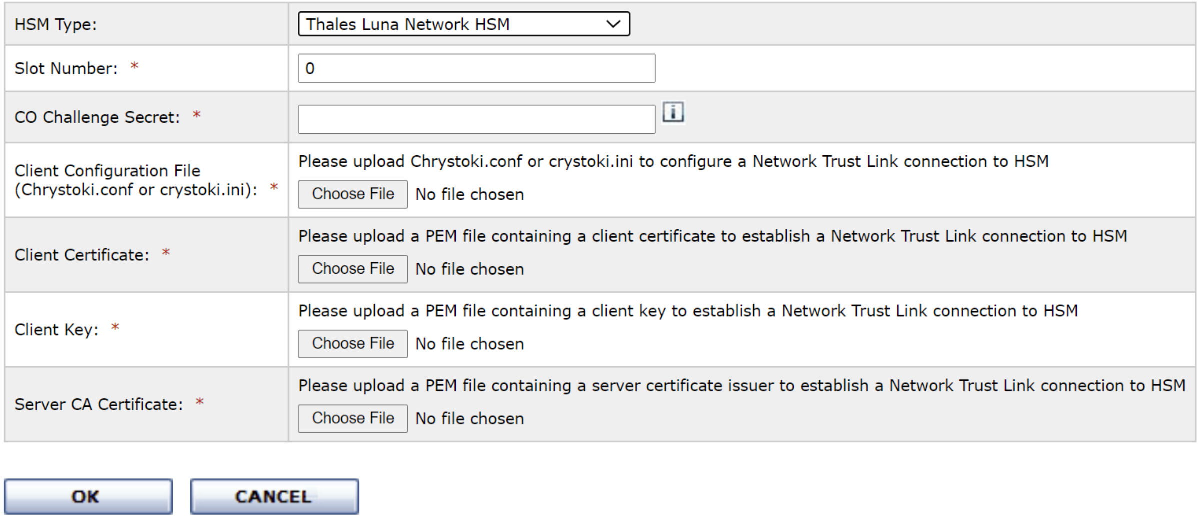


Click "Populate HSM" to copy Root/Primary/Signing/Communication CAs and to move their associated keys to the configured Signing HSM

Step 5: Optionally regenerate your KeyTalk private CA using the KeyTalk “Generate Certificate Tree” wizard to ensure your private CA signing keys only touched the HSM and no other system.

11.1.2 Thales Luna 7 HSM configuration with and without PED

First ensure your Thales Luna HSM has been configured for at least 1 Slot.



HSM Type:	Thales Luna Network HSM
Slot Number: *	0
CO Challenge Secret: *	
Client Configuration File (Chrystoki.conf or crystoki.ini): *	Please upload Chrystoki.conf or crystoki.ini to configure a Network Trust Link connection to HSM Choose File No file chosen
Client Certificate: *	Please upload a PEM file containing a client certificate to establish a Network Trust Link connection to HSM Choose File No file chosen
Client Key: *	Please upload a PEM file containing a client key to establish a Network Trust Link connection to HSM Choose File No file chosen
Server CA Certificate: *	Please upload a PEM file containing a server certificate issuer to establish a Network Trust Link connection to HSM Choose File No file chosen

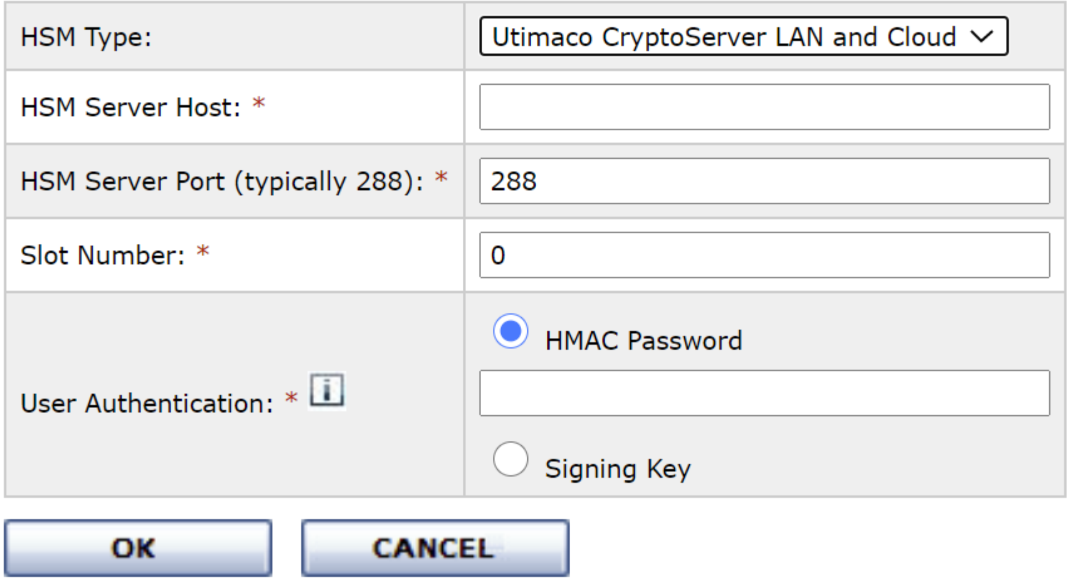
Perform these steps for each KeyTalk virtual appliance instance:

Step 1: Connect from your Windows or Linux environment using the Luna client to your HSM
Step 2: Copy the Chrystoki.conf file from Linux (/etc/Chrystoki.conf) or C:\Program Files\SafeNet\LunaClient\chrystoki.ini on Windows)
Step 3: Copy the client certificate and key as PEM files, as used to securely connect to the HSM from your client
Step 4: Copy the issuing CA in PEM format used for issuing the certificate to SafeNet Luna allowing for the Network Trust Link connection
Step 5: Upload the described files under steps 2-4 in the KeyTalk environment for the appropriate assigned slot/partition and then set the Crypto Officer (CO) challenge/secret/password:
Step 6: Select OK
Step 7: By default Luna HSM is configured to check hostname of NTLS clients connecting to it meaning that a native attempt to connect from KeyTalk server will fail with a mismatched host name error.
The easiest way to fix this HSM-side is to disable NTLS client source IP address check using 'ntls ipcheck disable' command.

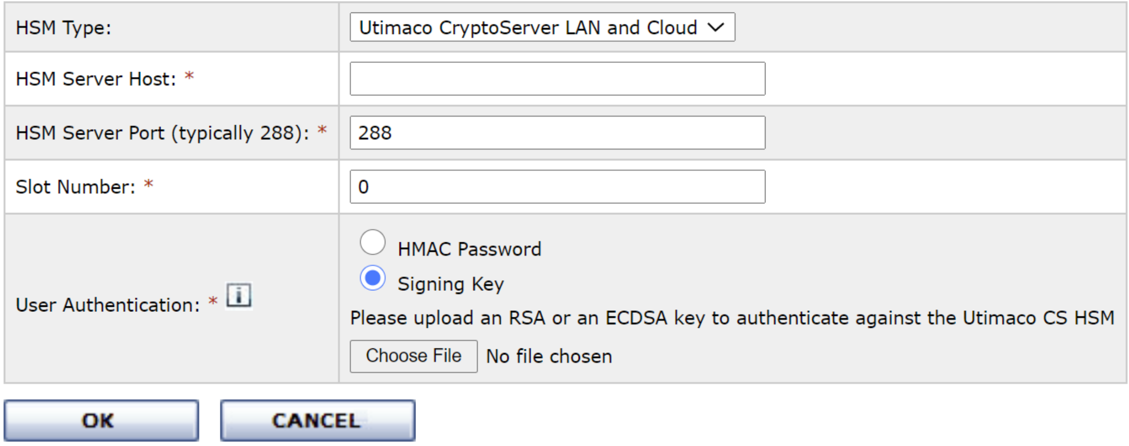
The following steps apply when configuring the HSM for KeyTalk internal CA purposes

Step 8: Optionally regenerate your KeyTalk private CA using the KeyTalk “Generate Certificate Tree” wizard to ensure your private CA signing keys only touched the HSM and no other system.

11.1.3 Utimaco CryptoServer LAN and cloud HSM



HSM Type:	Utimaco CryptoServer LAN and Cloud
HSM Server Host: *	
HSM Server Port (typically 288): *	288
Slot Number: *	0
User Authentication: *	● HMAC Password ○ Signing Key



HSM Type:	Utimaco CryptoServer LAN and Cloud
HSM Server Host: *	
HSM Server Port (typically 288): *	288
Slot Number: *	0
User Authentication: *	○ HMAC Password ● Signing Key Please upload an RSA or an ECDSA key to authenticate against the Utimaco CS HSM Choose File No file chosen

Ensure the proper host, port, slot and authentication credentials are configured and saved (press OK)

11.1.3.1 Utimaco Cloud HSM for Azure hosted KeyTalk CKMS

To connect your Utimaco HSM to KeyTalk (or any other server/TEMPLATE) running from Microsoft Azure, ensure you have an Azure Expressroute circuit configured, which is effectively a security measure to ensure a secure connection.
To create an Azure Expressroute circuit, follow the following steps:

We assume a resource group with virtual network and network security group has already been created in Azure.

1. For the resource group, create a new Virtual network gateway. If it does exist in the main menu, look for it under "more TEMPLATEs". Or click "add" in the resources overview, select "Networking" and then Virtual network gateway.
   1. Click Add

    2. Add a name, select "ExpressRoute" as gateway type, choose the new virtual network, create a new public IP (it's not needed, but it seems we need to create one), make sure subscription and location match, and click Create. If choosing the virtual network is not possible you have to extend the address space (only /24 is created automatically, use e.g. /16).
    3. Modify the outbound security rules
    4. Select the "network security group" resource for your resource group
    5. Click "outbound security rules" and click "Add"
    6. Select source "virtual network", destination "IP Addresses", enter "10.255.0.0/16" as destination IP addresses, enter "*" as destination port and "HSM" as name.
    7. Click Ok.

2. Create a new "ExpressRoute circuit" (maybe available with "More TEMPLATEs").
    1. Click Add - "create new" is already selected
    2. Add circuit name
    3. Select provider "Interxion", peering location "Frankfurt". If not available, select “Amsterdam”.
    4. Select Bandwidth "50 Mbps", SKU "standard", billing model "metered".
    5. Select the same subscription and existing resource group as for your other resources.
    6. Click on Create.
    7. Wait for deployment (1-2 mins)
    8. Click refresh and then on the name of the circuit you've just created.
    9. From the overview page, copy the TEMPLATE key (there is an icon for this)

3. Send KeyTalk support the TEMPLATE key (support@keytalk.com).
   Utimaco will then initiate the connection and KeyTalk will send you the primary subnet information

4. When the provide status of the circuit changed to "Provisioned", complete configuration:
    1. Click on "Azure private", either in the overview page or in "Peerings"
    2. Enter "65000" as peer ASN, primary subnet information, "42" as VLAN ID
        1. We don't use a shared BGP key here since it's a private connection anyway
        2. Leave secondary subnet field empty
        3. Click "Save"
        4. From the circuit menu, select "Connections" and click "Add"
        5. Enter a name, select the Virtual network gateway
        6. Click Ok.

Now, the HSM should be accessible from the Azure KeyTalk Virtual Machine (IP address will be sent with subnet information).

12. Issuing certificates to end-points

After setting up the KeyTalk virtual appliance, generating your unique KeyTalk internal Certificate Authority, and optionally a KeyTalk virtual appliance cluster in combination with the MySQL Db and a load-balancer, certificates need to be enrolled to, and managed for, end-points.

The end-point can have its certificate (and key) deployed using the KeyTalk REST-API, have the KeyTalk agent installed, or make use of KeyTalk CKMS side configured end-point native interface (for example F5 REST-API or Mobile Device Management). KeyTalk agent can be found here: https://keytalk.com/support

The KeyTalk REST API or KeyTalk app is configured with **public information**, ie the KeyTalk Application Server FQDN or IP, one or more KeyTalk TEMPLATE references, and a trust to the KeyTalk private CA-tree. All these requirements are automatically dealt with using the KeyTalk agent and a KeyTalk client configuration file.

To obtain this configuration information, simply have your designated KeyTalk CKMS management user generate this configuration file under TEMPLATES. It's referred to as KeyTalk Real Client Configuration Data or RCCD, and make this configuration file available in any way you see fit, for example from a web-url on your server.
Most company IT departments choose to silently embed this configuration file when deploying the KeyTalk client/app using their corporate deployment solutions.

To request a certificate, a Certificate Signing Request is required and the resulting certificate and key need to be installed in the appropriate key-chain, certificate-store and/or TPM of the client Operating System, and possibly also on a target application running on said target end-point.

KeyTalk automates the CSR generation by generating the CSR on its KeyTalk application server and pushing the signed certificate and keypair in a secure manner in the correct certificate format to the end-point (the default setting). Optionally KeyTalk can send the CSR meta-data to the end-point KeyTalk client/app, have the end-point generate the actual CSR, sign the returned CSR and allow for using an available virtual smartcard on Windows end-points.

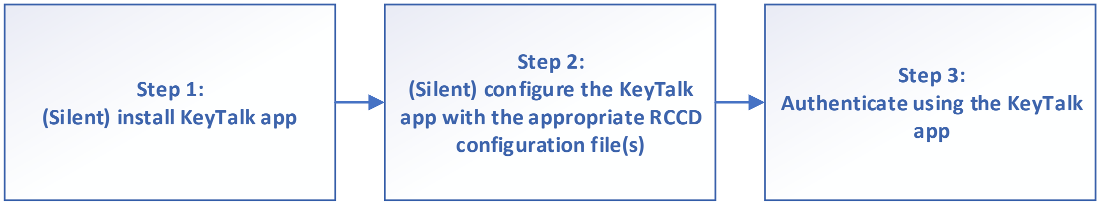
For developers the KeyTalk application server can send the CSR details to the client based on the KeyTalk REST API, where the client/app can generate the CSR and corresponding key-pair, send the CSR to the KeyTalk application server which verifies the CSR content and sends the client an appropriate CA signed certificate. This way the private key is never exposed to a third party in readable format.

In either case, the KeyTalk app will pick up the received certificate (and key-pair), uninstall the old one if it's not an S/MIME certificate, and install the new certificate (and keypair).
When creating your own software using the REST API these (un)installation steps need be coded separately. KeyTalk provides sample code to enable these steps on its Github page.

In order for an end-point to generate a strong crypto key-pair, sufficient entropy is required.
KeyTalk filed and was granted an international patent in 2006 describing a method to generate a key-pair and certificate network-side and push it in a secure manner to an end-point for an initial secure connection.
By generating the CSR network-side with KeyTalk, or choosing to generate it client side, the customer is in full control of the generated keypair entropy.

For compliance purposes you might need private keys to be only generated on end-points, which is supported by default using our API, and can also be supported using the KeyTalk client/app.

To the end point the process based on KeyTalk clients/API to requesting certificates flows like:


Step 1:
(Silent) install KeyTalk app

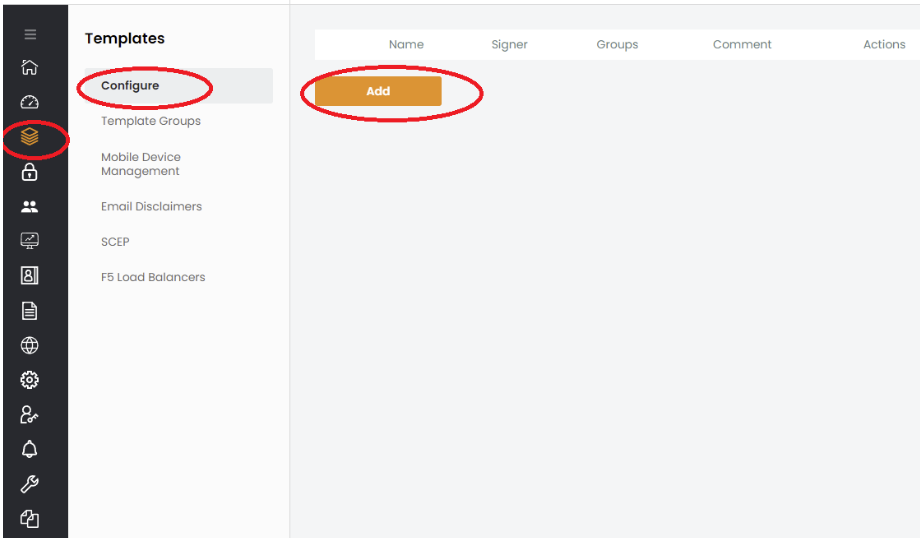
Step 2:
(Silent) configure the KeyTalk app with the appropriate RCCD configuration file(s)

Step 3:
Authenticate using the KeyTalk app

While an end-user will NOT be asked to authenticate manually when he has a domain joined Windows device (ie using Kerberos authentication), most non-Windows device users will need to manually authenticate to request a new certificate when using the KeyTalk app. A server or IoT device can make use of the KeyTalk certificate validity verification scripts which check the renewal parameters every 60 seconds. These renewal parameters are covered in **chapter 13.**

Do note that some end-user Operating Systems allow for seamless certificate (de)installation while others demand some user interaction as indicated by the KeyTalk app or the Operating System.

12.1 KeyTalk TEMPLATE settings: certificate Registration Authority basic settings

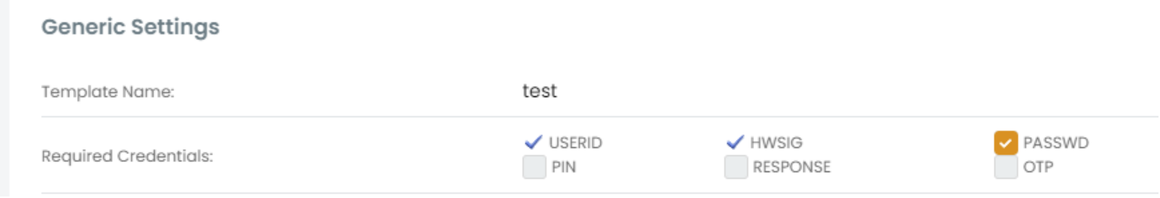


KeyTalk enables the customer Admin to connect a certificate TEMPLATE, CA source, Hardware Recognition, LDAP certificate publishing and CRL/CDP settings to a configured Registration Authority (Identity Provider)

Multiple TEMPLATEs can be used by a single end-point in need of client or server certificates. This way multiple types of certificates can be issued to the same end-point using a single solution.

A TEMPLATE has an alphanumeric name, and is unique to a KeyTalk cluster.
A KeyTalk cluster or single KeyTalk server is also referred to as a "KeyTalk provider".

When you want a TEMPLATE to issue certificates without username/password authentication, simply use the default USERID+HWSIG.
OTP is KeyTalk's internal OTP solution in combination with SMTP and InternalDB RegistrationAuthority



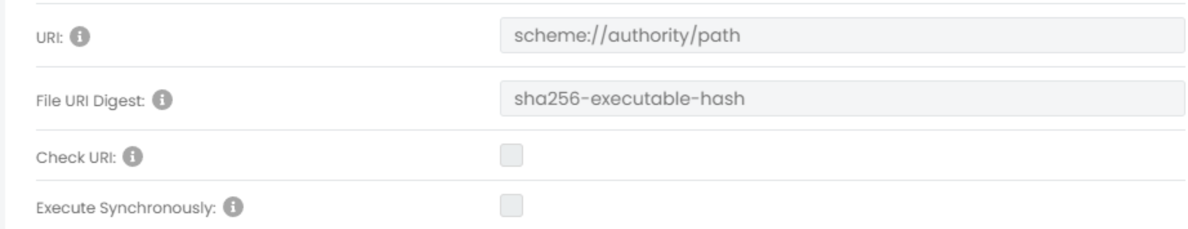
Generic Settings

Template Name: test

Required Credentials: ✓ USERID ✓ HWSIG ☑ PASSWD
□ PIN □ RESPONSE □ OTP

To use an LDAP or Active Directory or OTP token based authentication additionally select PASSWD. PIN is supported, as well as CHALLENGE RESPONSE as provided by some advanced token solutions. When using CHALLENGE RESPONSE do **NOT** select PASSWD or PIN

The URI is used to have KeyTalk client start a specific URI after a new certificate has been obtained.



URI: scheme://authority/path

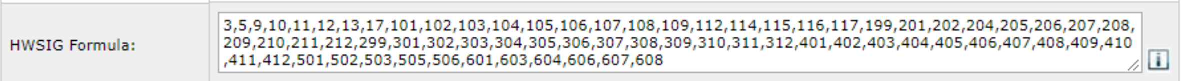
File URI Digest: sha256-executable-hash

Check URI: □

Execute Synchronously: □

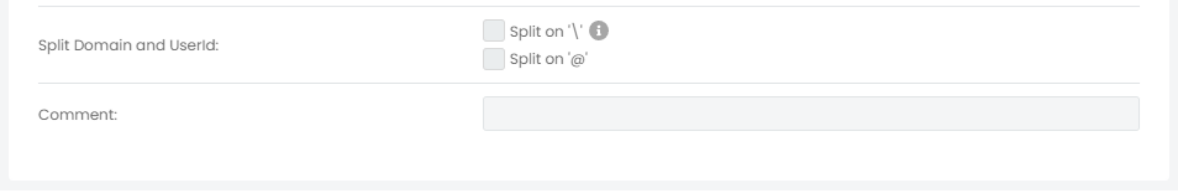
Placing a checkmark in "Check URI" triggers a DNS lookup and matching between the KeyTalk application server and the client/app. When both match the cert is issued, if not a cert wont be issued.

The HWSIG formula defines in what order and how often which components for what Operating Systems need to be made part of the KeyTalk end-point trusted device hardware recognition hash.



HWSIG Formula: 3,5,9,10,11,12,13,17,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,601,603,604,606,607,608

Do note that using for example MAC address is prone to a different value under Wifi or 4G, thus requiring 2 trusted hardware recognition hash slots per user end-point to be configured on KeyTalk when applicable.
The KeyTalk HWSIG is thus a static pre-shared secret meant to make phishing attacks harder.



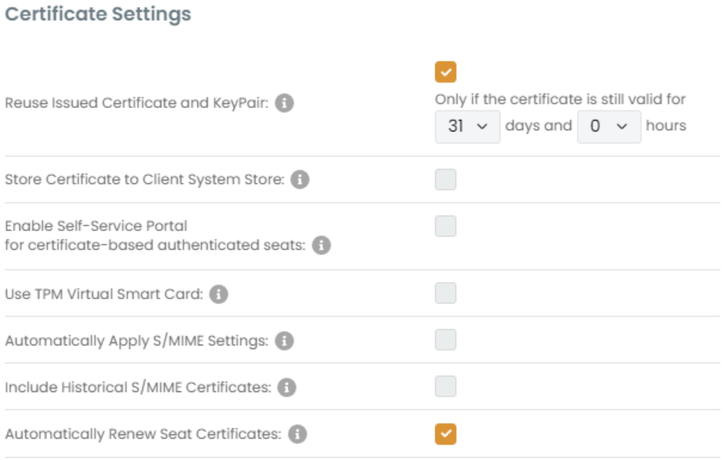
Split Domain and UserId: □ Split on '\' □ Split on '@'

Comment:

Split domain and UserID allows the optional splitting of the username and password on user input before the result is sent to for example the Active Directory BIND.
Do note that the full user input is used in the CN value of the issued certificate unless overwritten with a value coming for example from an AD attribute such as DisplayName.

12.2 KeyTalk TEMPLATE settings: your default certificate source, meta-data, and certificate policies

Within each TEMPLATE you define the basic RA (ie the authentication) requirements but also the required CA source which issues the end-point certificate under the particular TEMPLATE.



Certificate Settings

Reuse Issued Certificate and KeyPair: ☑ Only if the certificate is still valid for 31 days and 0 hours

Store Certificate to Client System Store: □

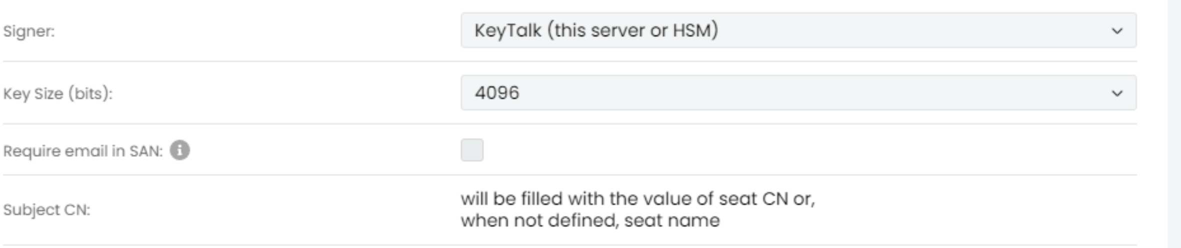
Enable Self-Service Portal for certificate-based authenticated seats: □

Use TPM Virtual Smart Card: □

Automatically Apply S/MIME Settings: □

Include Historical S/MIME Certificates: □

Automatically Renew Seat Certificates: ☑



Signer: KeyTalk (this server or HSM)

Key Size (bits): 4096

Require email in SAN: □

Subject CN: will be filled with the value of seat CN or, when not defined, seat name

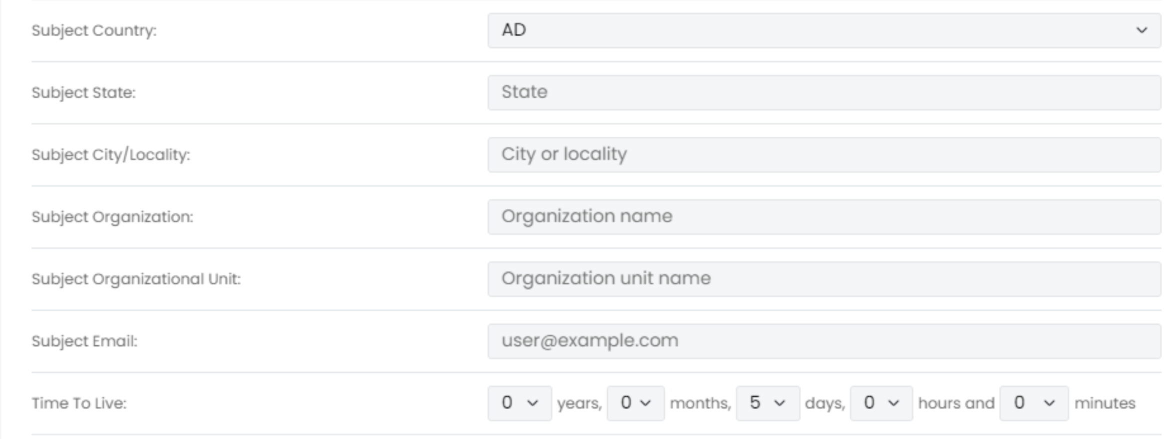
When choosing an external (Qualified) Trusted Certificate Provider, the Q TSP will enforce a specific certificate policy and/or certificate meta-data.

Where-as the KeyTalk private CA and possibly some other 3rd party private CA solutions, will allow you to define your certificate template and policies in a very detailed manner.

Note that connecting your Microsoft CAs (as of Server 2012R2) can only be connected through NDES as KeyTalk is a network domain independent solution and therefore cannot support DCOM.

When using a CSP you **MUST have a vetted account already with the CSP** and often your KeyTalk solution public IP will need to be whitelisted with the CSP, before being able to use it. The KeyTalk solution will not request your account or start its vetting process with the CSP. Depending on the CSP different details must be entered in the TEMPLATE to allow end-point certificate requests.

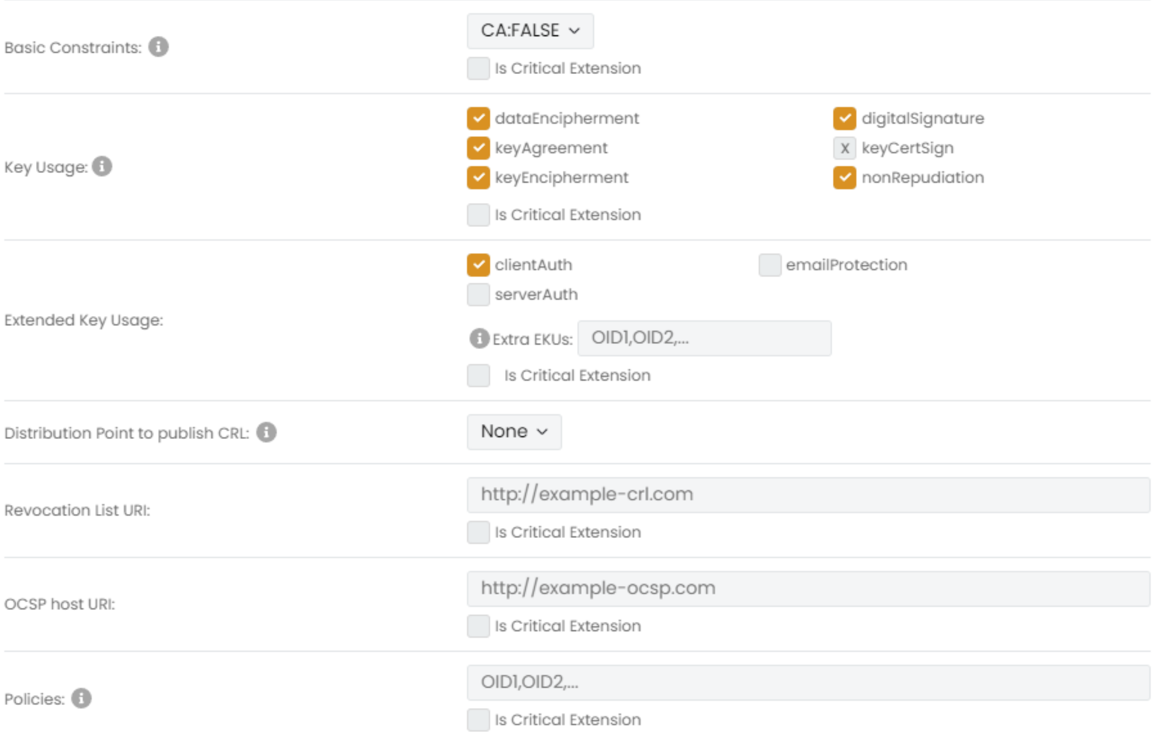
Certificate template meta data is defined in a KeyTalk TEMPLATE provided the CA source allows for it:



Subject Country:	AD
Subject State:	State
Subject City/Locality:	City or locality
Subject Organization:	Organization name
Subject Organizational Unit:	Organization unit name
Subject Email:	user@example.com
Time To Live:	0 years, 0 months, 5 days, 0 hours and 0 minutes

Common Name (CN) is by default derived from the used username to authenticate during a certificate (and key) request. Whereby the username becomes the CN unless otherwise configured in the KeyTalk CKMS. The actual used CN value can also come from the connected RA's Authentication Module account attribute, or be replaced by the used machinename. This is a configuration setting in the connected RA (Authentication Module) to the certificate TEMPLATE.

Certificate policies/extensions are set in KeyTalk using checkmarks for the most commonly used ones:



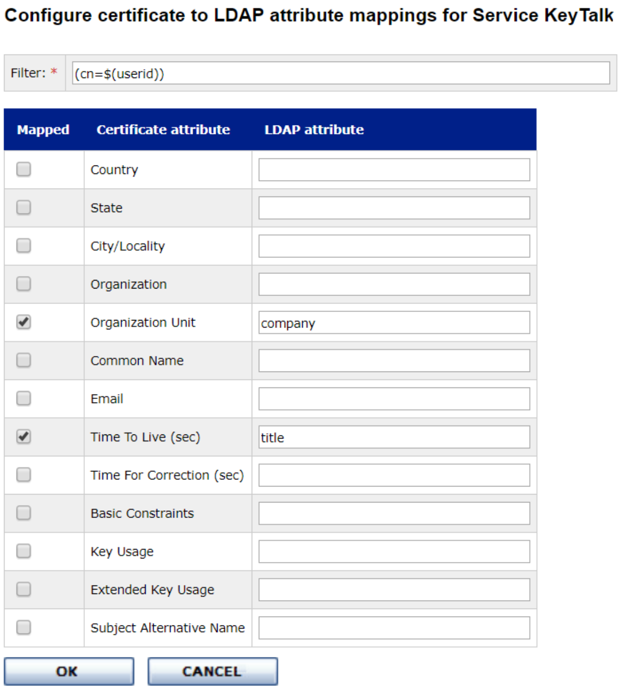
Basic Constraints:	CA:FALSE □ Is Critical Extension
Key Usage:	☑ dataEncipherment ☑ digitalSignature ☑ keyAgreement ☒ keyCertSign ☑ keyEncipherment ☑ nonRepudiation □ Is Critical Extension
Extended Key Usage:	☑ clientAuth □ emailProtection □ serverAuth Extra EKUs: OID1,OID2,... □ Is Critical Extension
Distribution Point to publish CRL:	None
Revocation List URI:	http://example-crl.com □ Is Critical Extension
OCSP host URI:	http://example-ocsp.com □ Is Critical Extension
Policies:	OID1,OID2,... □ Is Critical Extension

Subject Alternative Name values come from the connected RA Authentication Module allowing the separate specification of the CN and SAN.

More advanced Extended Key Usages can simply be included by adding one or more appropriate Object Identifier (OID) separated by comma's
For example, add 1.3.6.1.4.1.311.67.1.1 for use with Bitlocker, add 1.3.6.1.4.1.311.20.2.2 to enable the certificate smartcard property, and/or add 1.3.6.1.4.1.311.54.1.2 for remote desktop authentication. The input in the TEMPLATE would look like: 1.3.6.1.4.1.311.20.2.2,1.3.6.1.4.1.311.67.1.1 ,or you could use the OID formal name.

No matter what choice you make all these values (provided the CSP and 3rd party CA product allows for it), can be overwritten on a unique end-point username level, by making use of the advanced

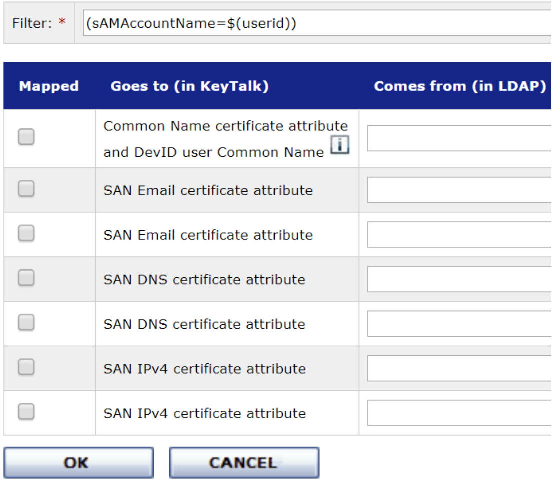
KeyTalk AUTHENTICATION feature: <u>AUTHENTICATION MODULE CERTIFICATE ATTRIBUTE MAPPING for **LDAP/AD or MySQL**</u>**:**



Configure certificate to LDAP attribute mappings for Service KeyTalk

Filter: * (cn=$(userid))

Mapped	Certificate attribute	LDAP attribute
□	Country	
□	State	
□	City/Locality	
□	Organization	
☑	Organization Unit	company
□	Common Name	
□	Email	
☑	Time To Live (sec)	title
□	Time For Correction (sec)	
□	Basic Constraints	
□	Key Usage	
□	Extended Key Usage	
□	Subject Alternative Name	



Filter: * (sAMAccountName=$(userid))

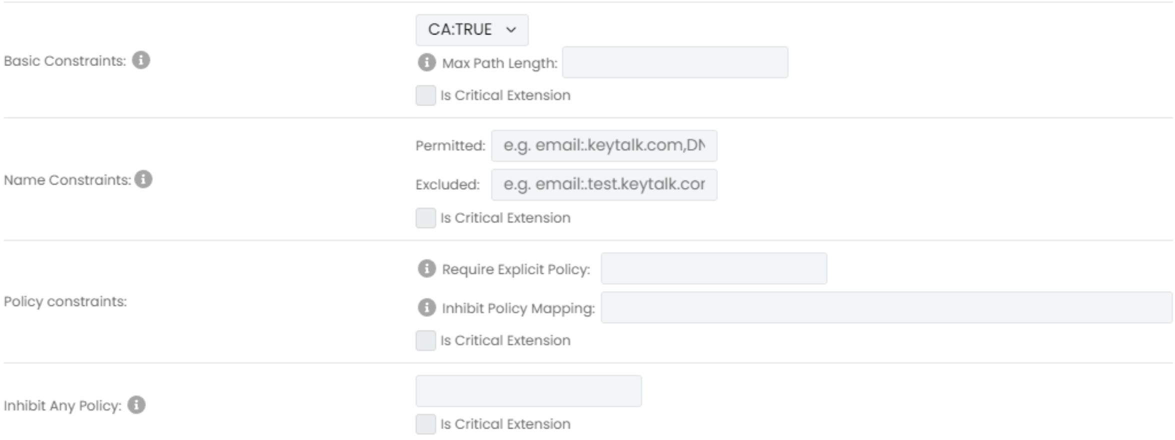
Mapped	Goes to (in KeyTalk)	Comes from (in LDAP)
□	Common Name certificate attribute and DevID user Common Name	
□	SAN Email certificate attribute	
□	SAN Email certificate attribute	
□	SAN DNS certificate attribute	
□	SAN DNS certificate attribute	
□	SAN IPv4 certificate attribute	
□	SAN IPv4 certificate attribute	

In the above example my TEMPLATE default value for Organization Unit is overwritten (provided a valid value exists) with a value coming from my AD/LDAP attribute field "company" for any particular account positively authenticating against the chosen TEMPLATE Authentication.

It's likely that instead of using a **CN** based searchfilter, you may need **sAMAccountName** or **userPrincipalName** depending on your Active Directory configuration.

BASIC CONSTRAINTS by default is always CA:FALSE
This means that no new certificates can be issued under your issued end-point certificate.
<u>**It is highly advised to keep this value always at CA:FALSE!**</u>

Only in rare use-cases should you set the value to TRUE which will enable a plethora of other options familiar to PKI specialists:



Basic Constraints: CA:TRUE
Max Path Length:
□ Is Critical Extension

Name Constraints:
Permitted: e.g. email:.keytalk.com,DN
Excluded: e.g. email:.test.keytalk.cor
□ Is Critical Extension

Policy constraints:
Require Explicit Policy:
Inhibit Policy Mapping:
□ Is Critical Extension

Inhibit Any Policy:
□ Is Critical Extension

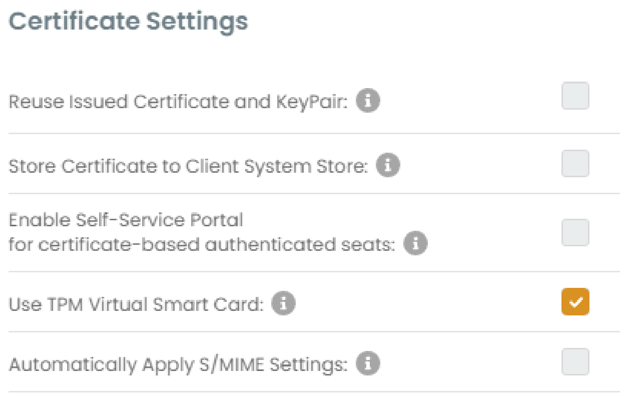
12.3 KeyTalk TEMPLATE settings: set optional TPM or Virtual Smart Card support

By default KeyTalk clients will request KeyTalk server to generate the Certificate Signing Request following our patent(s), and thus the key-pair, on the KeyTalk virtual appliance. This way a company maintains full control over its key management, as well as key generation entropy.

The REST API can be used to enforce client side CSR generation without having to share the private key with the KeyTalk virtual appliance.

Additionally, TPM 2.0 and Virtual Smart Cards are supported, provided a TPM 2.0 is available and the VSC is enabled on Windows end-points. As a result, KeyTalk can NOT store the generated certificate and key-pair for reuse/roll-over to other devices, since the private key remains in the local device TPM.

To enable this feature under TEMPLATES select:



Certificate Settings

Reuse Issued Certificate and KeyPair: □

Store Certificate to Client System Store: □

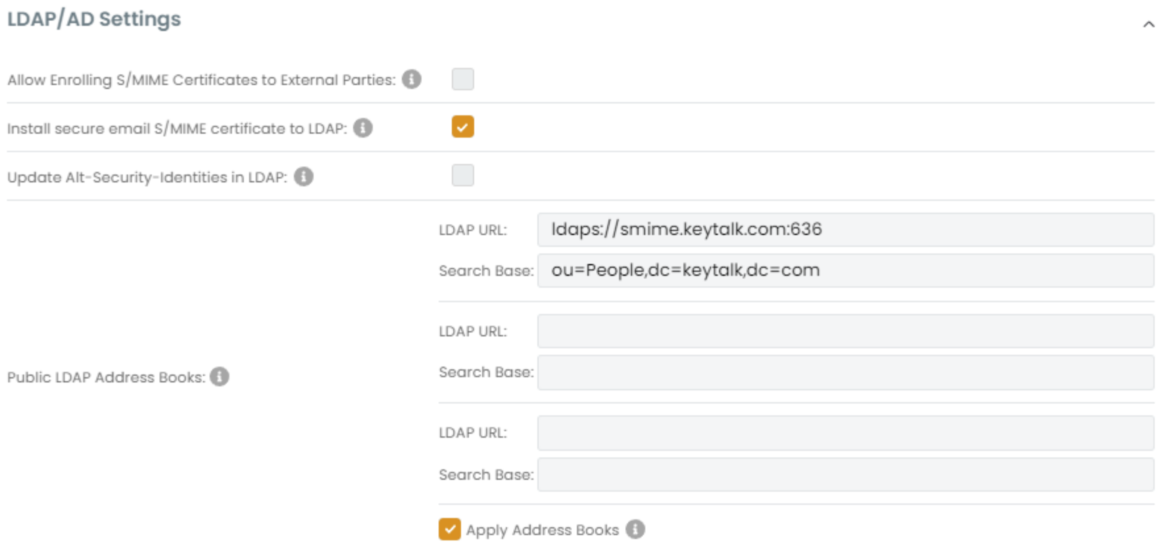
Enable Self-Service Portal for certificate-based authenticated seats: □

Use TPM Virtual Smart Card: ☑

Automatically Apply S/MIME Settings: □

To enable VSC on your Windows 10 client, follow this guide:
https://docs.microsoft.com/en-us/windows/security/identity-protection/virtual-smart-cards/virtual-smart-card-tpmvscmgr
Or use a 3rd party Virtual Smart Card solution.

12.4 KeyTalk TEMPLATE settings: set optional LDAP address book and write certificates to AD



LDAP/AD Settings

Allow Enrolling S/MIME Certificates to External Parties: □

Install secure email S/MIME certificate to LDAP: ☑

Update Alt-Security-Identities in LDAP: □

Public LDAP Address Books:

LDAP URL: ldaps://smime.keytalk.com:636
Search Base: ou=People,dc=keytalk,dc=com

LDAP URL:
Search Base:

LDAP URL:
Search Base:

☑ Apply Address Books

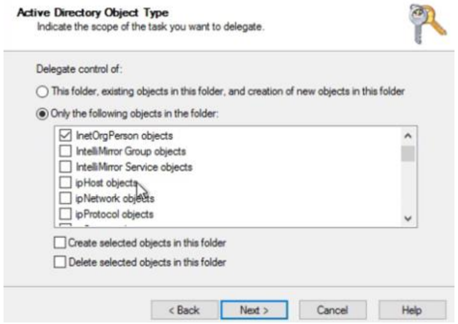
Within each TEMPLATE the KeyTalk Admin can **optionally** set LDAP server details which can be fetched with the REST API to for example configure an external address book in an email client.

To write the certificate (not the private key) to an LDAP and/or an AD “UserCertificate attribute”, checkmark the “Install S/MIME certificate to LDAP” option. Writing the certificate to the target AD/LDAP is done using the requesting user’s authentication details or using a configured AD TEMPLATE user admin account. Similarly removing a certificate that gets replaced/updated is done solely using the configured AD TEMPLATE user admin account.

To write the certificate (not the private key) to an AD “Alt-Security-Identities” attribute (often used for Single Sign On or SharePoint or OWA), checkmark the “Update Alt-Security-Identities in LDAP” option. Writing the certificate to the target AD/LDAP “Alt-Security-Identities“ is done using the KeyTalk LDAP Authentication Module set TEMPLATE User Admin details (an account with elevated rights to lookup user details and write to appropriate attribute, usually a domain admin), as writing values to “Alt-Security-Identities“ is not allowed using regular user rights in most AD configurations.

NOTE: The LDAP Address book settings set in this menu item are only used for public communication. The actual LDAP Address book connector settings are set in the connected LDAP Authentication module as Address book only !!

NOTE: The AD must be configured to allow certificates to be written to it when choosing to also write the issued certificate to it. Though a vanilla/new AD will allow for it, most enterprise AD environments have been configured not to allow for it, so ensure in AD the proper permissions are set.
You can check this in your AD by going to: Delegation of control wizard -> Delegation of control of -> InetOrgPerson objects -> Write Usercert



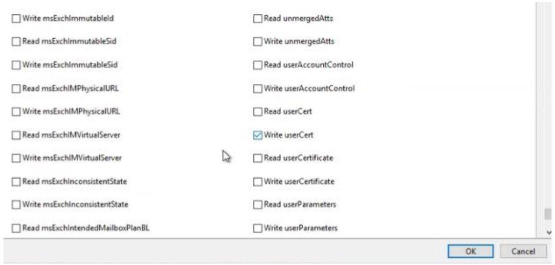
Active Directory Object Type
Indicate the scope of the task you want to delegate.

Delegate control of:

○ This folder, existing objects in this folder, and creation of new objects in this folder
● Only the following objects in the folder:

☑ InetOrgPerson objects
□ IntelliMirror Group objects
□ IntelliMirror Service objects
□ ipHost objects
□ ipNetwork objects
□ ipProtocol objects

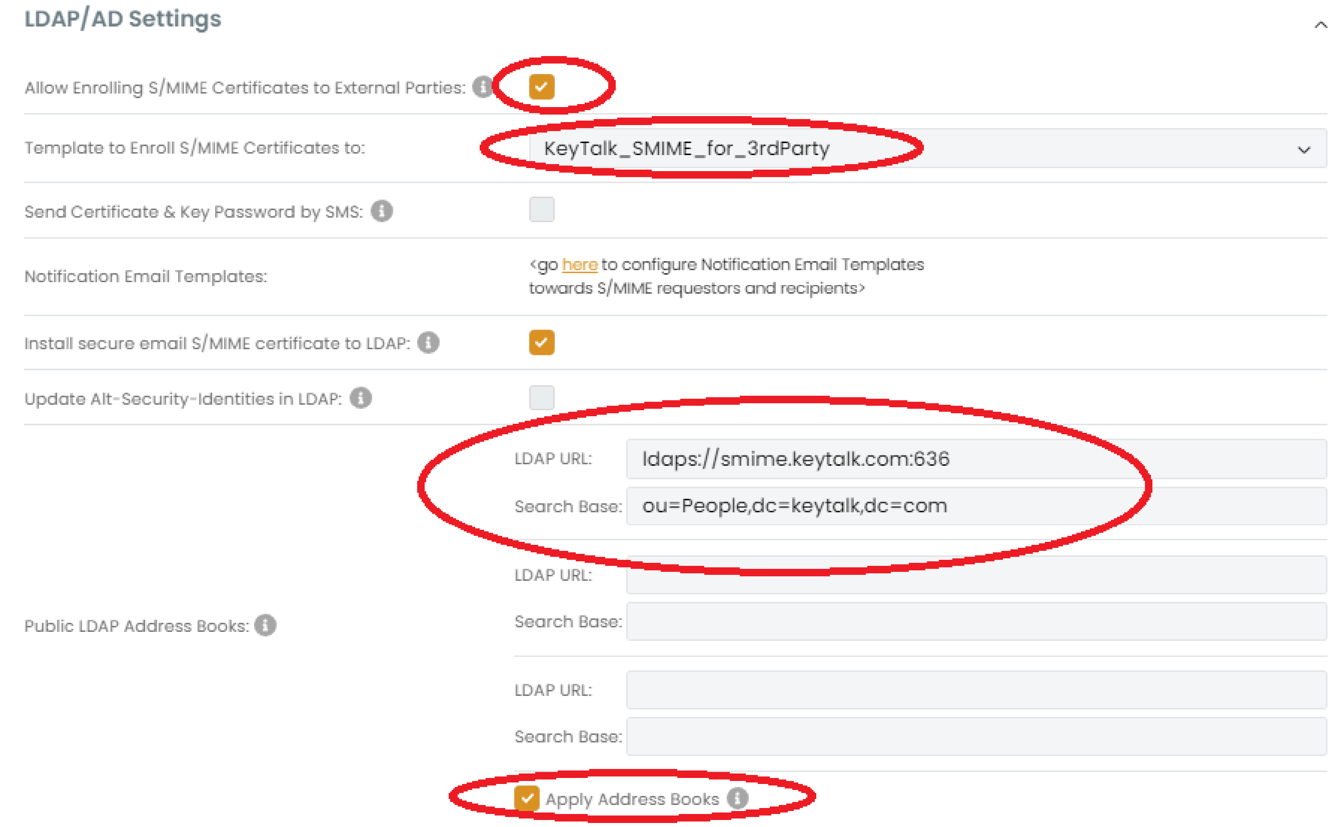
□ Create selected objects in this folder
□ Delete selected objects in this folder



□ Write msExchImmutableId
□ Read msExchImmutableSid
□ Write msExchImmutableSid
□ Read msExchIMPhysicalURL
□ Write msExchIMPhysicalURL
□ Read msExchIMVirtualServer
□ Write msExchIMVirtualServer
□ Read msExchInconsistentState
□ Write msExchInconsistentState
□ Read msExchIntendedMailboxPlanBL

□ Read unmergedAtts
□ Write unmergedAtts
□ Read userAccountControl
□ Write userAccountControl
□ Read userCert
☑ Write userCert
□ Read userCertificate
□ Write userCertificate
□ Read userParameters
□ Write userParameters

12.5 KeyTalk TEMPLATE settings: request and deploy S/MIME certificates to 3rd parties



LDAP/AD Settings

Allow Enrolling S/MIME Certificates to External Parties: ☑

Template to Enroll S/MIME Certificates to: KeyTalk_SMIME_for_3rdParty

Send Certificate & Key Password by SMS: □

Notification Email Templates: <go here to configure Notification Email Templates towards S/MIME requestors and recipients>

Install secure email S/MIME certificate to LDAP: ☑

Update Alt-Security-Identities in LDAP: □

Public LDAP Address Books:

LDAP URL: ldaps://smime.keytalk.com:636
Search Base: ou=People,dc=keytalk,dc=com

LDAP URL:
Search Base:

LDAP URL:
Search Base:

☑ Apply Address Books

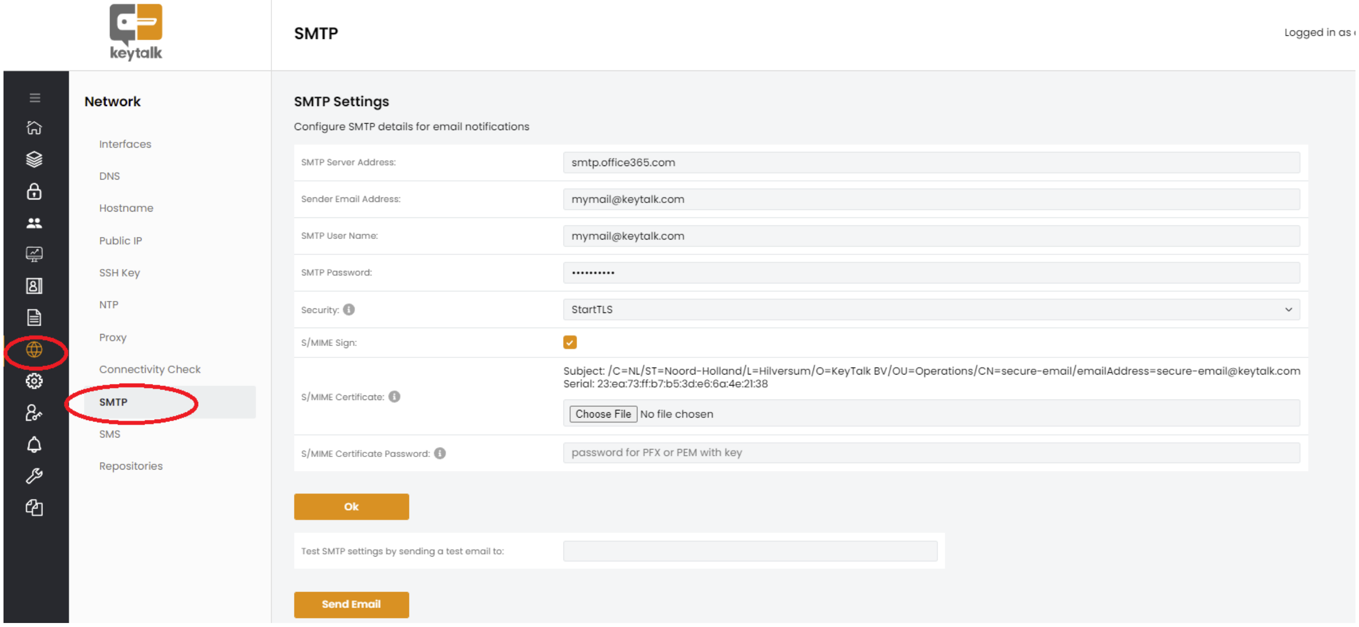
When PKI X.509 certificate based email encryption is required, the recipient must have a known X.509 S/MIME certificate as well, but this is often not the case when communicating outside your company.

Provided that your KeyTalk TEMPLATE is configured to issue certificates that at least support “clientauth” and “emailProtection” to your regular users, you can configure KeyTalk to enable the recipients of the client certificates coming from this TEMPLATE to request S/MIME certificates to third parties with whom they wish to exchange secure email.

By checkmarking “*Allow Enrolling S/MIME Certificates to External Parties*” in the TEMPLATE, and enable the KeyTalk Self TEMPLATE Portal, and configuring in the TEMPLATE at least 1 Public LDAP Address Book, and enforcing reuse certificate, and enforcing certificate based authentication to the KeyTalk virtual appliance, you empower your users to request an S/MIME certificate and keypair for a target recipient.
This S/MIME certificate and keypair are issued from a selected CA in a connected configured KeyTalk TEMPLATE which is tied to an LDAP Authentication Module designated as address book only. Typically the KeyTalk S/MIME secure email LDAP is used for this purpose.

Other than just using your internal CA, KeyTalk IT Security has also made several agreements with Trusted Certificate SERVICE Providers, to enable our KeyTalk customers to request these certificates for third party recipients. For more information on the commercials surrounding these agreements kindly contact sales@keytalk.com .

Several messages will be sent by the KeyTalk virtual appliance to both the requestor and the target recipient.
By default, these messages are sent by email, thus requiring an SMTP server to be configured in the KeyTalk:



SMTP Settings

Configure SMTP details for email notifications

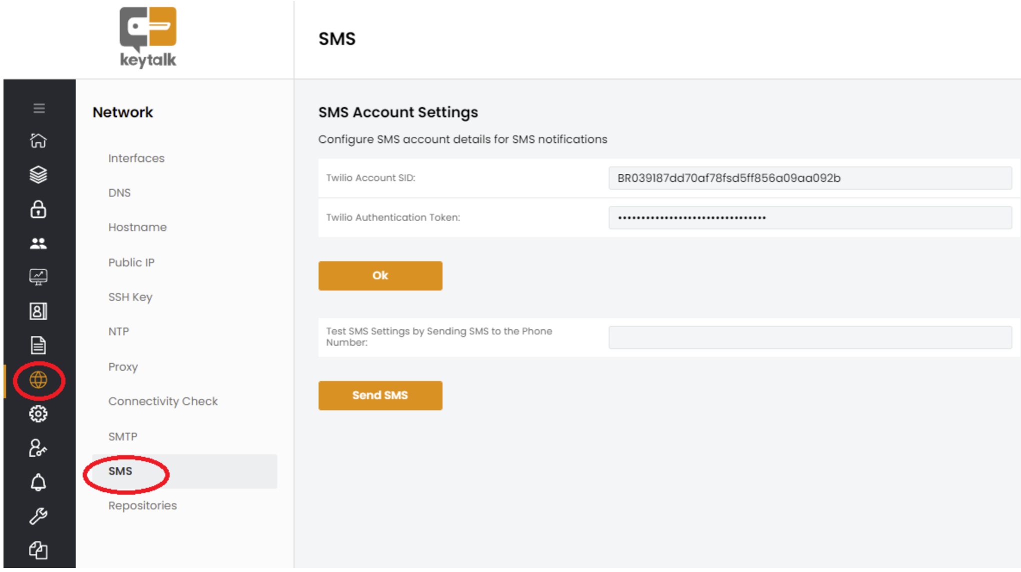
SMTP Server Address:	smtp.office365.com
Sender Email Address:	mymail@keytalk.com
SMTP User Name:	mymail@keytalk.com
SMTP Password:	••••••••••
Security:	StartTLS
S/MIME Sign:	☑
S/MIME Certificate:	Subject: /C=NL/ST=Noord-Holland/L=Hilversum/O=KeyTalk BV/OU=Operations/CN=secure-email/emailAddress=secure-email@keytalk.com Serial: 23:ea:73:ff:b7:b5:3d:e6:6a:4e:21:38 Choose File No file chosen
S/MIME Certificate Password:	password for PFX or PEM with key

Ok

Test SMTP settings by sending a test email to:

Send Email

Additionally, an optional SMS based sidechannel to send passwords separately from email sent downloads links is supported as well. Currently https://www.twilio.com/ is supported as SMS gateway, but other gateways as well as other side-channels, can be integrated depending on your business-case. Kindly contact sales@keytalk.com with your integration needs:



SMS Account Settings

Configure SMS account details for SMS notifications

Twilio Account SID: BR039187dd70af78fsd5ff856a09aa092b

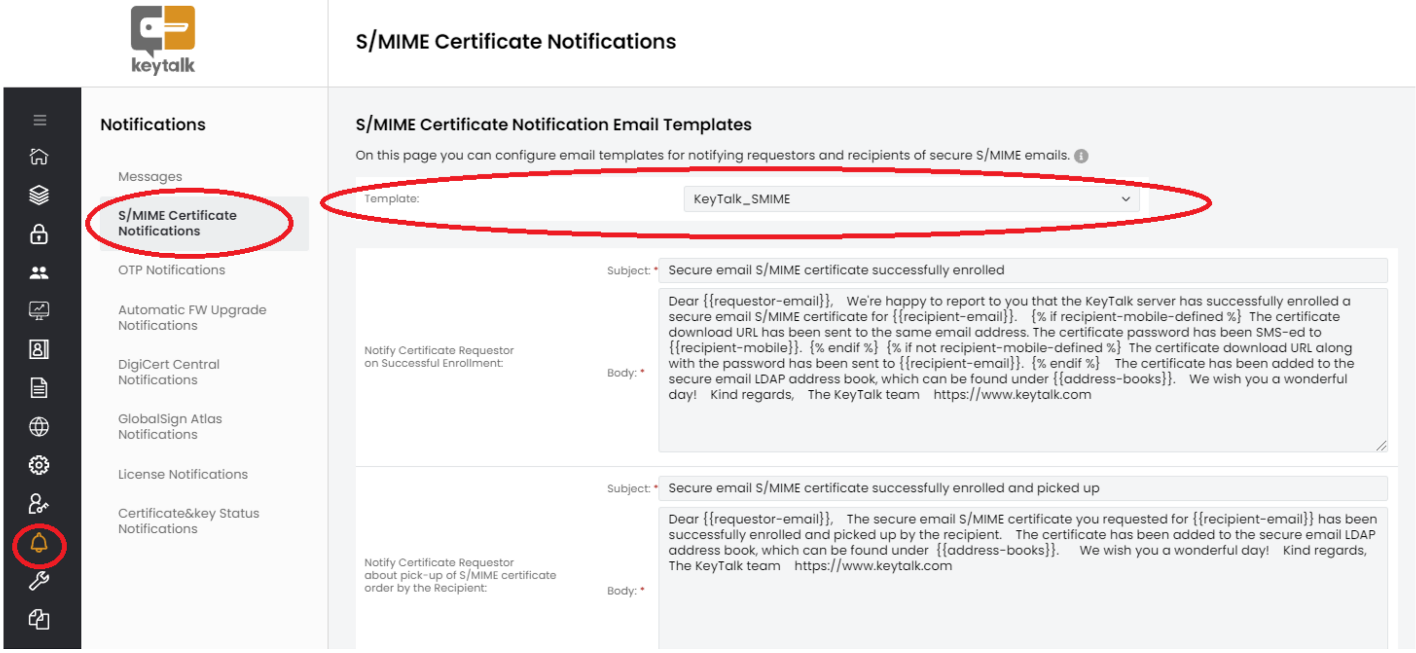
Twilio Authentication Token: ••••••••••••••••••••••••••••••••••

Ok

Test SMS Settings by Sending SMS to the Phone Number:

Send SMS

Messages sent from the KeyTalk virtual appliance are pre-configured, but can be changed and saved as well by the customer per TEMPLATE, under NOTIFICATIONS -> S/MIME certificate Notifications:


keytalk
S/MIME Certificate Notifications
Notifications
Messages
S/MIME Certificate Notifications
OTP Notifications
Automatic FW Upgrade Notifications
DigiCert Central Notifications
GlobalSign Atlas Notifications
License Notifications
Certificate&key Status Notifications

S/MIME Certificate Notification Email Templates

On this page you can configure email templates for notifying requestors and recipients of secure S/MIME emails.

Template: KeyTalk_SMIME

Notify Certificate Requestor on Successful Enrollment:

Subject: * Secure email S/MIME certificate successfully enrolled

Body: * Dear {{requestor-email}}, We're happy to report to you that the KeyTalk server has successfully enrolled a secure email S/MIME certificate for {{recipient-email}}. {% if recipient-mobile-defined %} The certificate download URL has been sent to the same email address. The certificate password has been SMS-ed to {{recipient-mobile}}. {% endif %} {% if not recipient-mobile-defined %} The certificate download URL along with the password has been sent to {{recipient-email}}. {% endif %} The certificate has been added to the secure email LDAP address book, which can be found under {{address-books}}. We wish you a wonderful day! Kind regards, The KeyTalk team https://www.keytalk.com

Notify Certificate Requestor about pick-up of S/MIME certificate order by the Recipient:

Subject: * Secure email S/MIME certificate successfully enrolled and picked up

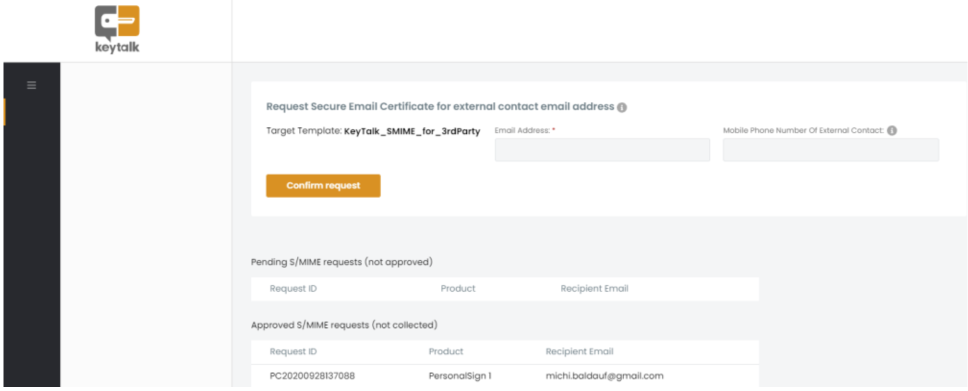
Body: * Dear {{requestor-email}}, The secure email S/MIME certificate you requested for {{recipient-email}} has been successfully enrolled and picked up by the recipient. The certificate has been added to the secure email LDAP address book, which can be found under {{address-books}}. We wish you a wonderful day! Kind regards, The KeyTalk team https://www.keytalk.com

The following parameters can be applied in the messages:

Value	**Description**
{{recipient-email}}	Email address for which the S/MIME certificate and keypair are being requested
{% if recipient-mobile-defined %}	Defined message text IF a mobile number is known as part of the request
{{recipient-mobile}}	Recipient email address in international format
{% endif %}	End of defined IF statement message
{% if not recipient-mobile-defined %}	Defined message text IF a mobile number is NOT known as part of the request
{{address-books}}	Lists the S/MIME LDAP address-book information as defined in the TEMPLATE designated to request third party certificates and keys
{{error}}	Displays the applicable error explaining why a request failed
{{requestor-email}}	Email address of the person requesting the certificate and key-pair for the recipient
{{smime-certkey-download-url}}	Unique download url for the recipients S/MIME certificate and key (usable once)
{{smime-certkey-password}}	Applicable password to install the certificate and key-pair of the recipient
{{smime-certkey-pickup-password}}	Applicable password to pickup/activate a certificate request with a third party CA vendor, such as GlobalSign or DigiCert-QuoVadis
{{ca-provider}}	Reference to the configured CA-source providing the S/MIME certificate to the third party
{{smime-cert-expiration-date}}	Validity end-date of the certificate as issued to the recipient

When a corporate user with a valid S/MIME client certificate access the KeyTalk self service portal, they will be able to request an S/MIME certificate for a target third party recipient, by entering the recipients email address, and optionally (depending on if this is being enforced by the KeyTalk Admin) the recipients mobile number (to receive a pickup or installation password.

The Self Service Portal would look like:



Request Secure Email Certificate for external contact email address

Target Template: **KeyTalk_SMIME_for_3rdParty**

Email Address: *

Mobile Phone Number Of External Contact:

Confirm request

Pending S/MIME requests (not approved)

Request ID	Product	Recipient Email

Approved S/MIME requests (not collected)

Request ID	Product	Recipient Email
PC20200928137088	PersonalSign 1	michi.baldauf@gmail.com

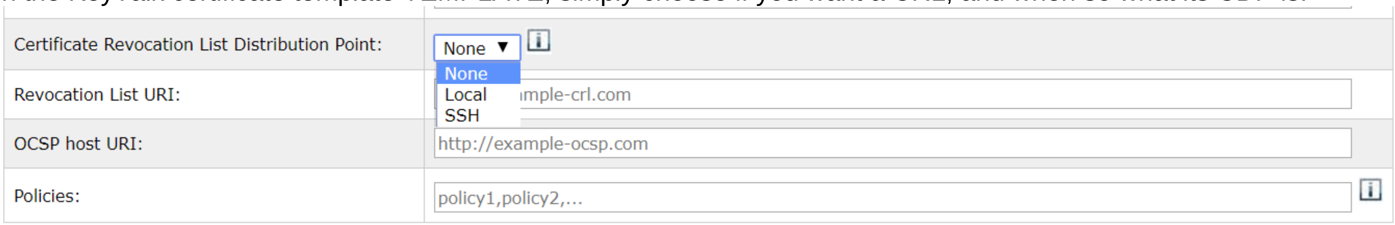
12.6 Certificate Revocation List (CRL) and Distribution Point (CDP)

Revocation Pointers are used to publish revoked certificates.
CSP / CA providers will publish their own RPs, as does the KeyTalk solution for its private CA.
Whenever you revoke a certificate of a configured CSP, then the KeyTalk CLM will transfer the revocation to the CSP thus updating its CDPs.

Short lived authentication certificates (ie authentication certificates with a validity shorted than the average CRL update cycle, practically do not require a CRL or an OCSP.
Still some customers want CRL to be supported for short lived authentication certificates, and other types of certificates independent of their life time.
For this purpose the KeyTalk Certificate Life Cycle Management solution supports CRL, with an update cycle of 1 hour.

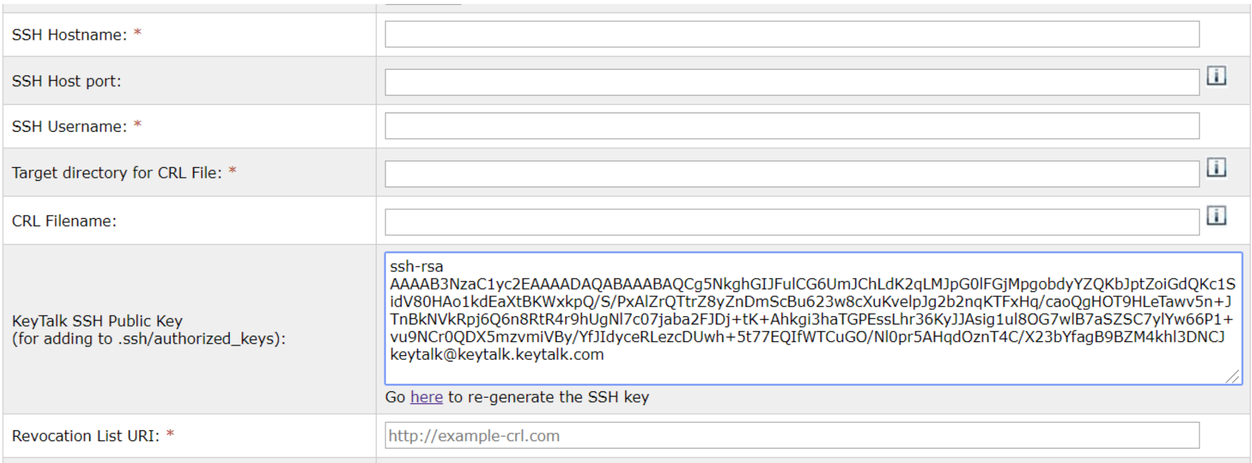
KeyTalk keeps track of a unique CRL for each certificate template TEMPLATE, whereby each CRL can be made available using the KeyTalk CLM as the CRL Distribution Point (CDP), or a target SSH key authentication based server can be configured as a CDP. (should for example SMB or another protocol or authentication be needed kindly contact us).

In the KeyTalk certificate template TEMPLATE, simply choose if you want a CRL, and when so what its CDP is:



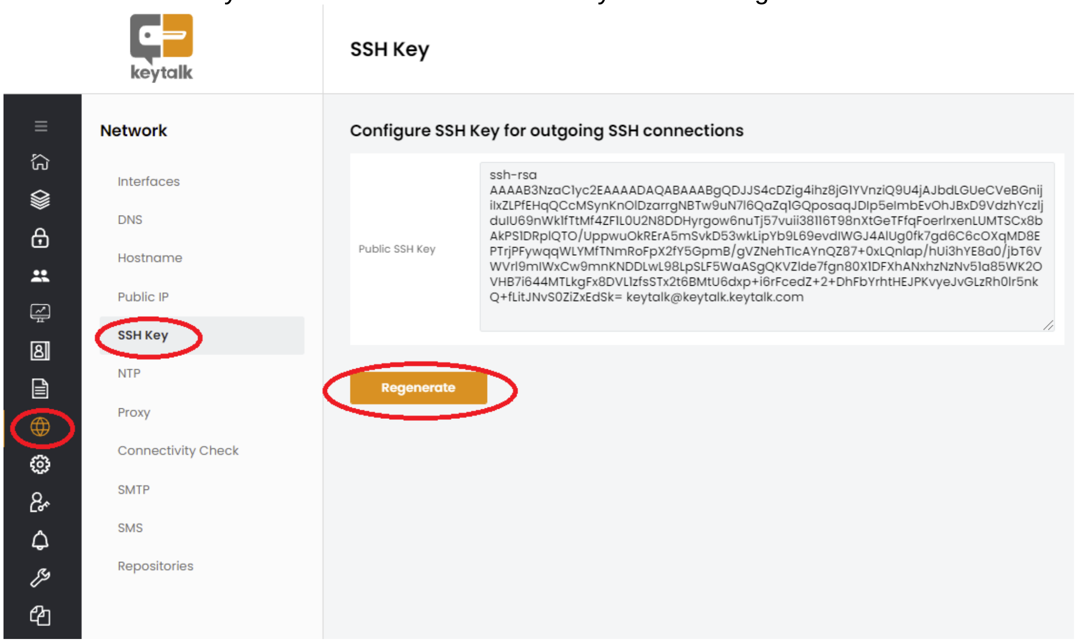
Certificate Revocation List Distribution Point:	None ▼
Revocation List URI:	ımple-crl.com
OCSP host URI:	http://example-ocsp.com
Policies:	policy1,policy2,...

- ➢ None: No CRL will be published
- ➢ Local: The CRL is made available using the KeyTalk server url/IP over HTTP, using the KeyTalk certificate template TEMPLATE name as the crl filename. For example http://keytalkdemo.keytalk.com/myTEMPLATEname.crl
- ➢ SSH: Allows the CRL to be hosted on a target CDP server



SSH Hostname: *	
SSH Host port:	
SSH Username: *	
Target directory for CRL File: *	
CRL Filename:	
KeyTalk SSH Public Key (for adding to .ssh/authorized_keys):	ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCg5NkghGIJFulCG6UmJChLdK2qLMJpG0lFGjMpgobdyYZQKbJptZoiGdQKc1S idV80HAo1kdEaXtBKWxkpQ/S/PxAlZrQTtrZ8yZnDmScBu623w8cXuKvelpJg2b2nqKTFxHq/caoQgHOT9HLeTawv5n+J TnBkNVkRpj6Q6n8RtR4r9hUgNl7c07jaba2FJDj+tK+Ahkgi3haTGPEssLhr36KyJJAsig1ul8OG7wlB7aSZSC7ylYw66P1+ vu9NCr0QDX5mzvmiVBy/YfJIdyceRLezcDUwh+5t77EQIfWTCuGO/Nl0pr5AHqdOznT4C/X23bYfagB9BZM4khl3DNCJ keytalk@keytalk.keytalk.com Go here to re-generate the SSH key
Revocation List URI: *	http://example-crl.com

Ensure that the KeyTalk server does have SSH key credentials generated under:



SSH Key

Configure SSH Key for outgoing SSH connections

Public SSH Key	ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDJJS4cDZig4ihz8jG1YVnziQ9U4jAJbdLGUeCVeBGnij ilxZLPfEHqQCcMSynKnOlDzarrgNBTw9uN7l6QaZq1GQposaqJDIp5eImbEvOhJBxD9VdzhYczlj duIU69nWk1fTtMf4ZFlL0U2N8DDHyrgow6nuTj57vuii38116T98nXtGeTFfqFoerlrxenLUMTSCx8b AkPS1DRplQTO/UppwuOkRErA5mSvkD53wkLipYb9L69evdlWGJ4AlUg0fk7gd6C6cOXqMD8E PTrjPFywqqWLYMfTNmRoFpX2fY5GpmB/gVZNehTlcAYnQZ87+0xLQnlap/hUi3hYE8a0/jbT6V WVrl9mlWxCw9mnKNDDLwL98LpSLF5WaASgQKVZlde7fgn80X1DFXhANxhzNzNv51a85WK2O VHB7i644MTLkgFx8DVL1zfsSTx2t6BMtU6dxp+i6rFcedZ+2+DhFbYrhtHEJPKvyeJvGLzRh0lr5nk Q+fLitJNvS0ZiZxEdSk= keytalk@keytalk.keytalk.com

Configure the SSH access on the CDP server using for example this guide: https://serverpilot.io/docs/how-to-use-ssh-public-key-authentication/

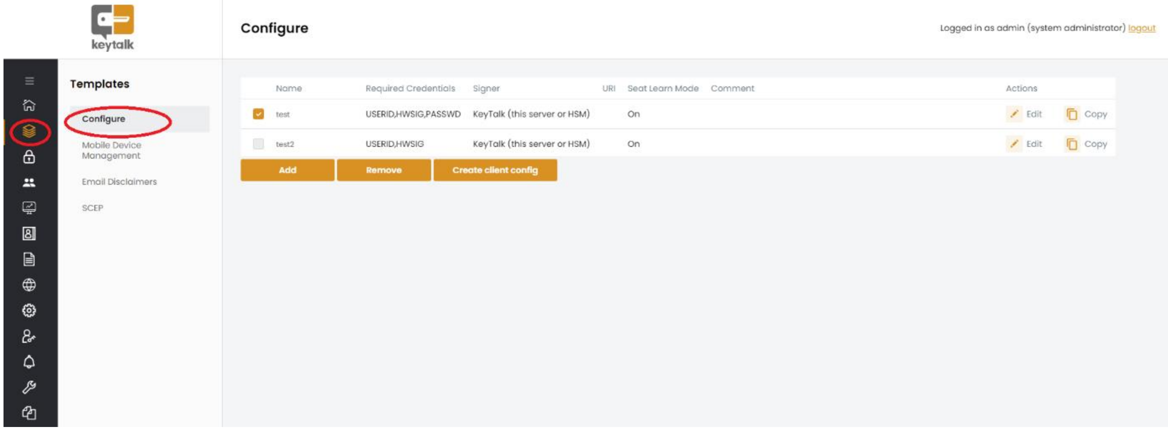
Additional information on KeyTalk's private CA CRL:
When generated the CRL will contain all the revoked, non-expired certificates of the TEMPLATE. It will be signed with KeyTalk server Signing CA key. It contains the datetime when it is generated and the datetime when the next CRL is expected, currently one hour after generating.

If no certificates are revoked, the CRL will show that no certificates are revoked. It will still be generated and published every hour.

Microsoft uses the "nextUpdate" field of a CRL differently than required by the RFC.
Microsoft sees the value in this field as the latest possible validity date of the CRL, meaning that after this date/time no other certificate issued under the connected CA is considered valid.
To prevent this problem from happening, the nextUpdate value is set to 24 hours after generating the CRL despite KeyTalk CLM actually generating the next CRL within 1 hour.

Microsoft uses a custom field named "nextPublish" to communicate when the next CRL update is due. However as this field is not part of the RFC, nor supported by OpenSSL, it is not used or filled by KeyTalk CLM.

12.7 KeyTalk TEMPLATE settings: trusted device slot assignment to end-points and API

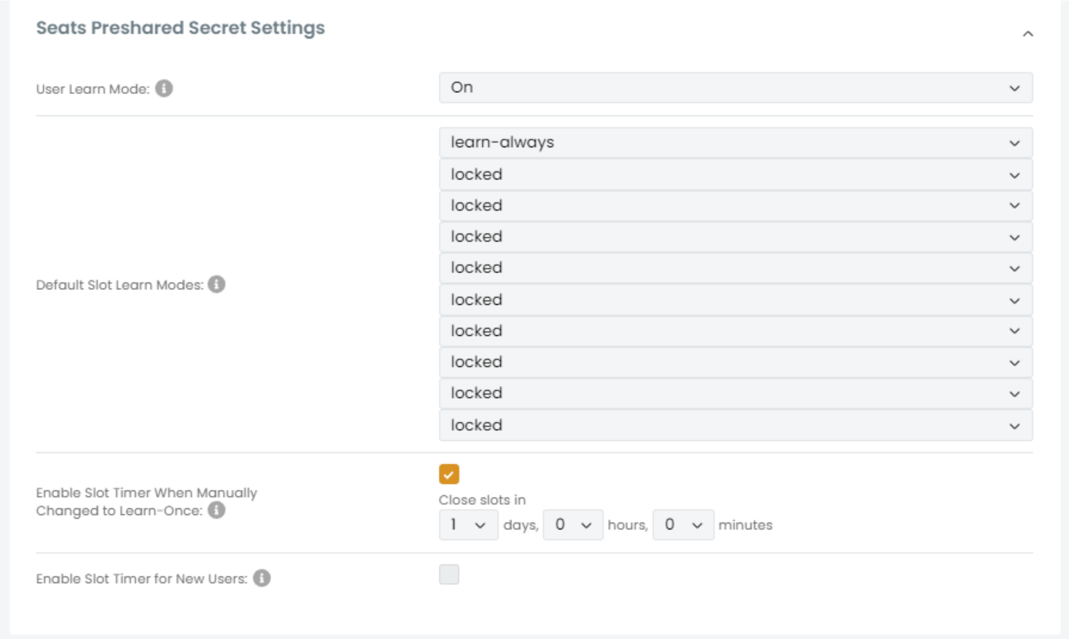


Configure

Logged in as admin (system administrator) logout

	Name	Required Credentials	Signer	URI	Seat Learn Mode	Comment	Actions
☑	test	USERID,HWSIG,PASSWD	KeyTalk (this server or HSM)		On		Edit Copy
□	test2	USERID,HWSIG	KeyTalk (this server or HSM)		On		Edit Copy

Within each TEMPLATE the KeyTalk Admin must also define the amount of trusted device slots an end-point account is allowed to use.



Seats Preshared Secret Settings

User Learn Mode: On

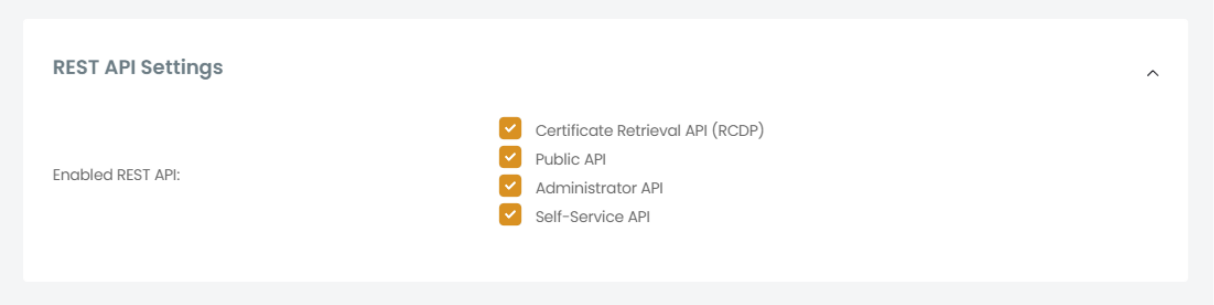
Default Slot Learn Modes:
learn-always
locked
locked
locked
locked
locked
locked
locked
locked
locked

Enable Slot Timer When Manually Changed to Learn-Once: ☑
Close slots in 1 days, 0 hours, 0 minutes

Enable Slot Timer for New Users: □

KeyTalk recommends during initial testing to set User Learn Slot to : ON
And Set the first slot to Learn-always only during initial testing.
In a production state set the slots to **learn-once** so devices able to request a certificate and key are limited
Further details are covered in chapter 14

Each template also allows for its API interface to be blocked or not. For exact specifications kindly refer to KeyTalk's CKMS API documentation
https://downloads.keytalk.com/downloads/documents/KeyTalkApi.pdf



REST API Settings

Enabled REST API:

☑ Certificate Retrieval API (RCDP)
☑ Public API
☑ Administrator API
☑ Self-Service API

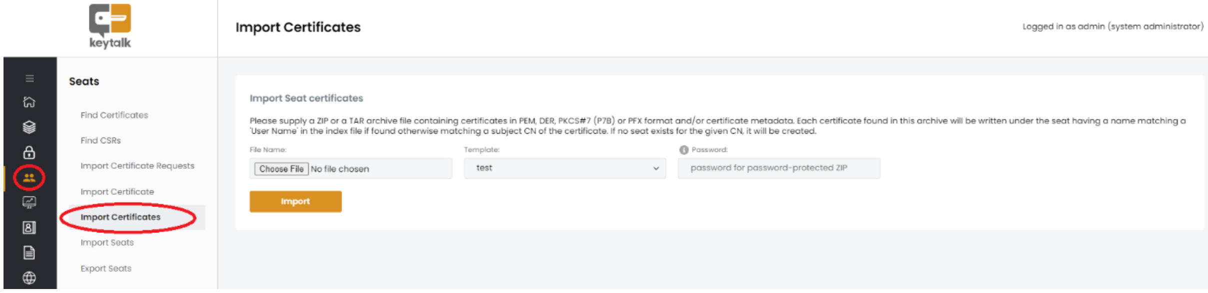
Certificate Retrieval API (RCDP):
Used for TLS based communication to authenticate end-points and fetch requested certificates and keys

Public API:
Used to support RCDP API, sometimes using TLS for secure calls, and sometimes using HTTP to fetch for example CA trusts (to prevent chicken and egg situation)

Administrator API
Used to support (delegated) administrator functions over TLS, allows for admin based functions without use of web UI

Self-Service API
Used to provide self-service portal functions over API

12.8 Certificate renewals versus new certificate requests



Import Certificates

Logged in as admin (system administrator)

Import Seat certificates

Please supply a ZIP or a TAR archive file containing certificates in PEM, DER, PKCS#7 (P7B) or PFX format and/or certificate metadata. Each certificate found in this archive will be written under the seat having a name matching a 'User Name' in the index file if found otherwise matching a subject CN of the certificate. If no seat exists for the given CN, it will be created.

File Name: Choose File No file chosen

Template: test

Password: password for password-protected ZIP

Import

Most (Qualified) Trusted Certificate Service Providers, give their customers an incentive to renew certificates with them.
When the KeyTalk system has been in use from day 1 in combination with the customer's CSP account, there is no need to populate the KeyTalk system with historic certificate order information, as the KeyTalk solution will "know" the order history

However when a customer already has a history with a CSP and starts to use the KeyTalk solution after the fact, then your KeyTalk solution likely needs to be populated with historic order data.

With some CSPs an API can be used to fetch historic certificate data, and with other CSPs its been proven to be hard, thus requiring a manual import.

Use this TEMPLATE certificate order history import feature for described CSPs to enable renewals on existing certificate orders.

KeyTalk will check if a certificate is eligible for renewal, when not a new certificate will be ordered.

Certificates are eligible for renewal when their CN value matches a past valid non-revoked order, and the certificate hasn't expired beyond its renewal date

12.9 About Common Name (CN) and Subject Alternative Name (SAN) in certificates

The KeyTalk CKMS by default assumes no relationship between CN and SAN ever since the CN value became deprecated as a fallback in the year 2011 in RFC 6125.

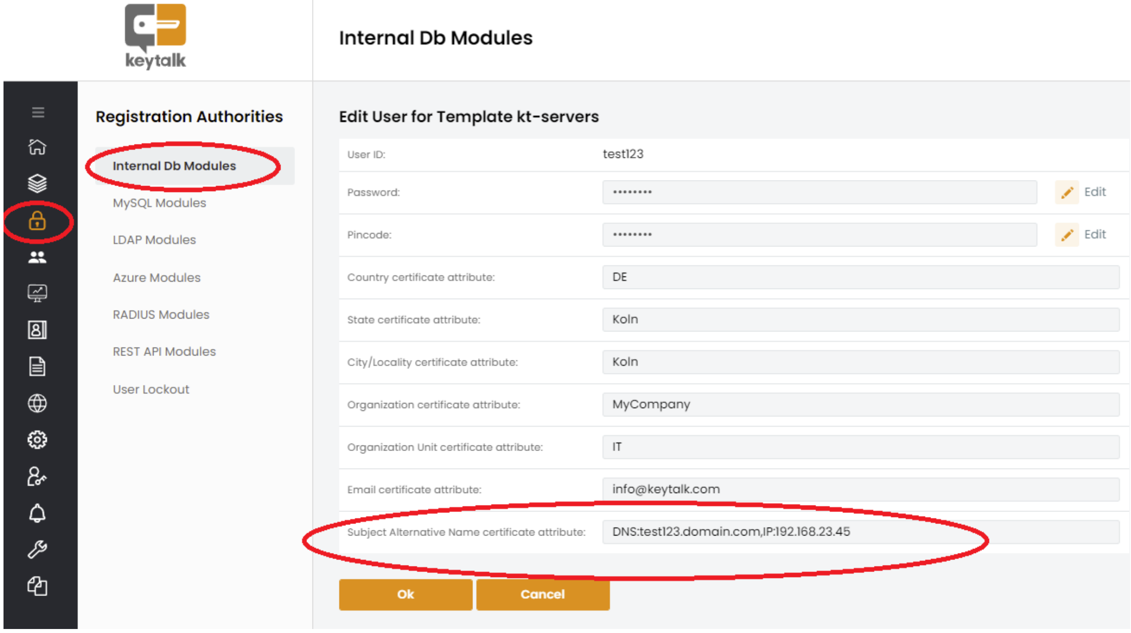
A username/id of an end-point when authenticating to fetch a certificate, is the default value for the CN, unless otherwise configured in the connected RA and its CN attribute mapping.

Similarly by default the username/id will not be copied automatically into the SAN as a DNS value (with or with or a domain).

In order to fill the SAN for any given end-point certificate, it is therefor required to specify the SAN value in the KeyTalk CKMS configured **R**egistration **A**uthority.

The SAN value is obtained from an attribute of the RA, for example the dNSHostName attribute in Active Directory for server certificates, or the mail attribute in an Active Directory in an S/MIME certificate (allowing the CN to be the (full) (DisplayName) of a user while the SAN has proper email address.

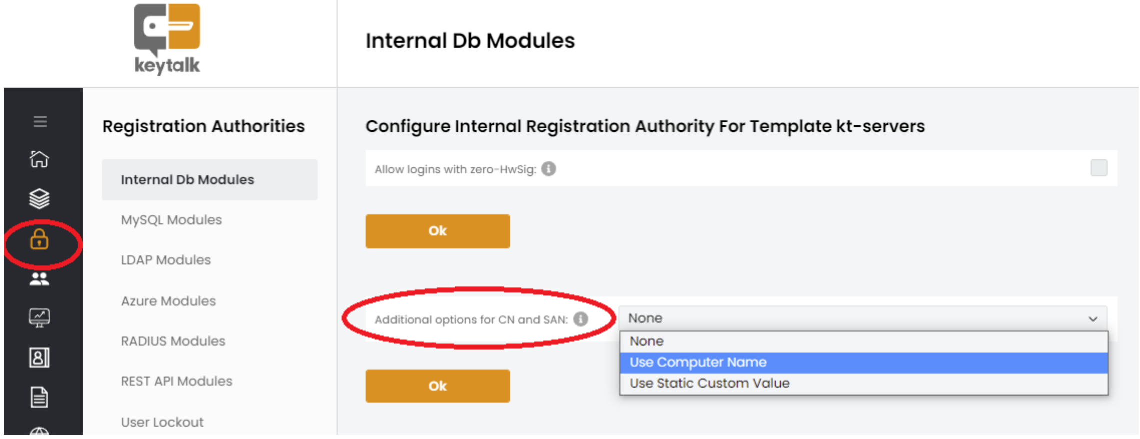
KeyTalk CKMS supports multiple values for SAN DNS, rfc822name, and IP. Simply add these values in the connected RA and mapped attribute(s).
When using the KeyTalk CKMS InternalDB as the connected RA, all SAN values can be entered per end-point. As an example:



Edit User for Template kt-servers

User ID:	test123
Password:	••••••••
Pincode:	••••••••
Country certificate attribute:	DE
State certificate attribute:	Koln
City/Locality certificate attribute:	Koln
Organization certificate attribute:	MyCompany
Organization Unit certificate attribute:	IT
Email certificate attribute:	info@keytalk.com
Subject Alternative Name certificate attribute:	DNS:test123.domain.com,IP:192.168.23.45

Additionally, the KeyTalk CKMS allows the CN and the SAN to be filled with non-RA based fetched values.



Configure Internal Registration Authority For Template kt-servers

Allow logins with zero-HwSig: □

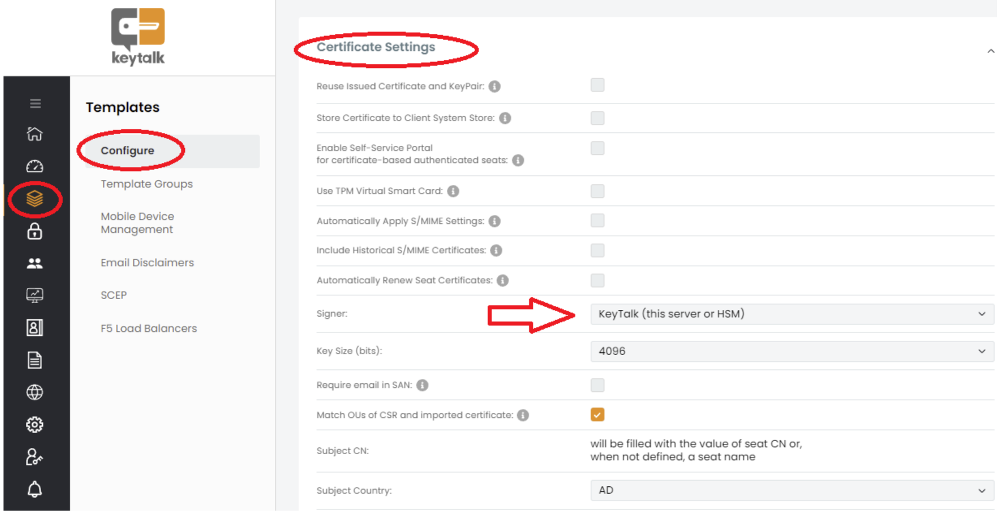
Additional options for CN and SAN: None

None: The default, ie CN is the used username/id, or the mapped value of the RA to the CN
Computer Name: After a positive username/id authentication, copies the value of the used device ComputerName into the CN and adds the computername as a SAN DNS entry. Typically used to issue machines certificates for 802.1x purposes to any device a corporate account (person) authenticated positively from
Static Cust Value: After a positive username/id authentication, copies the given static value into the CN and adds it as a SAN DNS value

12.10 Supported CA’s / TSPs

KeyTalk CKMS supports various CA sources for certificate issuance / CSR signing purposes.

Select a CA source per certificate TEMPLATE in the drop down.



Reuse Issued Certificate and KeyPair: □

Store Certificate to Client System Store: □

Enable Self-Service Portal for certificate-based authenticated seats: □

Use TPM Virtual Smart Card: □

Automatically Apply S/MIME Settings: □

Include Historical S/MIME Certificates: □

Automatically Renew Seat Certificates: □

Signer: KeyTalk (this server or HSM)

Key Size (bits): 4096

Require email in SAN: □

Match OUs of CSR and imported certificate: ☑

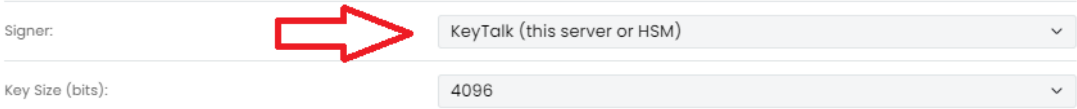
Subject CN: will be filled with the value of seat CN or, when not defined, a seat name

Subject Country: AD

12.10.1 *KeyTalk Private CA*

KeyTalk’s private CA offers a 2 or 3 tier CA hierarchy for issuing end-point certificates (and keys).

In a certificate TEMPLATE, select KeyTalk:



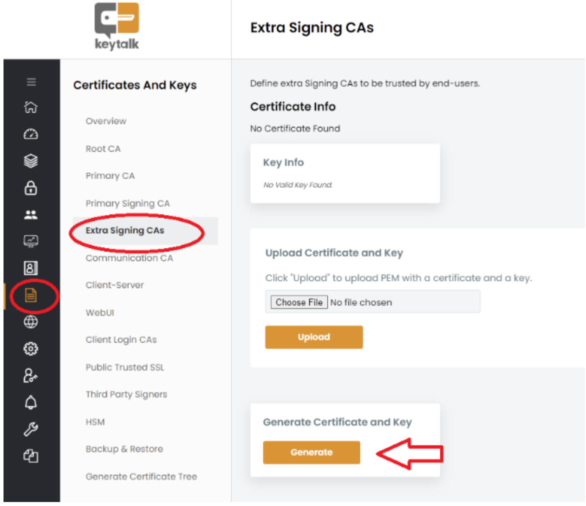
Signer: KeyTalk (this server or HSM)

Key Size (bits): 4096

The default KeyTalk CA trustchain assumes a 2 tier setup, ie Primary -> Primary Issuing CA, but can be extended to include a Root -> Primary -> Primary Issuing CA.

While the Primary Issuing CA can be changed and revoked, it should never be deleted!

Under the Primary CA, multiple Extra Issuing CAs can be created, after which they can be selected from the certificate TEMPLATE signer selection dropdown:



Extra Signing CAs

Define extra Signing CAs to be trusted by end-users.

Certificate Info

No Certificate Found

Key Info

No Valid Key Found

Upload Certificate and Key

Click "Upload" to upload PEM with a certificate and a key.

No file chosen

Generate Certificate and Key

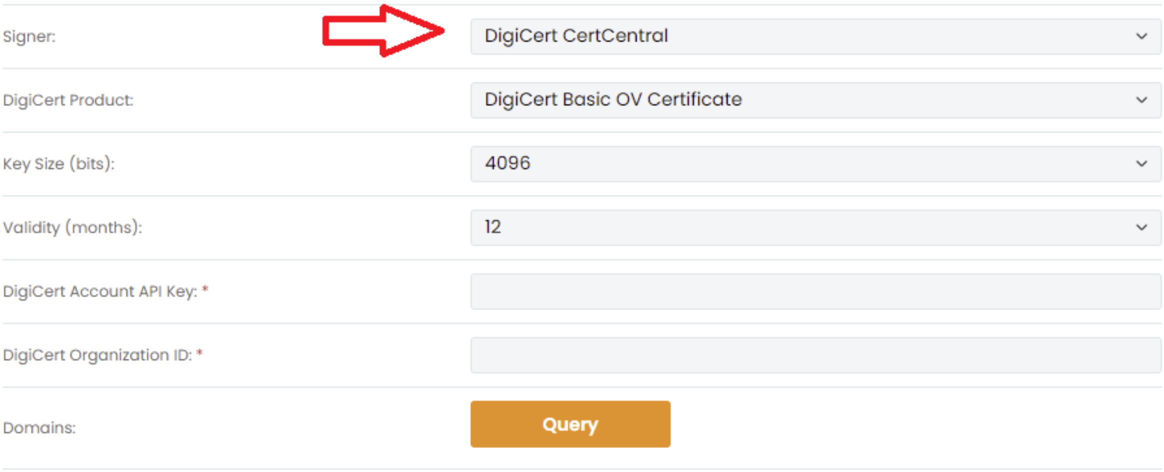
To split KeyTalk used private CA from your primary KeyTalk instance, you can connect to a secondary KeyTalk server using SCEP. KeyTalk is working on a future REST API connector to a split Keytalk CA

12.10.2 DigiCert: DigiCert, GeoTrust, QuoVadis, RapidSSL, Thawte

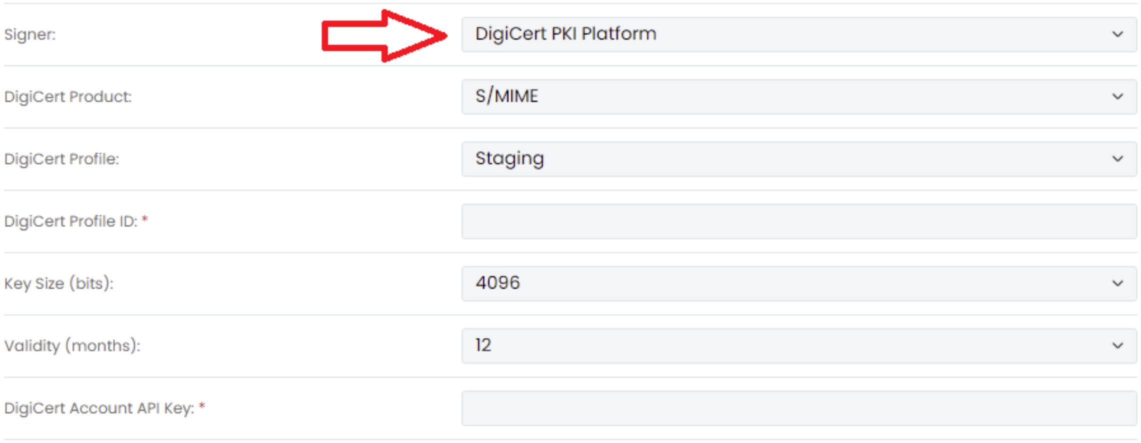
KeyTalk’s CKMS supports direct certificate issuance from the DigiCert CertCentral, and Enterprise PKI platform, supporting all their SSL brands.

Note that at the time of writing, the QuoVadis platform is a separate platform and goes through separate vetting processes compared to DigiCert’s CertCentral platform.

KeyTalk CKMS can manage and issue certificates from the DigiCert platforms, but cannot do the vetting.
Kindly ensure you’re vetting is done first for your Company/companies and your domainname(s)



Signer:	DigiCert CertCentral
DigiCert Product:	DigiCert Basic OV Certificate
Key Size (bits):	4096
Validity (months):	12
DigiCert Account API Key: *	
DigiCert Organization ID: *	
Domains:	Query



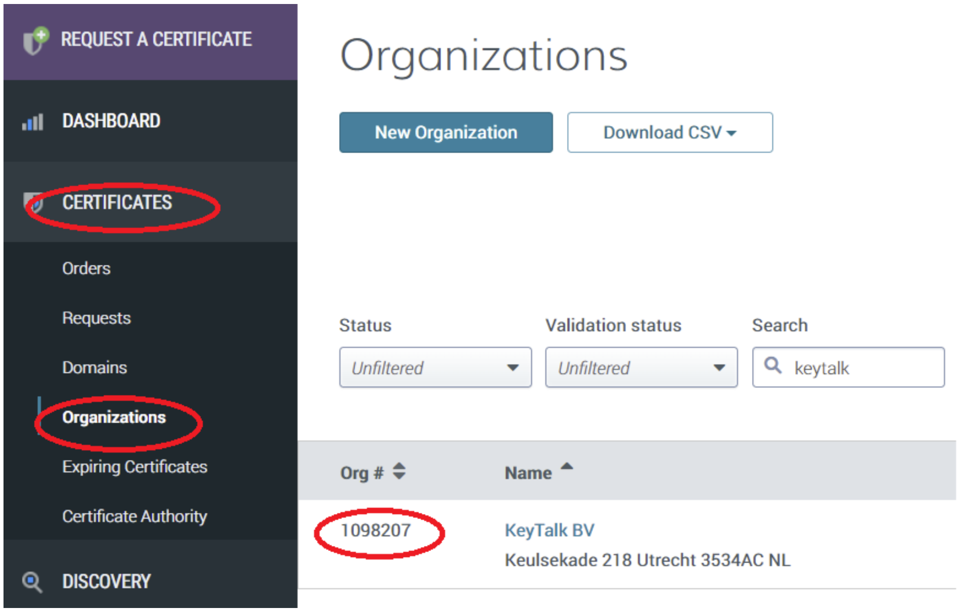
Signer:	DigiCert PKI Platform
DigiCert Product:	S/MIME
DigiCert Profile:	Staging
DigiCert Profile ID: *	
Key Size (bits):	4096
Validity (months):	12
DigiCert Account API Key: *	

Using DigiCert CertCentral, requires a API Key (Administrator or Manager), and an Organization ID. Select QUERY to check if the provided details are correct.

If there QUERY fails to fetch at least 1 domainname, then either the domain vetting did not take place yet, or no domains exist on the account.

The next page describes where in CertCentral to obtain the required connector credentials:

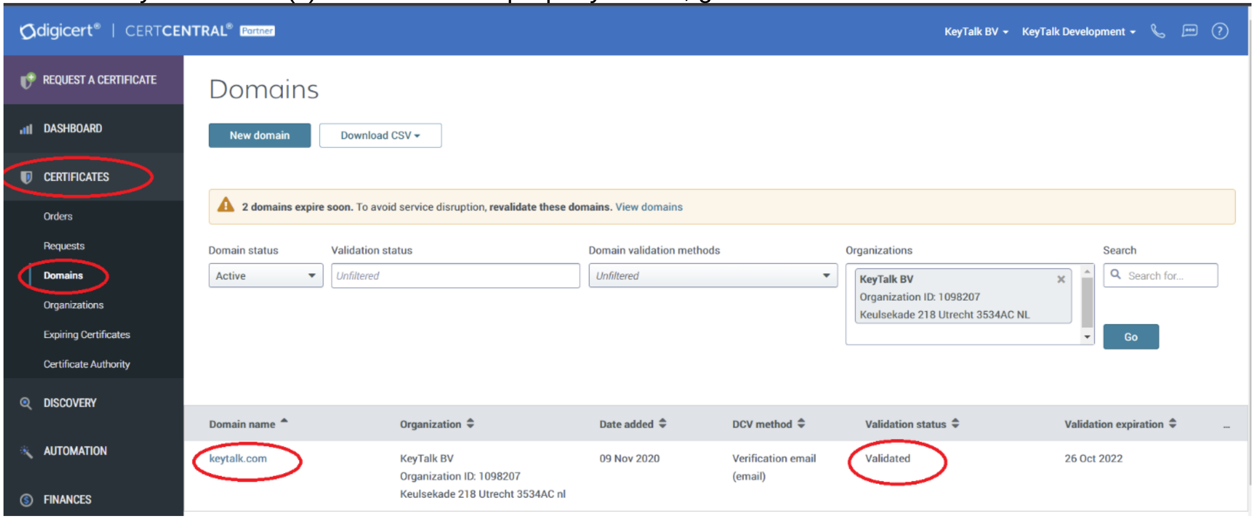
To obtain you OrganizationID used on CertCentral, login into:
https://www.digicert.com/account/login.php



Status: Unfiltered
Validation status: Unfiltered
Search: keytalk

Org #	Name
1098207	KeyTalk BV Keulsekade 218 Utrecht 3534AC NL

To check if your domain(s) has/have been properly vetted, go to:



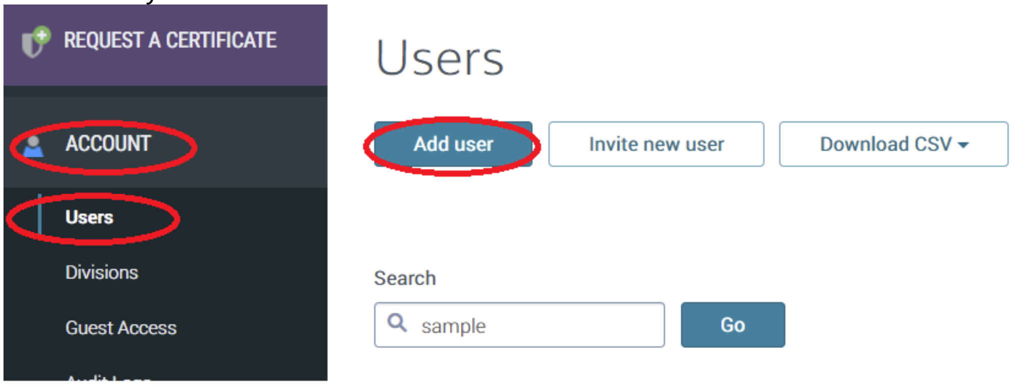
2 domains expire soon. To avoid service disruption, **revalidate these domains.** View domains

Domain status: Active
Validation status: Unfiltered
Domain validation methods: Unfiltered
Organizations: KeyTalk BV, Organization ID: 1098207, Keulsekade 218 Utrecht 3534AC NL
Search: Search for...

Domain name	Organization	Date added	DCV method	Validation status	Validation expiration
keytalk.com	KeyTalk BV Organization ID: 1098207 Keulsekade 218 Utrecht 3534AC nl	09 Nov 2020	Verification email (email)	Validated	26 Oct 2022

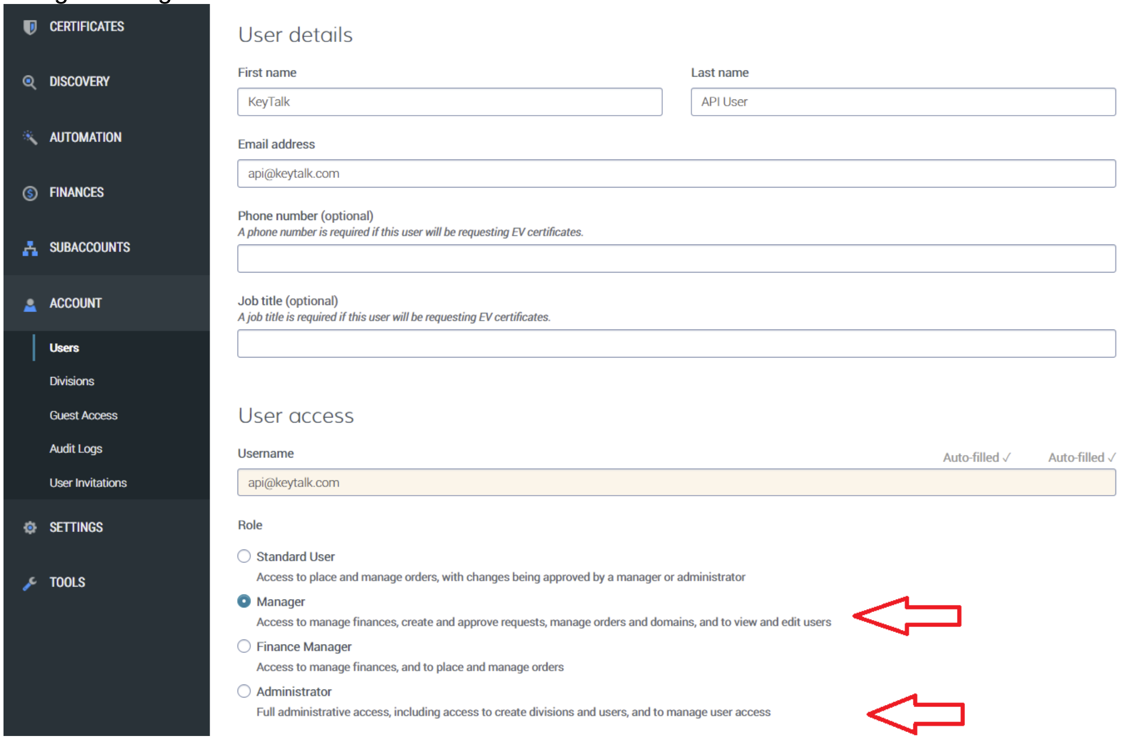
To obtain your API key, follow these steps:

1) Create a KeyTalk API User:



Search: sample

2) Assign Manager or Administrator Role



User details

First name: KeyTalk

Last name: API User

Email address: api@keytalk.com

Phone number (optional)
A phone number is required if this user will be requesting EV certificates.

Job title (optional)
A job title is required if this user will be requesting EV certificates.

User access

Username: api@keytalk.com

Role

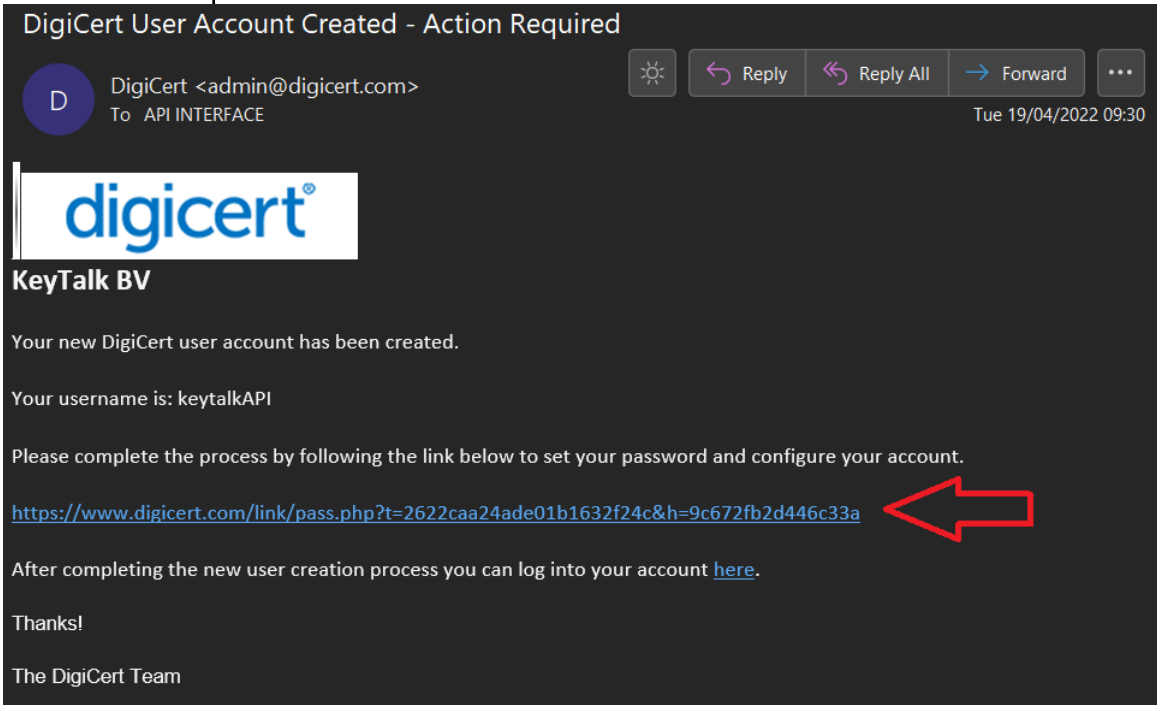
○ Standard User
Access to place and manage orders, with changes being approved by a manager or administrator

● Manager
Access to manage finances, create and approve requests, manage orders and domains, and to view and edit users

○ Finance Manager
Access to manage finances, and to place and manage orders

○ Administrator
Full administrative access, including access to create divisions and users, and to manage user access

3) Validate the user per received email:



DigiCert User Account Created - Action Required

DigiCert <admin@digicert.com>
To API INTERFACE
Tue 19/04/2022 09:30

digicert

KeyTalk BV

Your new DigiCert user account has been created.

Your username is: keytalkAPI

Please complete the process by following the link below to set your password and configure your account.

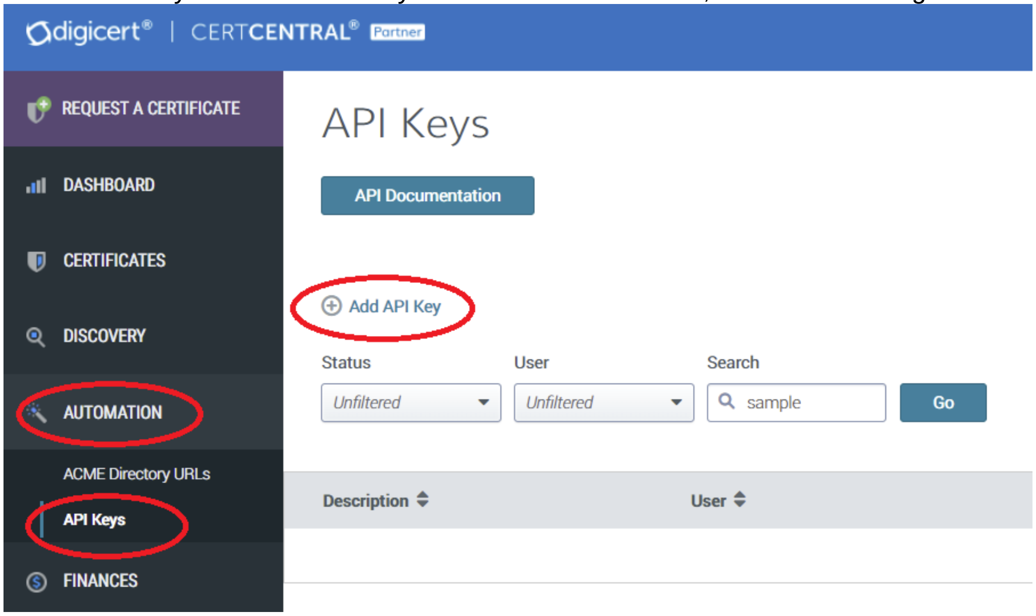
https://www.digicert.com/link/pass.php?t=2622caa24ade01b1632f24c&h=9c672fb2d446c33a

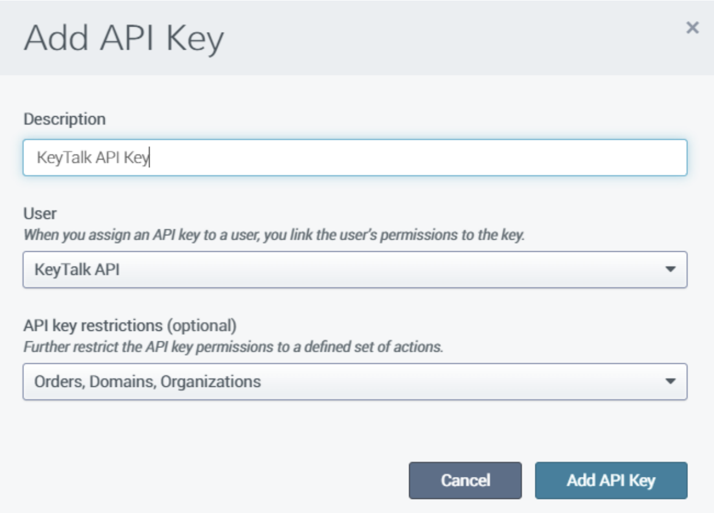
After completing the new user creation process you can log into your account here.

Thanks!

The DigiCert Team

4) Add an API key to the created KeyTalk API User for Orders, Domains and Organizations




Add API Key

Description

KeyTalk API Key

User

When you assign an API key to a user, you link the user's permissions to the key.

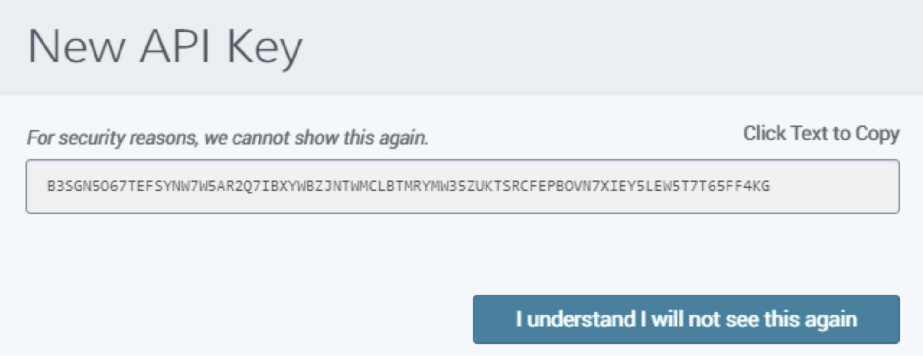
KeyTalk API

API key restrictions (optional)

Further restrict the API key permissions to a defined set of actions.

Orders, Domains, Organizations

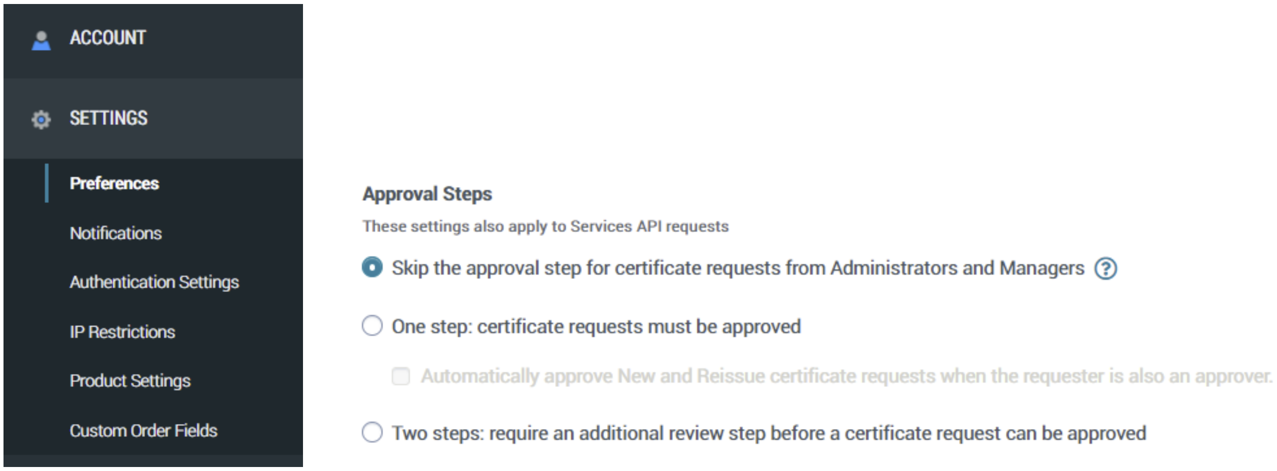
5) Copy the created API key:


New API Key

For security reasons, we cannot show this again.

```
B3SGN5O67TEFSYNW7W5AR2Q7IBXYWBZJNTWMCLBTMRYMW35ZUKTSRCFEPBOVN7XIEY5LEW5T7T65FF4KG
```

In order to automate the DigiCert CertCentral processing in combination with KeyTalk CKMS, set the following in Settings -> Advanced Settings -> Approval Steps -> Skip approval step



Approval Steps

These settings also apply to Services API requests

● Skip the approval step for certificate requests from Administrators and Managers ?

○ One step: certificate requests must be approved

□ Automatically approve New and Reissue certificate requests when the requester is also an approver.

○ Two steps: require an additional review step before a certificate request can be approved

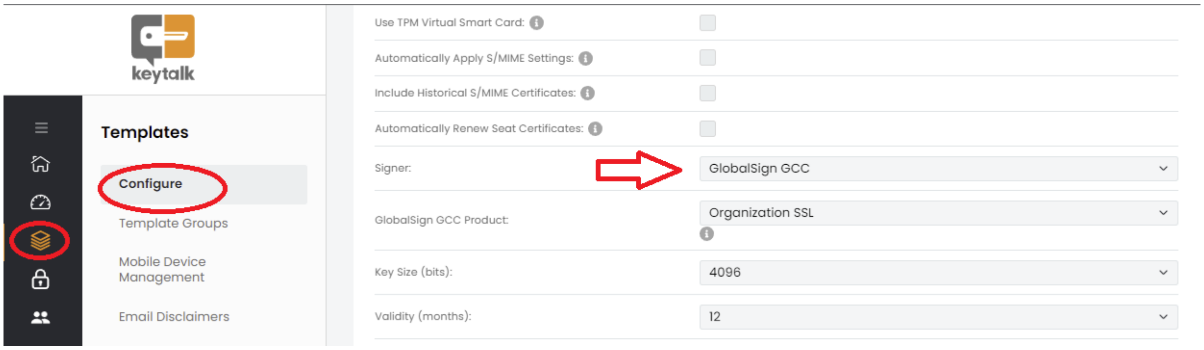
Without this setting each automated certificate request coming from KeyTalk will need additional approval (KeyTalk CKMS currently does not support waiting for the CertCentral Manager manual approval)

12.10.3 EJBCA

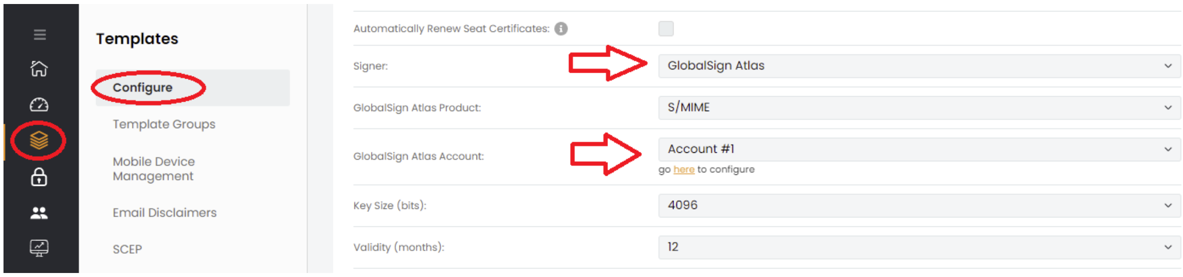
KeyTalk CKMS provides EJBCA support, however its currently in BETA. Kindly contact us, sales@keytalk.com to get access to our BETA features.

12.10.4 *GlobalSign: GCC and ATLAS*

KeyTalk CKMS interfaces natively with both GlobalSign's GCC and GlobalSign's High Volume ATLAS platform.



Use TPM Virtual Smart Card: □
Automatically Apply S/MIME Settings: □
Include Historical S/MIME Certificates: □
Automatically Renew Seat Certificates: □
Signer: GlobalSign GCC
GlobalSign GCC Product: Organization SSL
Key Size (bits): 4096
Validity (months): 12



Automatically Renew Seat Certificates: □
Signer: GlobalSign Atlas
GlobalSign Atlas Product: S/MIME
GlobalSign Atlas Account: Account #1
go here to configure
Key Size (bits): 4096
Validity (months): 12

Account Organization vetting always needs to be done by GlobalSign's vetting team and cannot be done through KeyTalk CKMS.

For GlobalSign's GCC platform, the domain vetting needs to be done through the GCC platform.

However domain vetting for GlobalSign's High Volume ATLAS platform can be done through KeyTalk, after which the KeyTalk CKMS will also warn when domain vetting is about to expire.

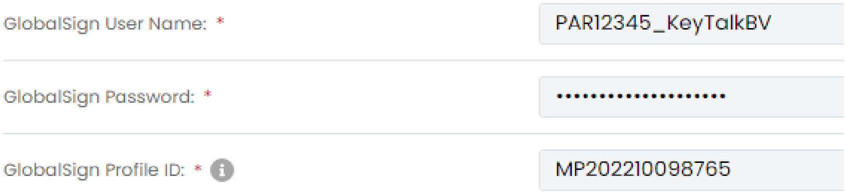
Ensure the selected validity period matches with your GlobalSign license restrictions

GlobalSign GCC

To configure GlobalSign GCC, depending on whether you are configuring server certificates or client certificates, various details are required:

For server SSL certificates:

KeyTalk TEMPLATE requires:

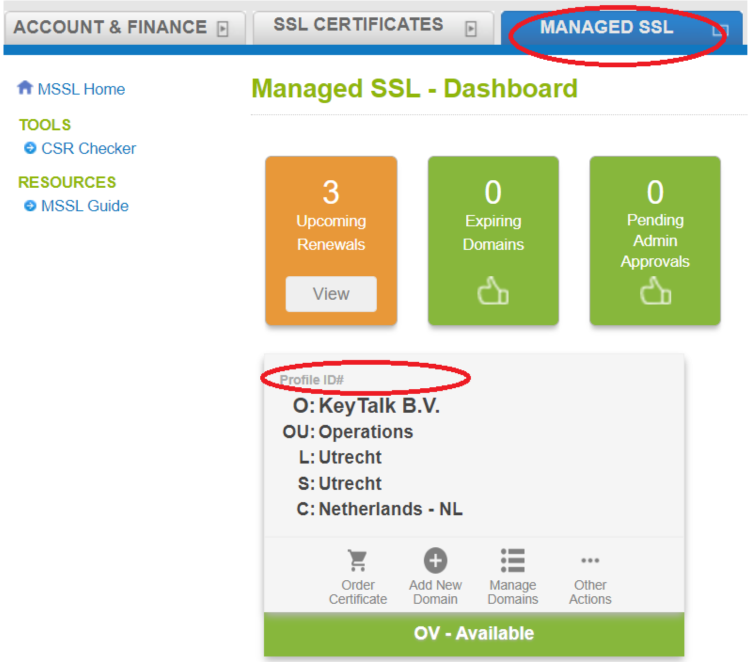


GlobalSign User Name: * PAR12345_KeyTalkBV

GlobalSign Password: * ••••••••••••••••••••

GlobalSign Profile ID: * MP202210098765

Login into your GCC account and go to Managed SSL to obtain your Profile ID:



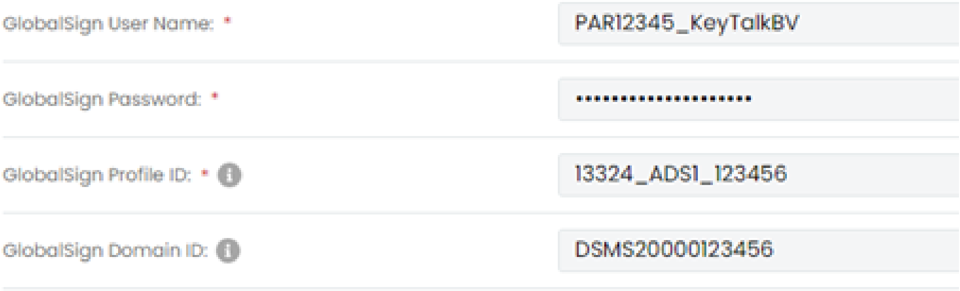
Profile ID#

O: KeyTalk B.V.
OU: Operations
L: Utrecht
S: Utrecht
C: Netherlands - NL

Further more, you must whitelist your KeyTalk CKMS public IP addresses.
To whitelist them, contact GlobalSign support or your account manager to whitelist your IP addresses.

For client certificates (ie S/MIME)

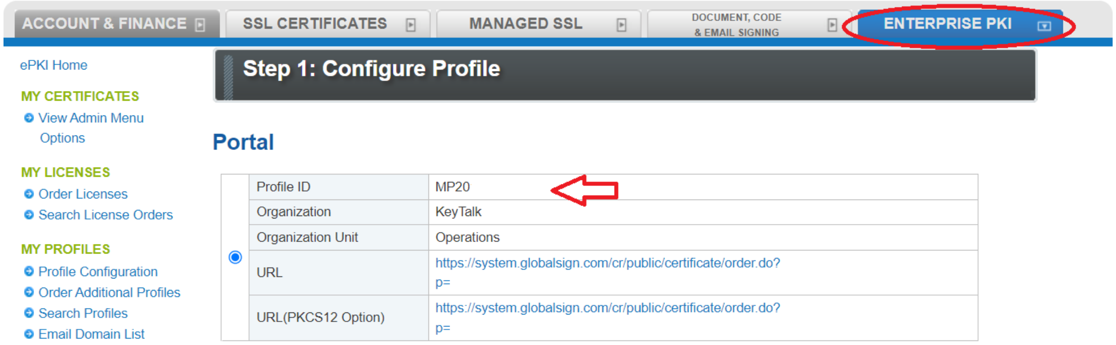
KeyTalk TEMPLATE requires the following fields:



GlobalSign User Name: *	PAR12345_KeyTalkBV
GlobalSign Password: *	••••••••••••••••••••
GlobalSign Profile ID: * ⓘ	13324_ADS1_123456
GlobalSign Domain ID: ⓘ	DSMS20000123456

The GlobalSign Domain ID is not required and will be auto fetched by KeyTalk CKMS. Filling in the static binds the template to solely allow the particular domain.

To obtain your profile ID login into GCC and go to Enterprise PKI -> Profiles:

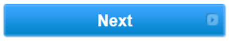


Step 1: Configure Profile

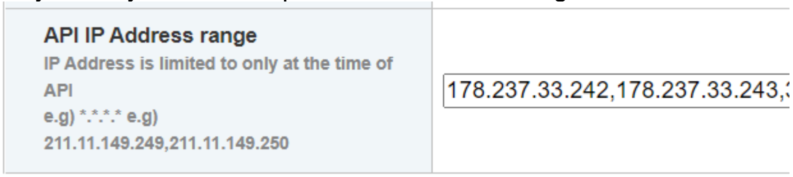
Portal

●	Profile ID	MP20
	Organization	KeyTalk
	Organization Unit	Operations
	URL	https://system.globalsign.com/cr/public/certificate/order.do?p=
	URL(PKCS12 Option)	https://system.globalsign.com/cr/public/certificate/order.do?p=

Ensure you EDIT the profile for additional information, by selecting:

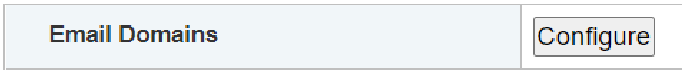


Add your KeyTalk CKMS public IP in the following field, use comma’s to include additional IPs:



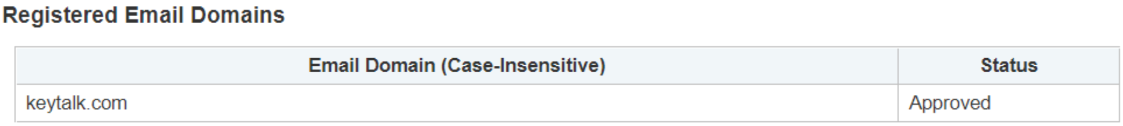
API IP Address range IP Address is limited to only at the time of API e.g) *.*.*.* e.g) 211.11.149.249,211.11.149.250	178.237.33.242,178.237.33.243,:

In the same profile configure your email domains:



Email Domains	Configure

Ensure all your used email domains are listed and have their vetting approved (!!)

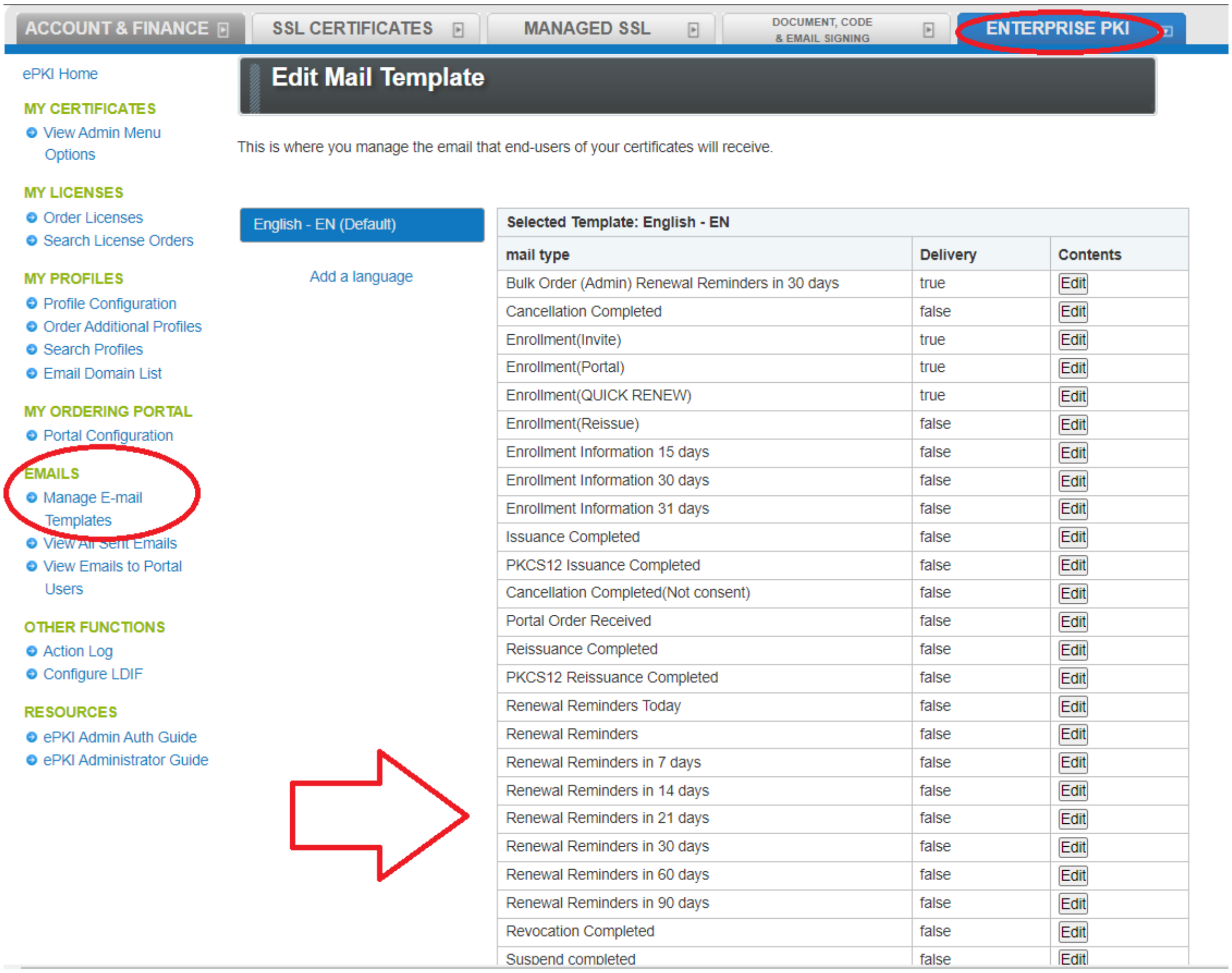


Registered Email Domains

Email Domain (Case-Insensitive)	Status
keytalk.com	Approved

By default the GlobalSign GCC platform will automatically inform your S/MIME users about upcoming renewals, including a direct URL to renew the certificate using the GCC portal.

To prevent the GCC platform from sending renewal reminders directly to the end-users, you can disable the reminders in GCC:



Edit Mail Template

This is where you manage the email that end-users of your certificates will receive.

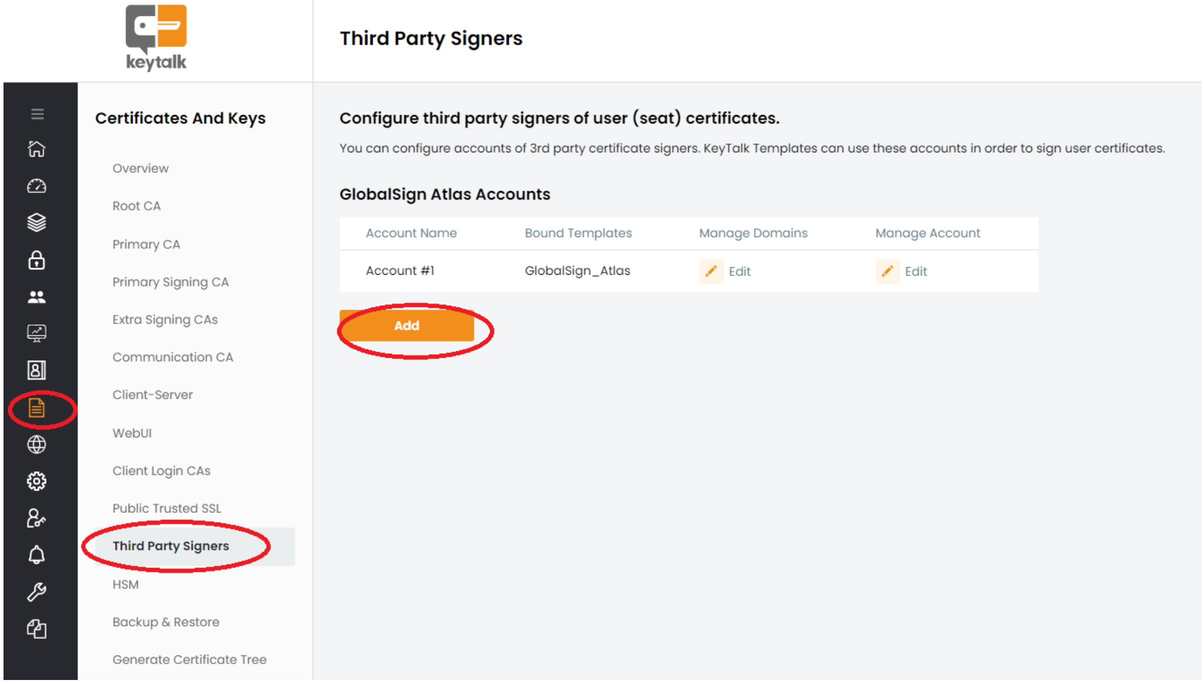
Selected Template: English - EN

mail type	Delivery	Contents
Bulk Order (Admin) Renewal Reminders in 30 days	true	Edit
Cancellation Completed	false	Edit
Enrollment(Invite)	true	Edit
Enrollment(Portal)	true	Edit
Enrollment(QUICK RENEW)	true	Edit
Enrollment(Reissue)	false	Edit
Enrollment Information 15 days	false	Edit
Enrollment Information 30 days	false	Edit
Enrollment Information 31 days	false	Edit
Issuance Completed	false	Edit
PKCS12 Issuance Completed	false	Edit
Cancellation Completed(Not consent)	false	Edit
Portal Order Received	false	Edit
Reissuance Completed	false	Edit
PKCS12 Reissuance Completed	false	Edit
Renewal Reminders Today	false	Edit
Renewal Reminders	false	Edit
Renewal Reminders in 7 days	false	Edit
Renewal Reminders in 14 days	false	Edit
Renewal Reminders in 21 days	false	Edit
Renewal Reminders in 30 days	false	Edit
Renewal Reminders in 60 days	false	Edit
Renewal Reminders in 90 days	false	Edit
Revocation Completed	false	Edit
Suspend completed	false	Edit

GlobalSign ATLAS

When dealing with high volume certificate issuance, or when you have a need for two or more email addresses in your S/MIME certificate, then GlobalSign ATLAS is typically used for GlobalSign customers.

In KeyTalk the ATLAS accounts are configured under:



Configure third party signers of user (seat) certificates.

You can configure accounts of 3rd party certificate signers. KeyTalk Templates can use these accounts in order to sign user certificates.

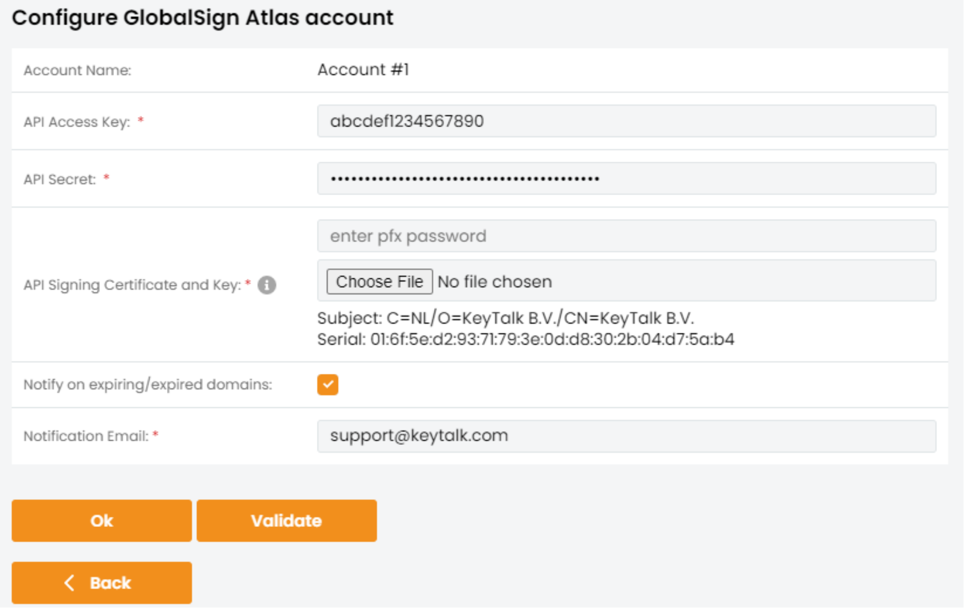
GlobalSign Atlas Accounts

Account Name	Bound Templates	Manage Domains	Manage Account
Account #1	GlobalSign_Atlas	Edit	Edit

Since GlobalSign offers its ATLAS platform without a UI, you must make use of your GlobalSign Account manager and the GlobalSign vetting team to setup your initial account, such as creating the access certificate.
Should help be required with the details requested by GlobalSign, feel free to reach out to KeyTalk support.

KeyTalk CKMS enables the vetting process of your ATLAS account domains. KeyTalk CKMS will also warn you (provided SMTP has been configured) when domain vetting is about to expire.

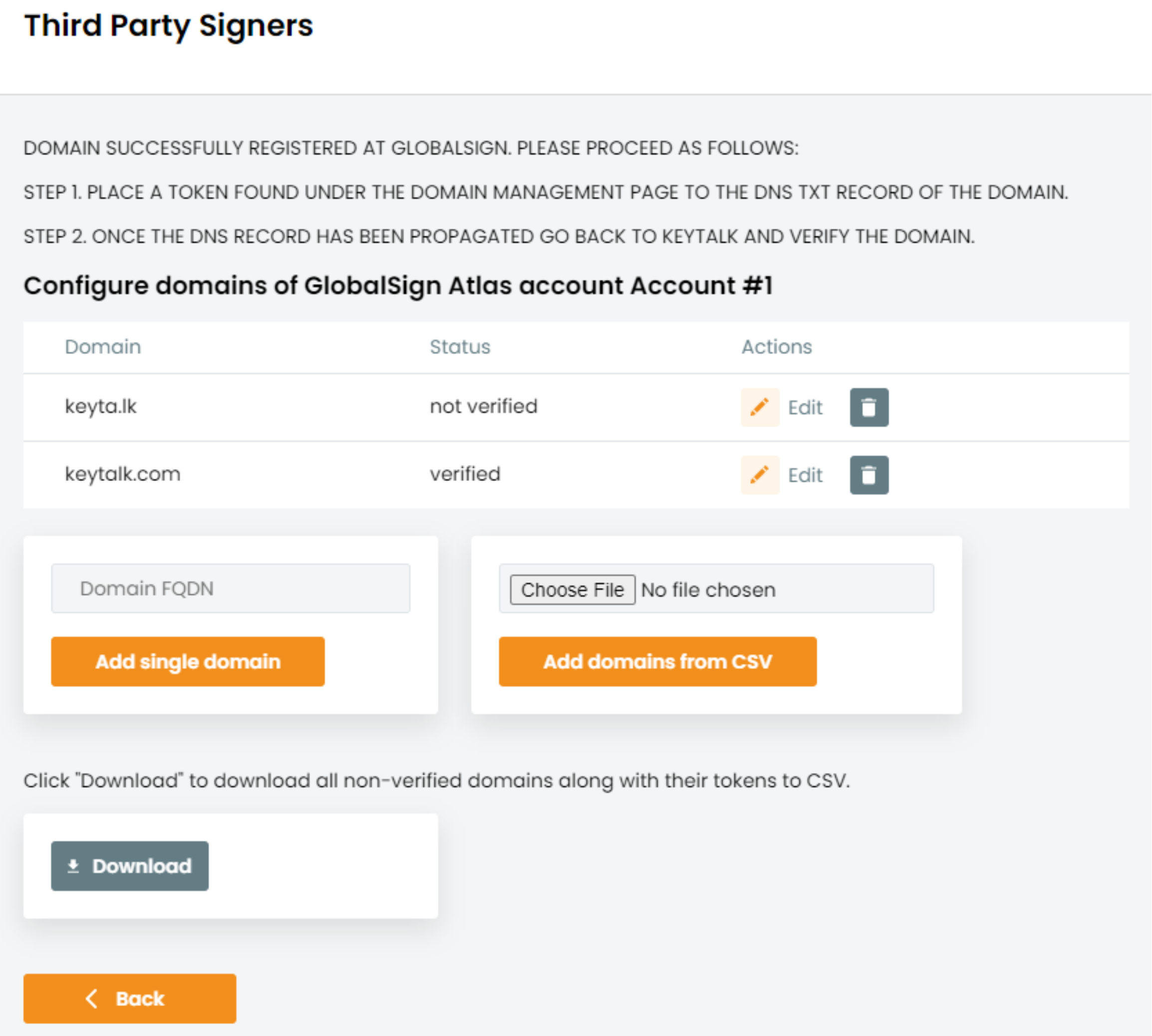
First provide the KeyTalk CKMS with your ATLAS details as provided by GlobalSign:



Configure GlobalSign Atlas account

Account Name:	Account #1
API Access Key: *	abcdef1234567890
API Secret: *	••
API Signing Certificate and Key: *	enter pfx password Choose File No file chosen Subject: C=NL/O=KeyTalk B.V./CN=KeyTalk B.V. Serial: 01:6f:5e:d2:93:71:79:3e:0d:d8:30:2b:04:d7:5a:b4
Notify on expiring/expired domains:	☑
Notification Email: *	support@keytalk.com

After entering your ATLAS account details, you can (mass) configure and start the vetting for your Domains:



Third Party Signers

DOMAIN SUCCESSFULLY REGISTERED AT GLOBALSIGN. PLEASE PROCEED AS FOLLOWS:

STEP 1. PLACE A TOKEN FOUND UNDER THE DOMAIN MANAGEMENT PAGE TO THE DNS TXT RECORD OF THE DOMAIN.

STEP 2. ONCE THE DNS RECORD HAS BEEN PROPAGATED GO BACK TO KEYTALK AND VERIFY THE DOMAIN.

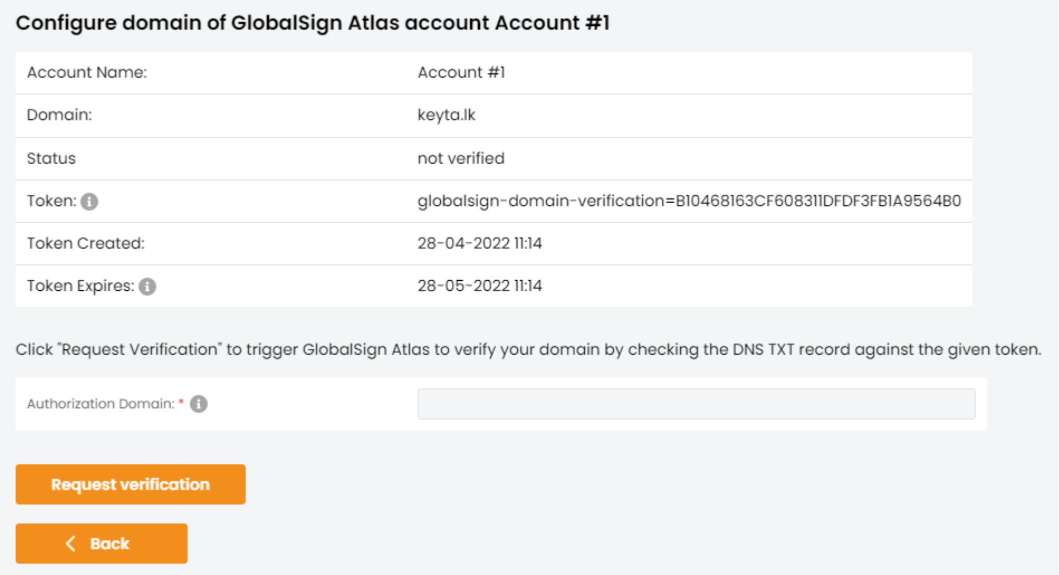
Configure domains of GlobalSign Atlas account Account #1

Domain	Status	Actions
keyta.lk	not verified	Edit
keytalk.com	verified	Edit

Click "Download" to download all non-verified domains along with their tokens to CSV.

To start the vetting, simply download the ATLAS domain verification token and enter them into your DNS.

One your DNS has been updated, go back to your KeyTalk CKMS and validate the domain:



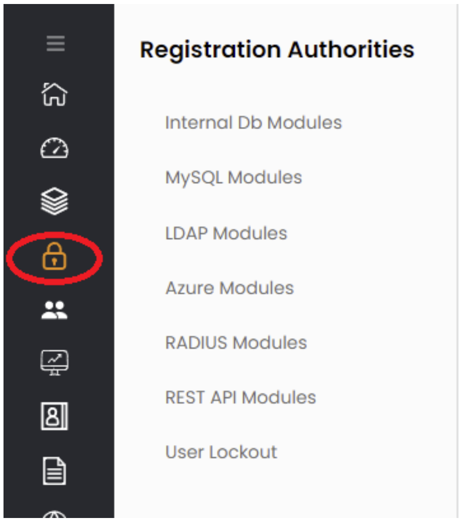
Configure domain of GlobalSign Atlas account Account #1

Account Name:	Account #1
Domain:	keyta.lk
Status	not verified
Token:	globalsign-domain-verification=B10468163CF608311DFDF3FB1A9564B0
Token Created:	28-04-2022 11:14
Token Expires:	28-05-2022 11:14

Click "Request Verification" to trigger GlobalSign Atlas to verify your domain by checking the DNS TXT record against the given token.

Authorization Domain: *

13. Registration Authority connectors: Authentication Modules



Registration Authorities

Internal Db Modules

MySQL Modules

LDAP Modules

Azure Modules

RADIUS Modules

REST API Modules

User Lockout

In order to verify that a certificate is allowed to be issued to an end-point, a Registration Authority (RA) is required to be configured.

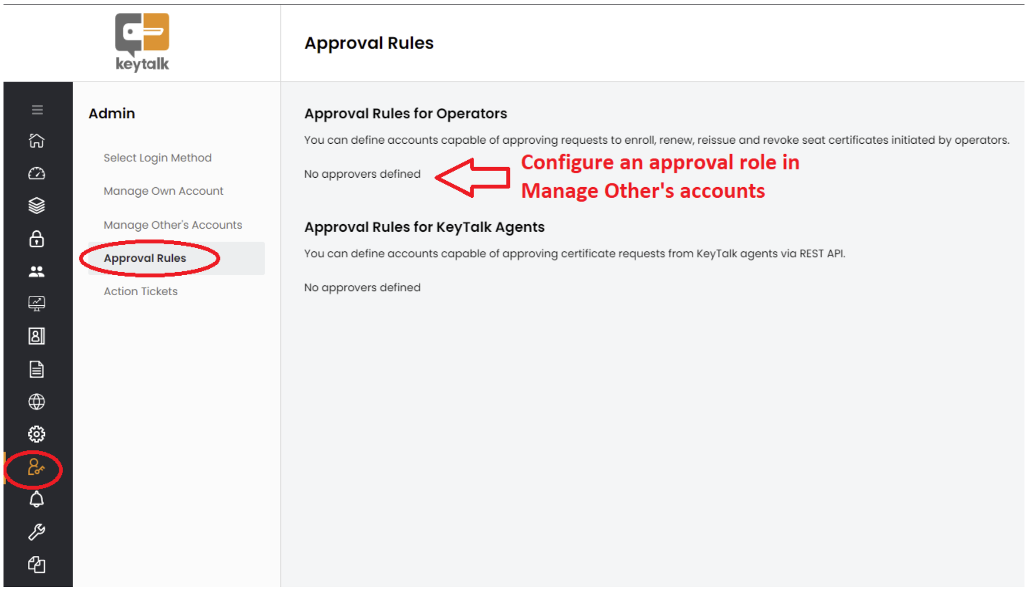
KeyTalk automates the RA process by binding the certificate issuance to your existing Identity Provider solution(s). Commonly these are an (Azure) Active Directory or LDAP within a company, or a Radius based One Time Password solutions, such as provided by RSA, Gemalto, or Vasco/OneSpan.

IoT Service providers and other leading software Service providers often register and manage their devices and/or user-accounts in a MySQL (or related) Database.

To support our current customer-base, KeyTalk uses non-proprietary, ie native, connectors, whereby we follow the philosophy that a device or user making use of an IDP is not necessarily part of the trusted domain community.

Should you require a connector currently not supported by KeyTalk, kindly get in touch with sales@keytalk.com and explain to us your business-case so that we may make available the required connector.

Eventhough the RA based authentication validation is automated, the KeyTalk CKMS allows for manual approval before issuing a certificate:



Approval Rules

Approval Rules for Operators

You can define accounts capable of approving requests to enroll, renew, reissue and revoke seat certificates initiated by operators.

No approvers defined

Configure an approval role in Manage Other's accounts

Approval Rules for KeyTalk Agents

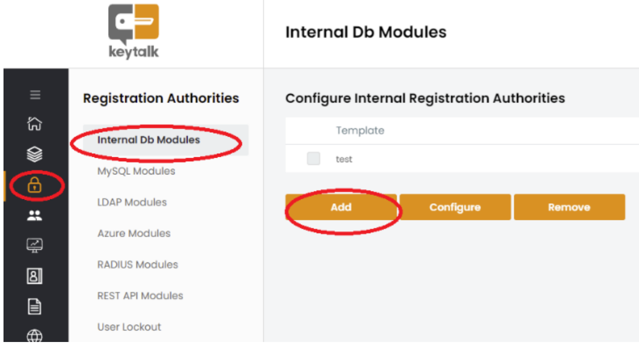
You can define accounts capable of approving certificate requests from KeyTalk agents via REST API.

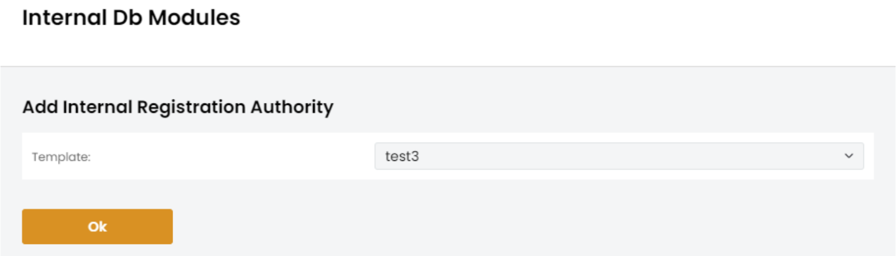
No approvers defined

13.1 Internal KeyTalk Db, MySQL based

Most KeyTalk admins start with the KeyTalk internal Db to test if their basic KeyTalk setup works without being dependent upon external connectors in order to limit point of failure trouble shooting.

- To add an internal KeyTalk Db to your KeyTalk TEMPLATE certificate template, add it under:

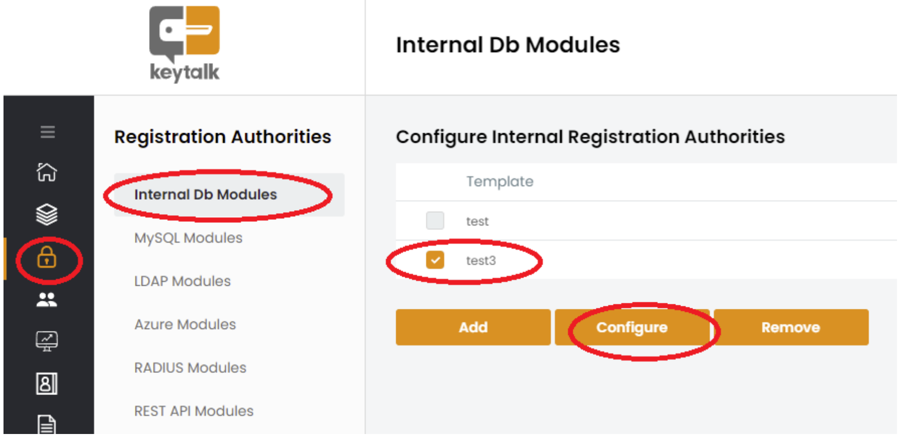


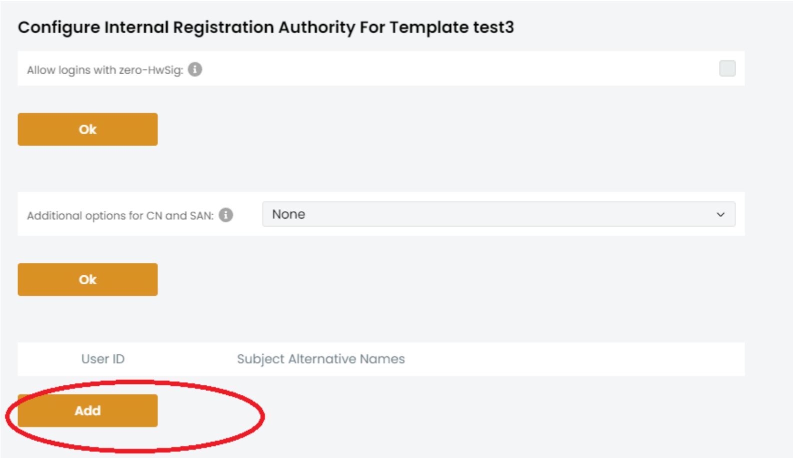

Internal Db Modules

Add Internal Registration Authority

Template: test3

- Now configure a certificate recipient (server/user/device) under:



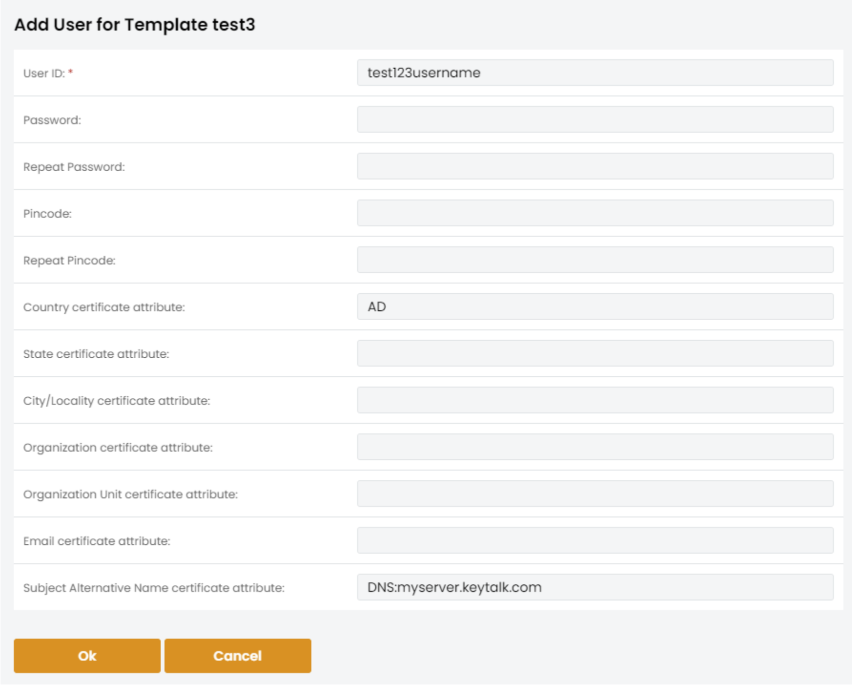

Configure Internal Registration Authority For Template test3

Allow logins with zero-HwSig: □

Additional options for CN and SAN: None

User ID	Subject Alternative Names

- Configure the new certificate recipient:

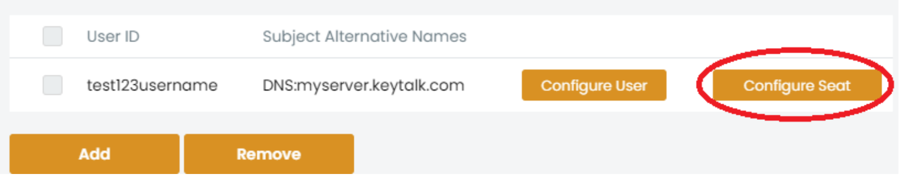


Add User for Template test3

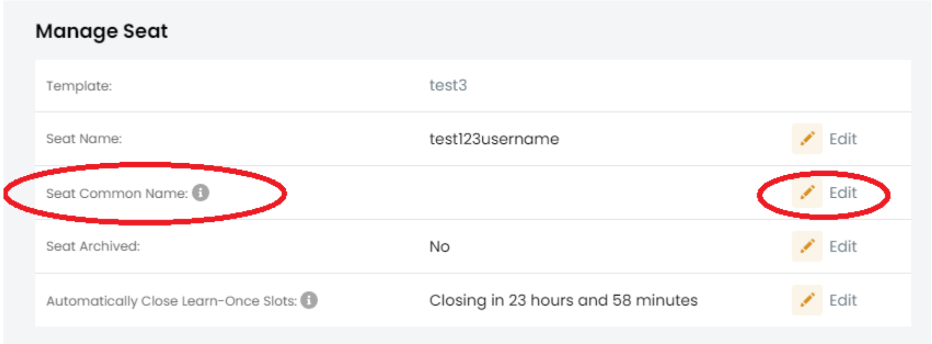
User ID: *	test123username
Password:	
Repeat Password:	
Pincode:	
Repeat Pincode:	
Country certificate attribute:	AD
State certificate attribute:	
City/Locality certificate attribute:	
Organization certificate attribute:	
Organization Unit certificate attribute:	
Email certificate attribute:	
Subject Alternative Name certificate attribute:	DNS:myserver.keytalk.com

Note 1: While configuring the Subject Alternative Name is not required for a simple test user account against the KeyTalk internal CA, it is recommended to always give the SAN value proper testing values as required for a client user or server user.
As soon as you wish to extend your test account to obtain certificates from a (Qualified) Trusted Certificate Provider, then the required data must be correct.

Note 2: the User ID will be used as CN, unless a CN is defined explicitly in the SEAT meta data of the certificate recipient.



□	User ID	Subject Alternative Names		
□	test123username	DNS:myserver.keytalk.com	Configure User	Configure Seat



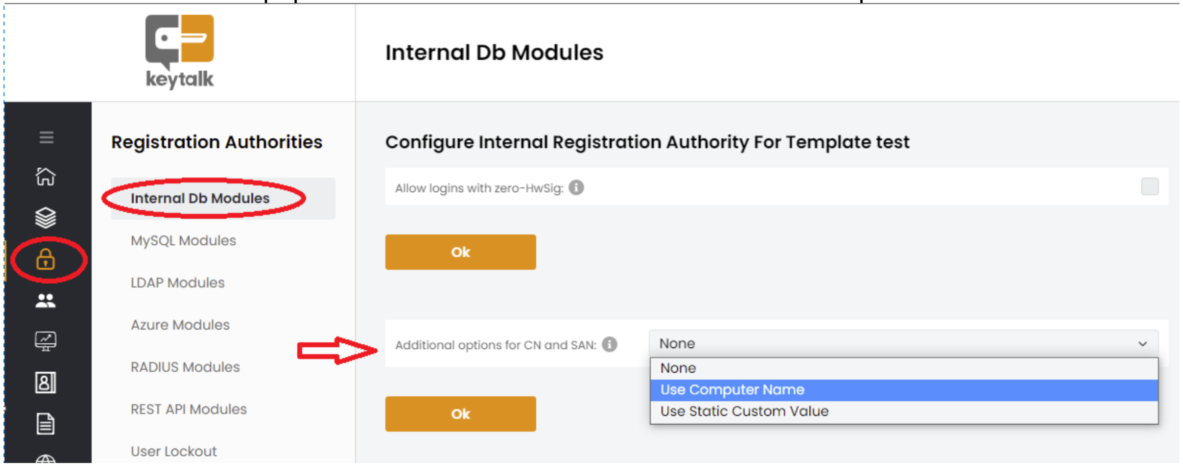
Manage Seat

Template:	test3	
Seat Name:	test123username	Edit
Seat Common Name: ⓘ		Edit
Seat Archived:	No	Edit
Automatically Close Learn-Once Slots: ⓘ	Closing in 23 hours and 58 minutes	Edit

13.1.1 Enrollment for Machine Certificates

KeyTalk’s Internal Db Authentication Module supports a mode (similar to its Active directory / LDAP counterpart) for issuing Machine Certificates to KeyTalk clients (currently Windows and Linux as of client 5.6.3)

Set the Internal Db to populate the CN and the SAN DNS with: Use Computer Name



Configure Internal Registration Authority For Template test

Allow logins with zero-HwSig: ☐

Additional options for CN and SAN: None

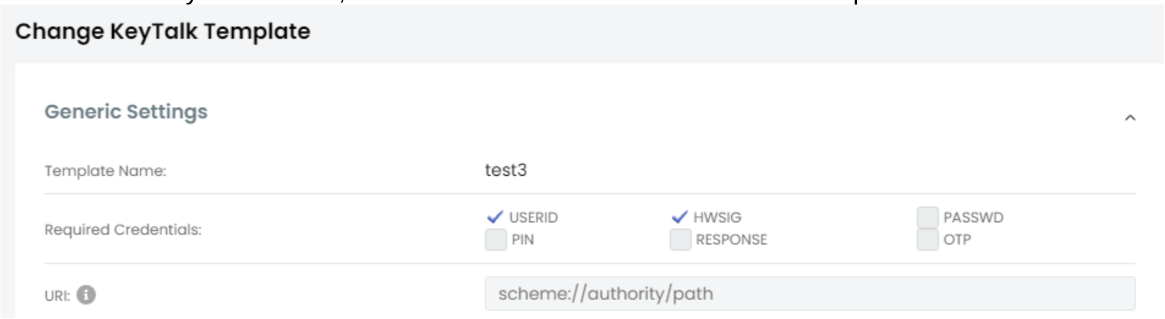
- None
- Use Computer Name
- Use Static Custom Value

This triggers the KeyTalk client to read the local machinename and have KeyTalk server apply it to the CN and SAN instead of the used Kerberos or regular username.

(Static) Custom Value replaces the CN with a static value often used for similar devices such as iPhone or Android whereby only 1 or 2 entries are preferred in a RADIUS.

To support easy enrollment of machine certificates for a handful or large amounts of end-points, you can also choose for the KeyTalk Internal Db Registration Authority to work in anonymous mode. This results in ANY device requesting a machine certificate, to be issued one without authentication. This anonymous Machine Certificate issuance mode is advised only for specific use-case cases whereby either a Proof of Concept is run, or whereby only trusted machines are in the network capable of reaching the KeyTalk server.

To enable anonymous mode, uncheck in the TEMPLATE the need for a password:



Change KeyTalk Template

Generic Settings

Template Name: test3

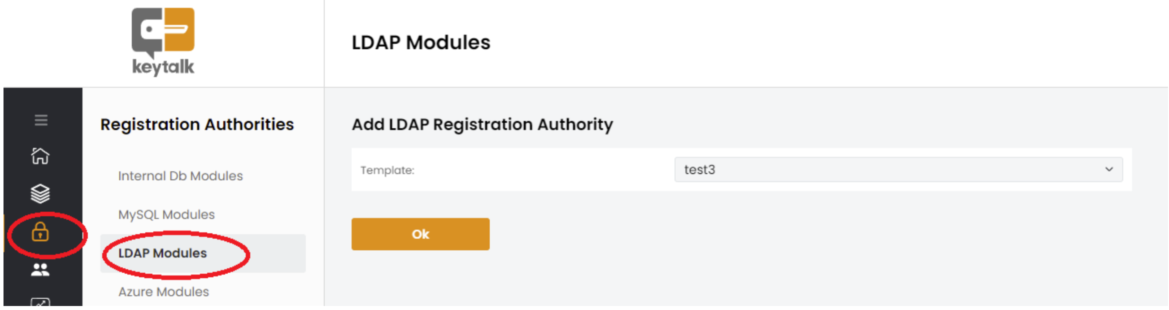
Required Credentials: ✓ USERID ✓ HWSIG ☐ PASSWD ☐ PIN ☐ RESPONSE ☐ OTP

URI: scheme://authority/path

13.2 Active Directory / LDAP / Kerberos

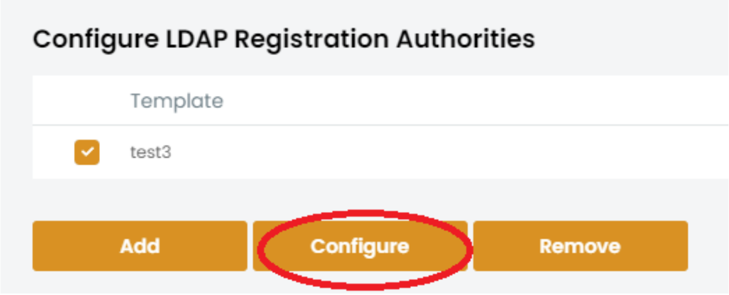
By far the most commonly used connector is Active Directory. With AD being very similar to an LDAP, and the connector working for both LDAP and AD, the Registration Authority connector is referred to as LDAP.

First make sure you have a KeyTalk TEMPLATE, ie the needed basic certificate template created and ADD it to a Registration Authority LDAP module in KeyTalk:



Template: test3

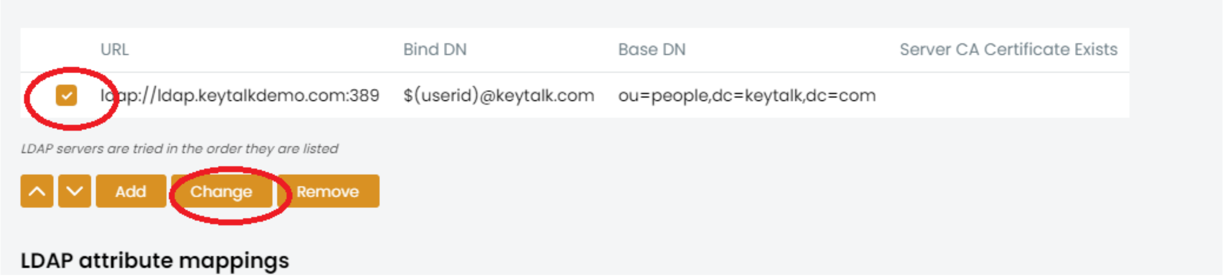
Start by configuring the initial default LDAP of the chosen KeyTalk TEMPLATE:



Configure LDAP Registration Authorities

	Template
☑	test3

And change the initial LDAP server settings:

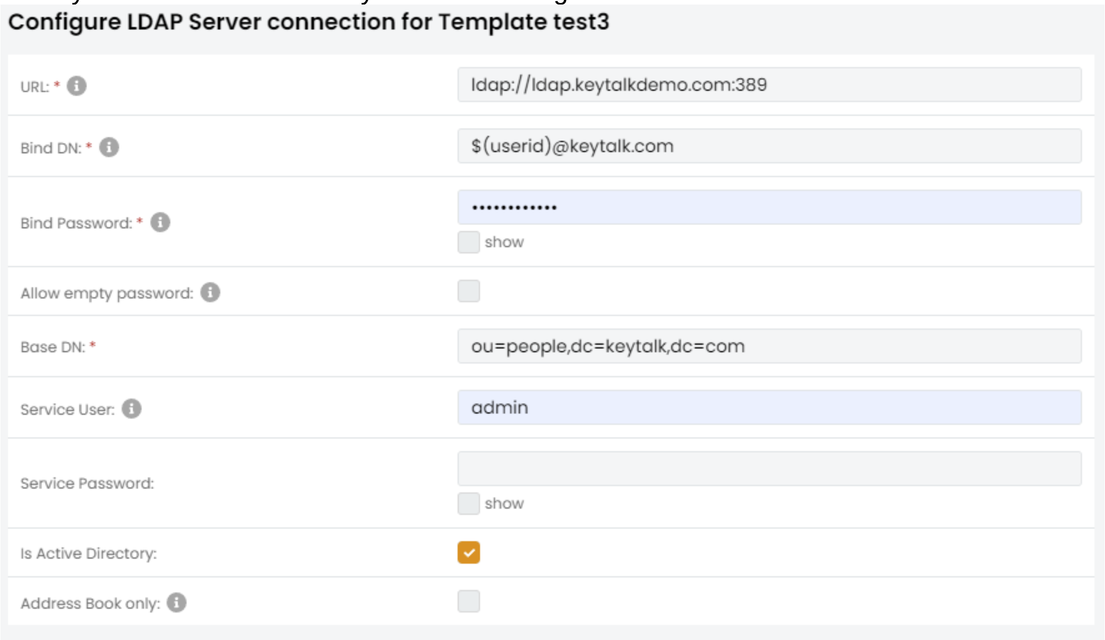


	URL	Bind DN	Base DN	Server CA Certificate Exists
☑	ldap://ldap.keytalkdemo.com:389	$(userid)@keytalk.com	ou=people,dc=keytalk,dc=com	

LDAP servers are tried in the order they are listed

LDAP attribute mappings

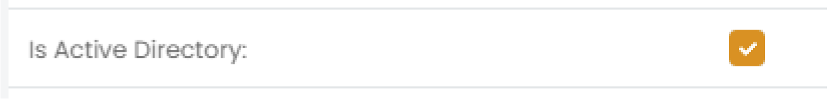
Enter your own Active Directory or LDAP settings:



Configure LDAP Server connection for Template test3

URL: *	ldap://ldap.keytalkdemo.com:389
Bind DN: *	$(userid)@keytalk.com
Bind Password: *	•••••••••••• □ show
Allow empty password:	□
Base DN: *	ou=people,dc=keytalk,dc=com
Service User:	admin
Service Password:	 □ show
Is Active Directory:	☑
Address Book only:	□

When configuring an AD ensure the “Is Active Directory is checkmarked:



Is Active Directory: ☑

It is highly recommended to take small steps in configuring your AD/LDAP connector, in order to easily troubleshoot potential issues

13.2.1 Initial configuration steps:

a) Ensure your firewall rules, and when needed custom network routing, allows the KeyTalk server to communicate with your AD/LDAP

b) Start without TLS, ie use port 389 and ldap://<fqdn_or_ip> This ensures that a potential failure to connect isn’t related to the use of LDAPS

c) Ensure you have a test account on your target AD or LDAP that is active, and which doesn’t use your actual password (using port 389 will send the password unencrypted over your network)

d) Use as Bind DN: $(userid)@<mydomainname.local>

e) Use as your Base DN: ou=people,dc=mydomainname,dc=local
You could start more elaborate by using in depth OU references in order to limit the account verification lookup to a specific Organizational Unit, but it is our experience that keeping the DN initially as limited as possible reduces points of potential failure.

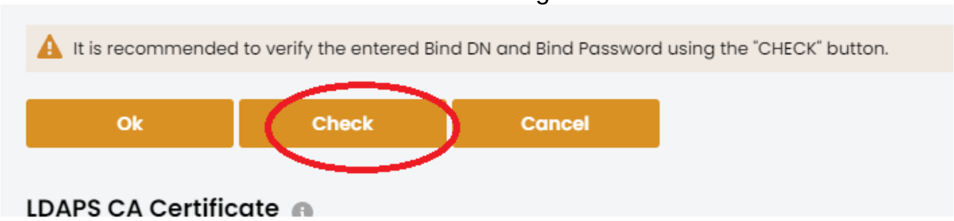
f) Leave TEMPLATE User empty

g) Leave TEMPLATE Password empty

h) Leave Address Book only empty

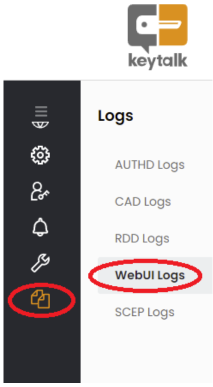
i) Leave LDAPS CA Certificate empty since we start testing over port 389 using ldap

j) Now CHECK to see if these connection settings work:



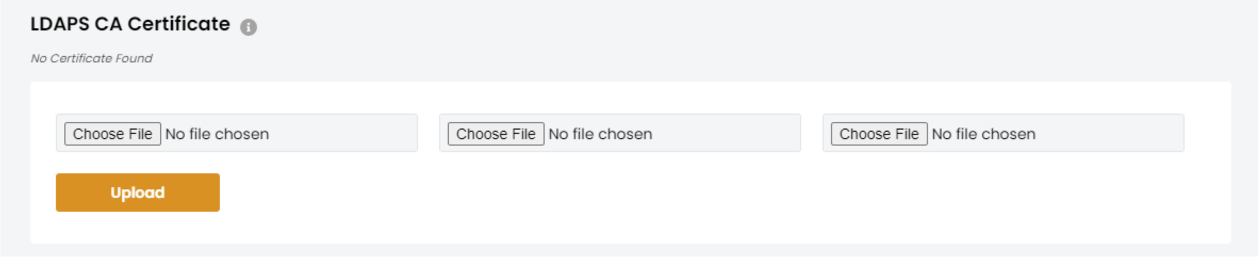
⚠ It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

k) When j) results in success: Congratulations you just confirmed your connector works, now follow the suggested advanced configuration steps.
If j) results in failure: Check your WEBUI log to see what went wrong.



13.2.2 Advanced configuration steps: LDAPS and Global Catalog

l) Now that your basic configuration is proven to be working, start by ensuring your connection is secured using LDAPS. To enable LDAPS upload the intermediate (also referred to as parent or issuing certificate) of your AD or LDAP.
This intermediate certificate is required to establish a TLS trust between KeyTalk and the target AD or LDAP. KeyTalk by default does not trust any CA, including its own, for all individual connector settings as this helps security.
To find out what your LDAP or AD certificate is, login into your LDAP or AD, and view your certificate to determine its intermediate.
Then download this intermediate from your AD or LDAP certificate store and upload it in PEM or DER format into KeyTalk :

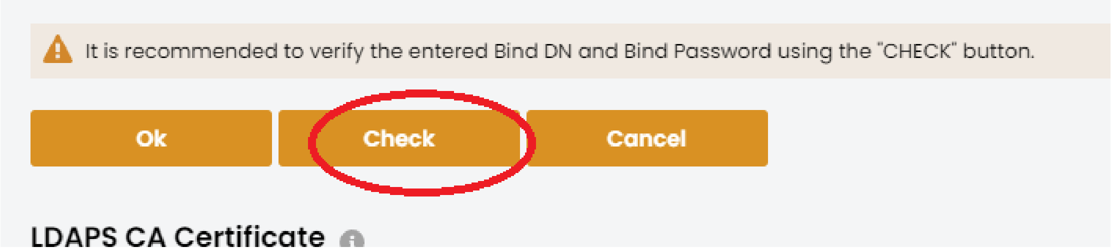

LDAPS CA Certificate

No Certificate Found

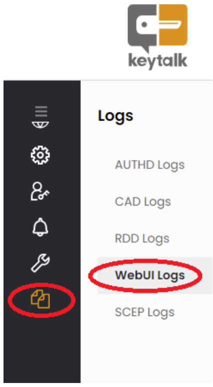
Choose File No file chosen
Choose File No file chosen
Choose File No file chosen
Upload

m) Ensure your URL refers to: ldaps://
And include port 636 instead of 389
You could also use Global Catalog ports, but do note that GC does not allow for example to write a user certificate to the AD

n) After uploading the CA trust of your LDAPS cert, CHECK to see if these connection settings still work:


It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

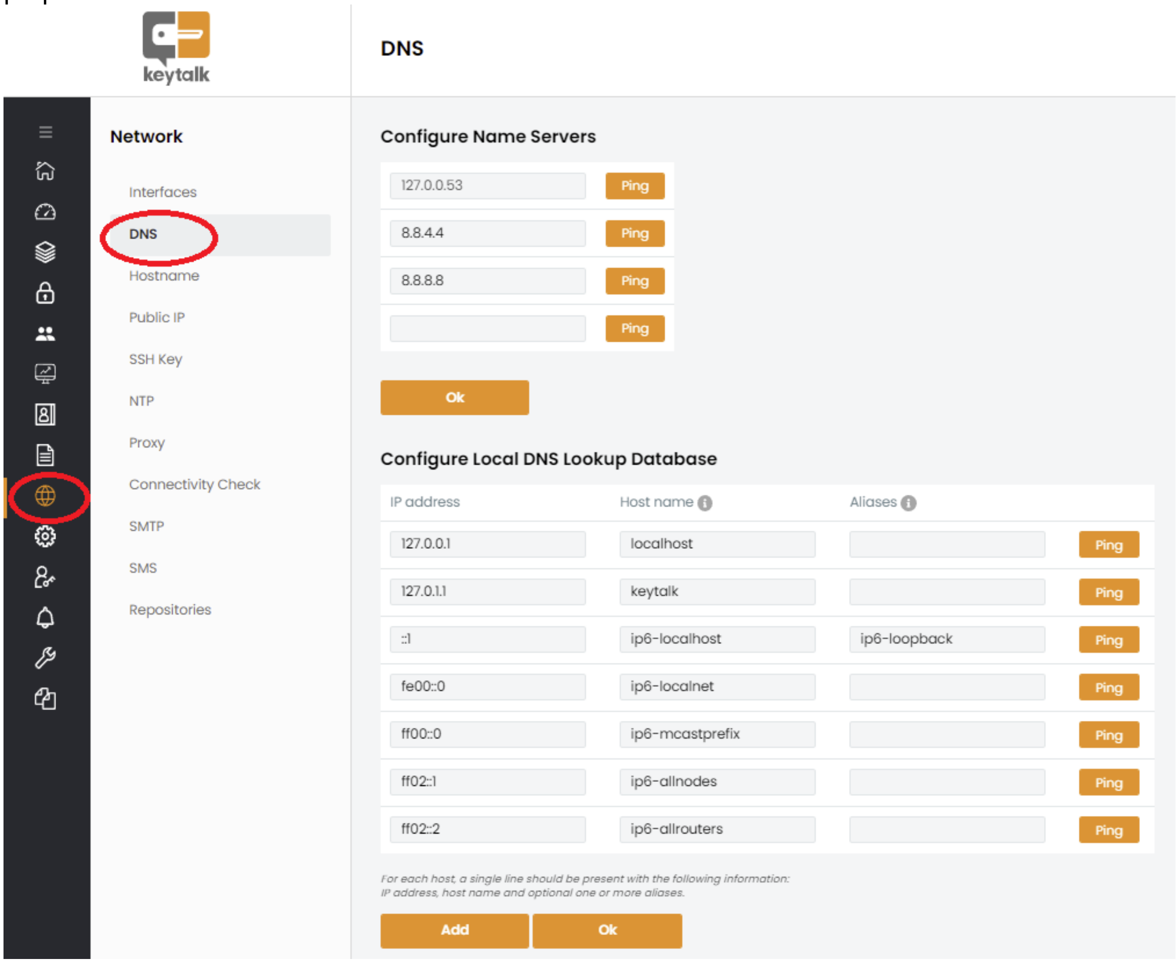
o) If n) results in success: Congratulations, you just confirmed your secure connector works
If n) results in failure: Check your WEBUI log to see what went wrong, but likely its either the trust of the connection failed, or your networking doesn't allow for LDAPS/port 636



Should the connection fail due to the LDAPS certificate trust failing, ensure that your used AD/LDAP FQDN and/or IP are actually part of the LDAPS certificate. You might be using ldaps://ldap.mycompany.**com** or ldaps://<ipv4/ipv6-address> whereby your LDAPS certificate was only created for support ldaps://ldap.mycompany.**local**.
Note that older certificates might only have the FQDN in the CN value, and not in the Subject Alternative Name, where-as TLS requires the FQDN to be part of the SAN as CN has been made obsolete as of 2017.
Also SHA1 can no longer be used for TLS connections, requiring your AD certificate to be updated with a SHA2 version.

Often failure is likely caused by a mismatch on the LDAPS Subject Alternative Name value, you can fix the LDAPS problem in 2 ways:

- Upload a new proper certificate as issued under the configured intermediate to your AD/LDAP (preferred and proper solution)
- Circumvent the problem on a network local DNS lookup level within KeyTalk. Though this will work, it is not recommended for operational environments as you would be introducing work-arounds instead of a proper fix!



Configure Name Servers

127.0.0.53	Ping
8.8.4.4	Ping
8.8.8.8	Ping
	Ping

Ok

Configure Local DNS Lookup Database

IP address	Host name	Aliases	
127.0.0.1	localhost		Ping
127.0.1.1	keytalk		Ping
::1	ip6-localhost	ip6-loopback	Ping
fe00::0	ip6-localnet		Ping
ff00::0	ip6-mcastprefix		Ping
ff02::1	ip6-allnodes		Ping
ff02::2	ip6-allrouters		Ping

For each host, a single line should be present with the following information: IP address, host name and optional one or more aliases.

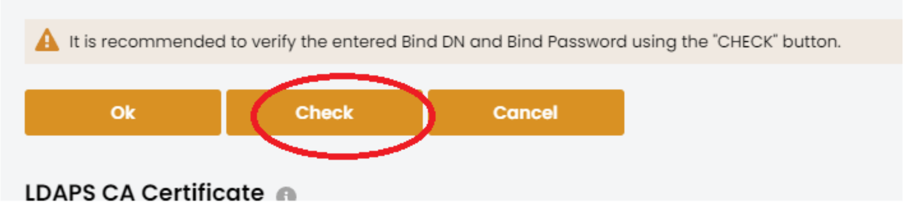
Add | Ok

p) Enterprise customers usually run elaborate AD setups. In these cases a regular bind based lookup over port 389 or 636 may be slow. In this case its best to use Global Catalog.
To enable Global Catalog:

- Check with your AD admin that Global Catalog is enabled
- Check with your firewall admin that the KeyTalk machine is allows to communicate over Global Catalog ports 3268 or 3269
- Change your ldap url to :   ldap://ldap.mycompany.com:3268 (non-secure)
  Or change it to: ldaps://ldap.mycompany.com:3269 (secure requiring LDAPS certificate)

NOTE: While GC is faster, it also has the limitation of not being able to write for example certificate data back to the AD!

q) Now CHECK to see if these connection settings work:

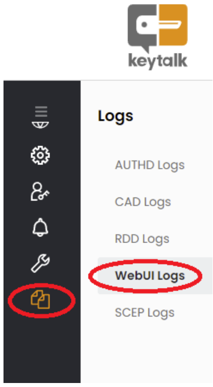


⚠ It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

Ok | Check | Cancel

LDAPS CA Certificate

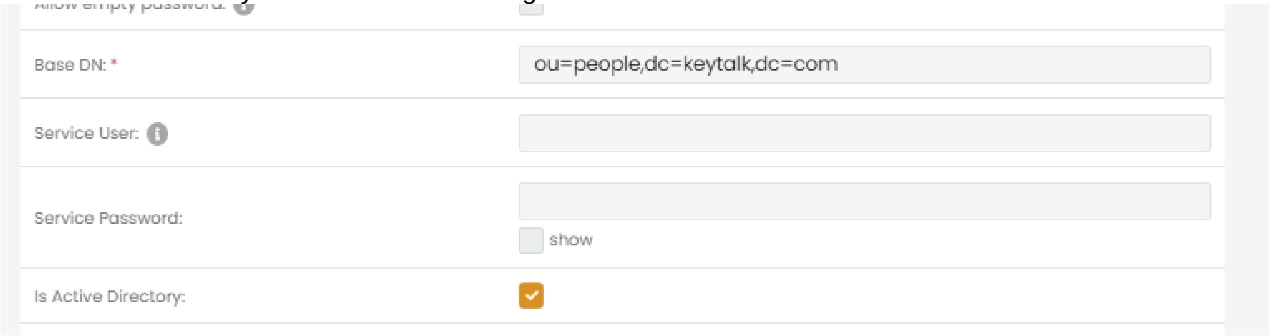
r) If n) results in success: Congratulations! You just confirmed your secure connector works
If n) results in failure: Check your WEBUI log to see what went wrong, but likely its either the trust of the connection failed or your networking doesn't allow for LDAPS/port 636



13.2.3 Advanced configuration steps: Password change support, attribute lookup and pre-enrolment

s) KeyTalk supports client-side password change 2 weeks in advance, provided that the password is about to expire in AD. KeyTalk uses the default Active Directory and LDAP features based on the existing valid account.

t) To support certificate pre-enrolment and password change on an account that had its password already expire, a TEMPLATE User with Account Operator authorization is required on your LDAP/AD.
Add these details in your LDAP server setting under:



Allow empty password:	□
Base DN: *	ou=people,dc=keytalk,dc=com
Service User:	
Service Password:	□ show
Is Active Directory:	☑

AttributeFields that KeyTalk will manipulate using the set AD TEMPLATE User (using ldap_modify_ext_s):

- ✓ userCertificate;binary (DER format)
- ✓ altSecurityIdentities (in the format: "X509:<I>%s<S>%s")
- ✓ unicodePwd

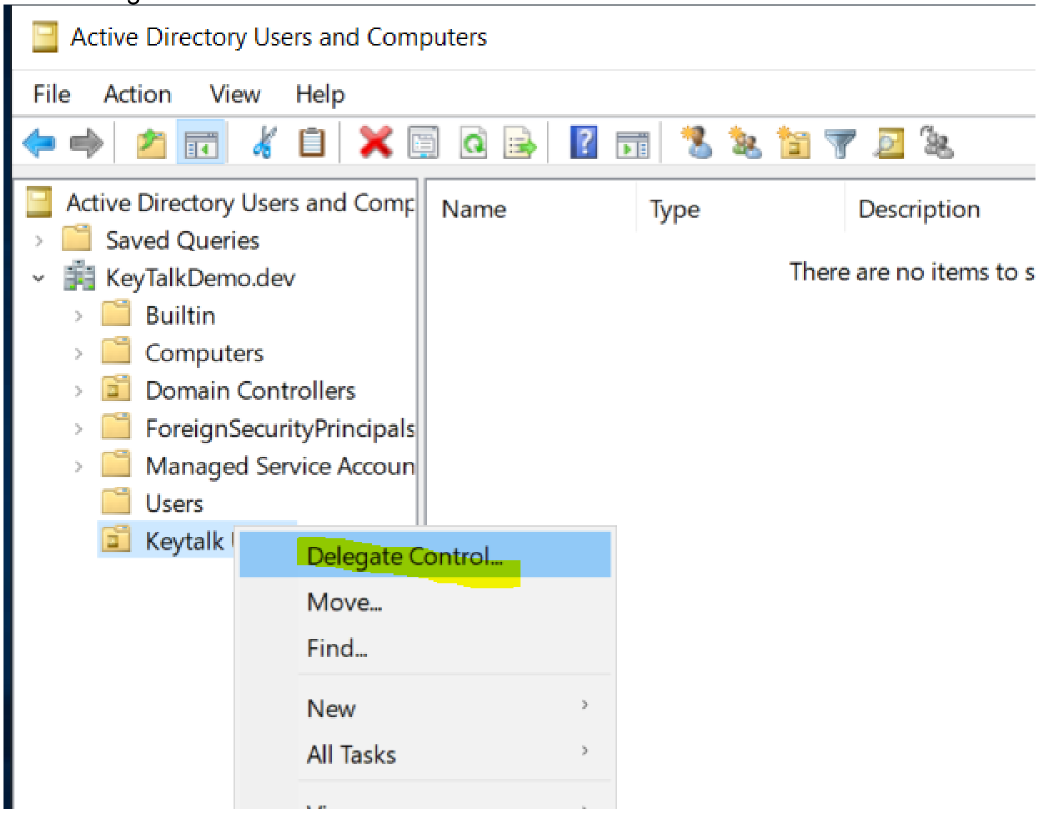
Using a Domain Admin user would be the route to guarantee access! But an AD specialist can more specifically limit the privileges of the TEMPLATE User account, whereby the generic Account Operator authorization role should suffice with rights to write/read certificates if needed. Follow below steps to set a custom authorization role in AD for the required TEMPLATE User in KeyTalk:

NOTE: The KeyTalk solution will attempt a Bind based on similar functionality as offered by LDP. Often an AD has policies in place that prevent the reading of the given Bind DN and its underlying children OUs. Ensure that these are readable.

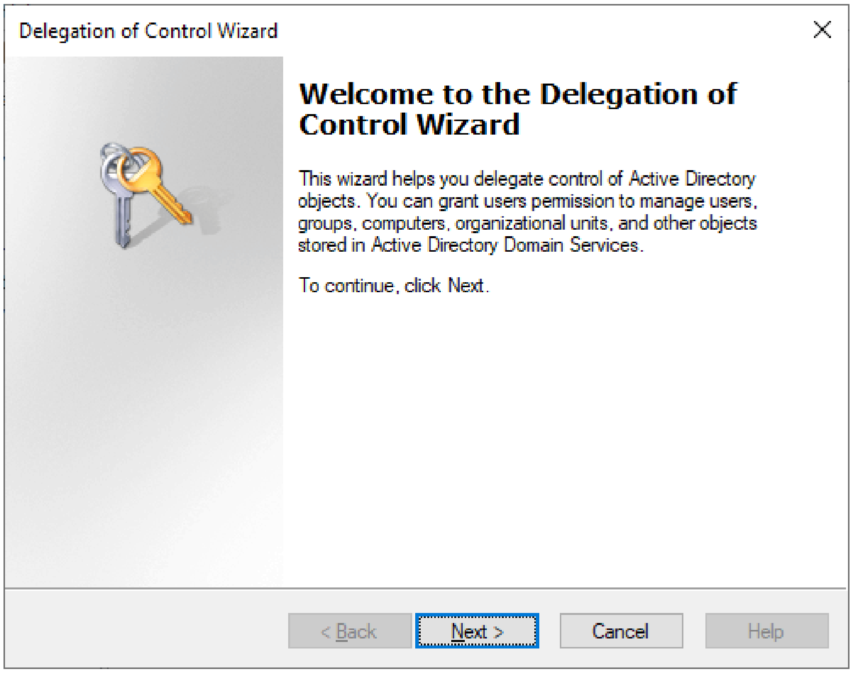
The KeyTalk TEMPLATE account needs permissions to change/reset the password, read/write the "altSecurityIdentities" and "userCertificate" attributes on User Objects.

These permission can be set as follows.

In Active Directory Users and Computers right click the OU (or entire domain) that holds the Keytalk users and click "Delegate Control"



On the Delegation Control Wizard screen click "Next"

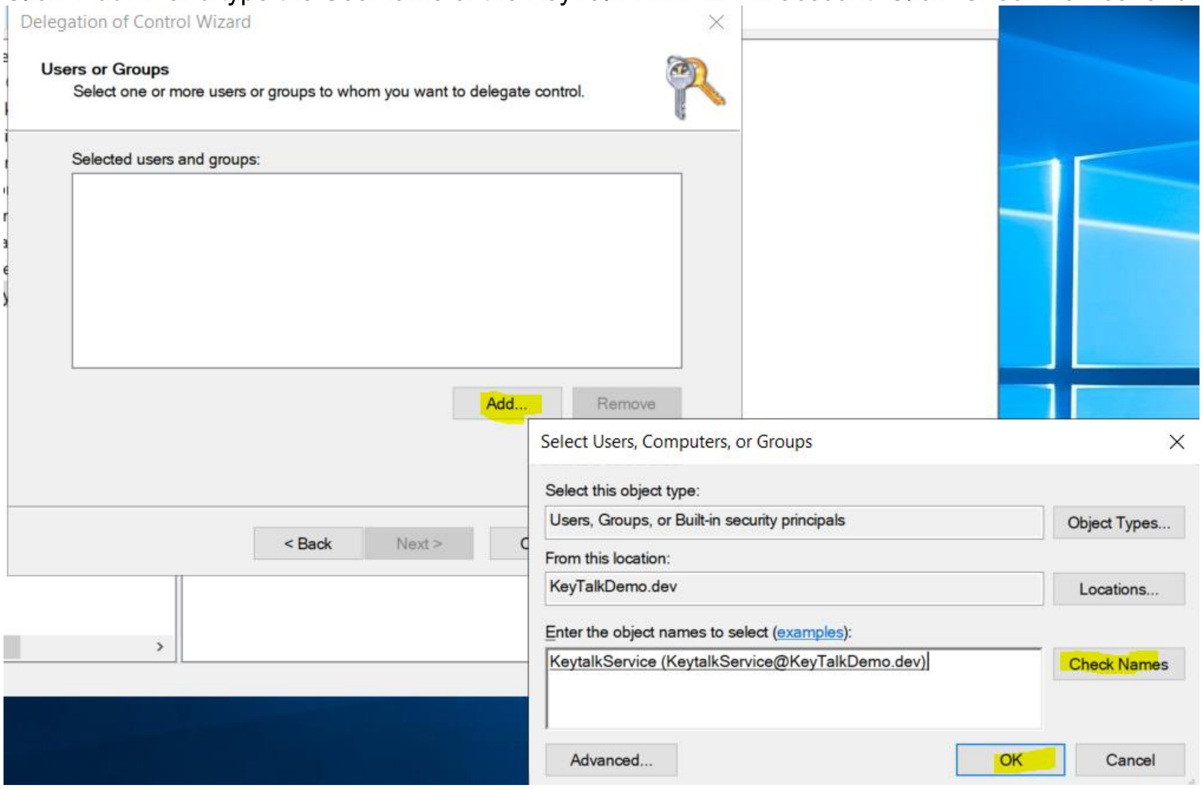

Delegation of Control Wizard

Welcome to the Delegation of Control Wizard

This wizard helps you delegate control of Active Directory objects. You can grant users permission to manage users, groups, computers, organizational units, and other objects stored in Active Directory Domain Services.

To continue, click Next.

Click “Add…” and type the Username of the KeyTalk TEMPLATE account. Click “Check Names” and “OK”



Delegation of Control Wizard

Users or Groups
Select one or more users or groups to whom you want to delegate control.

Selected users and groups:

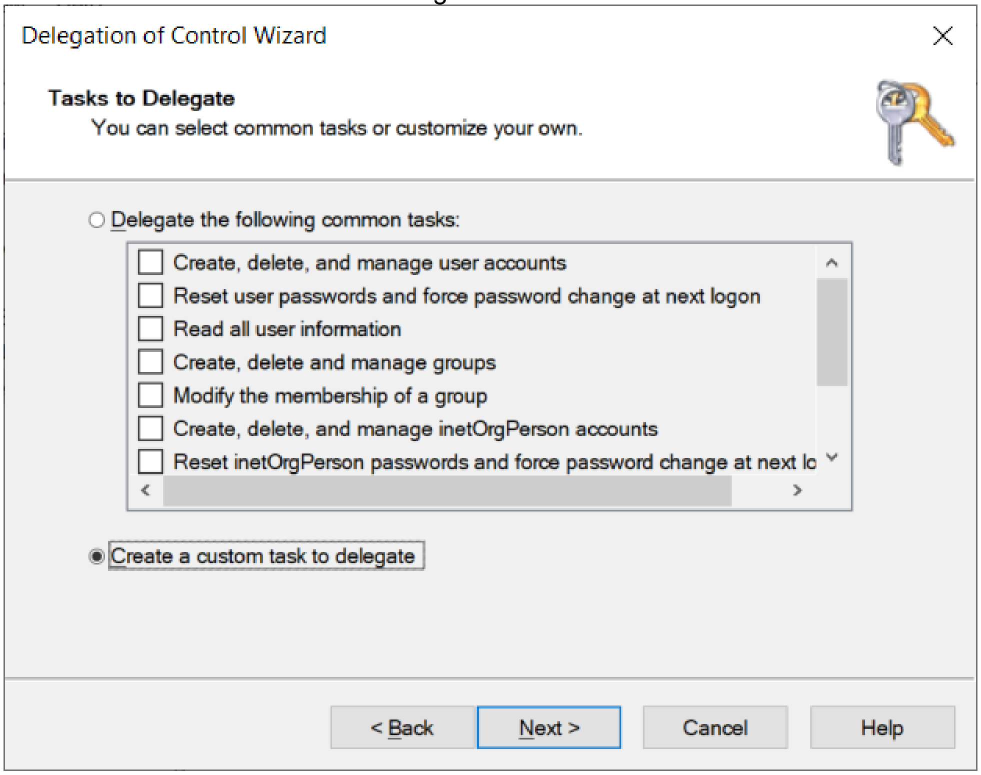
Select Users, Computers, or Groups

Select this object type: Users, Groups, or Built-in security principals

From this location: KeyTalkDemo.dev

Enter the object names to select (examples): KeytalkService (KeytalkService@KeyTalkDemo.dev)

Select “Create a custom task to delegate” and click “Next”



Delegation of Control Wizard

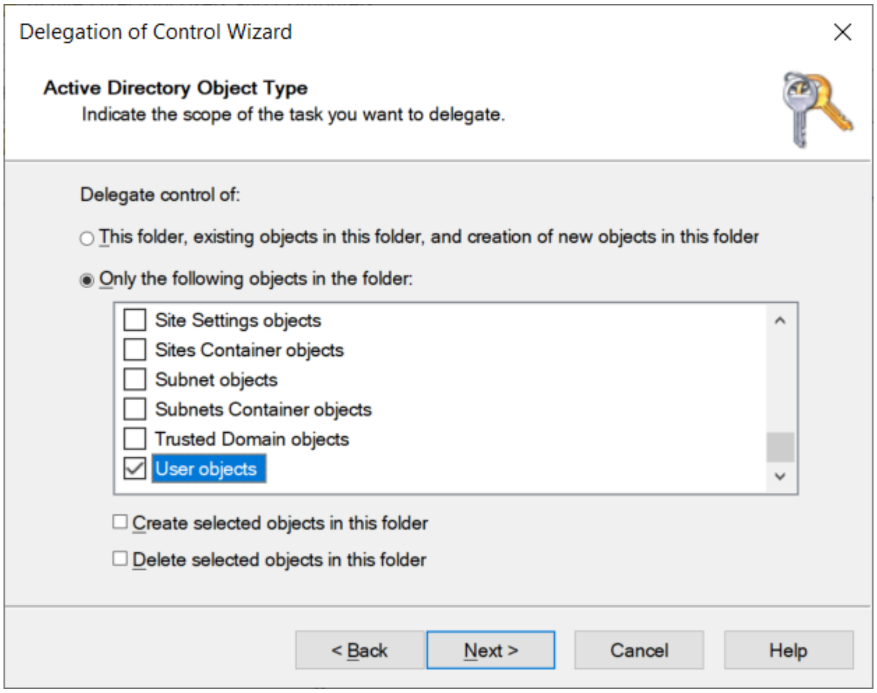
Tasks to Delegate
You can select common tasks or customize your own.

○ Delegate the following common tasks:

- □ Create, delete, and manage user accounts
- □ Reset user passwords and force password change at next logon
- □ Read all user information
- □ Create, delete and manage groups
- □ Modify the membership of a group
- □ Create, delete, and manage inetOrgPerson accounts
- □ Reset inetOrgPerson passwords and force password change at next lo

● Create a custom task to delegate

Select “Only the following objects in the folder:” Scroll down in the list and check “User Objects” and click “Next”



Delegation of Control Wizard

Active Directory Object Type

Indicate the scope of the task you want to delegate.

Delegate control of:

○ This folder, existing objects in this folder, and creation of new objects in this folder

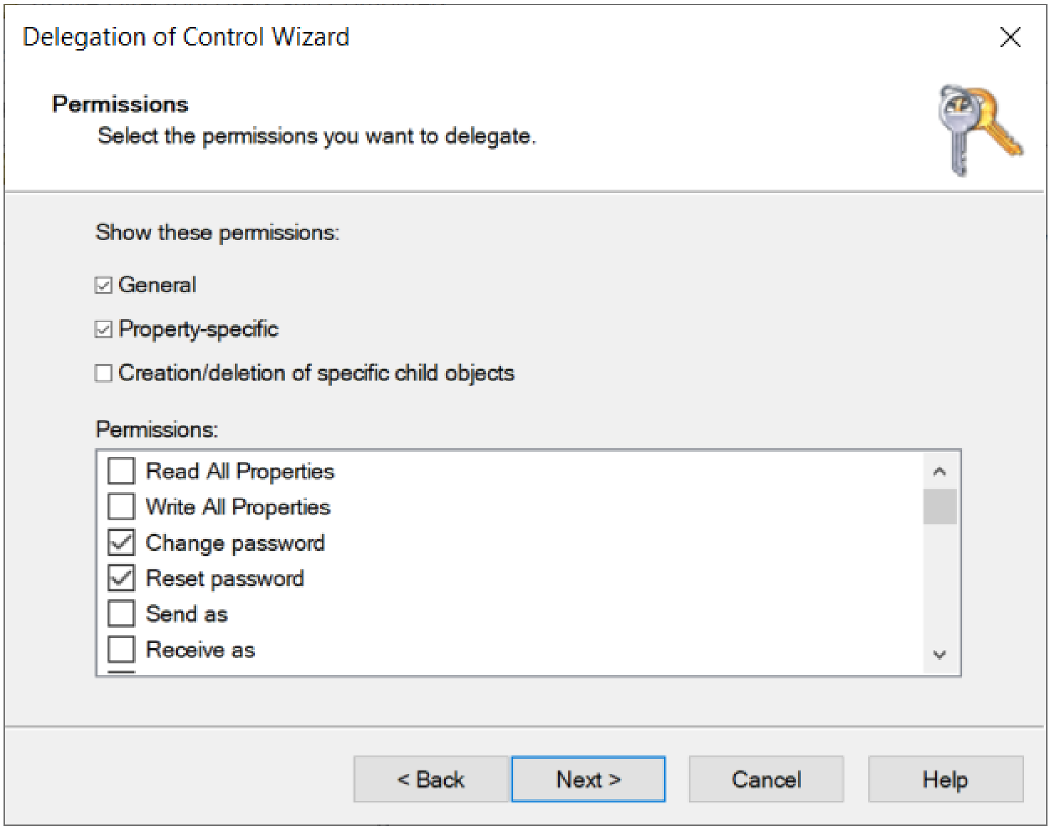
● Only the following objects in the folder:

- □ Site Settings objects
- □ Sites Container objects
- □ Subnet objects
- □ Subnets Container objects
- □ Trusted Domain objects
- ☑ User objects

□ Create selected objects in this folder

□ Delete selected objects in this folder

Under “Show these permissions” check “General” and “Property-specific”
In the list “Permissions:” check “Change password”, “Reset password”,“Read altSecurityIdentities”, “Write altSecurityIdentities”, “Read userCertificate”, “Write userCertificate”. Then click “Next”



Delegation of Control Wizard

Permissions

Select the permissions you want to delegate.

Show these permissions:

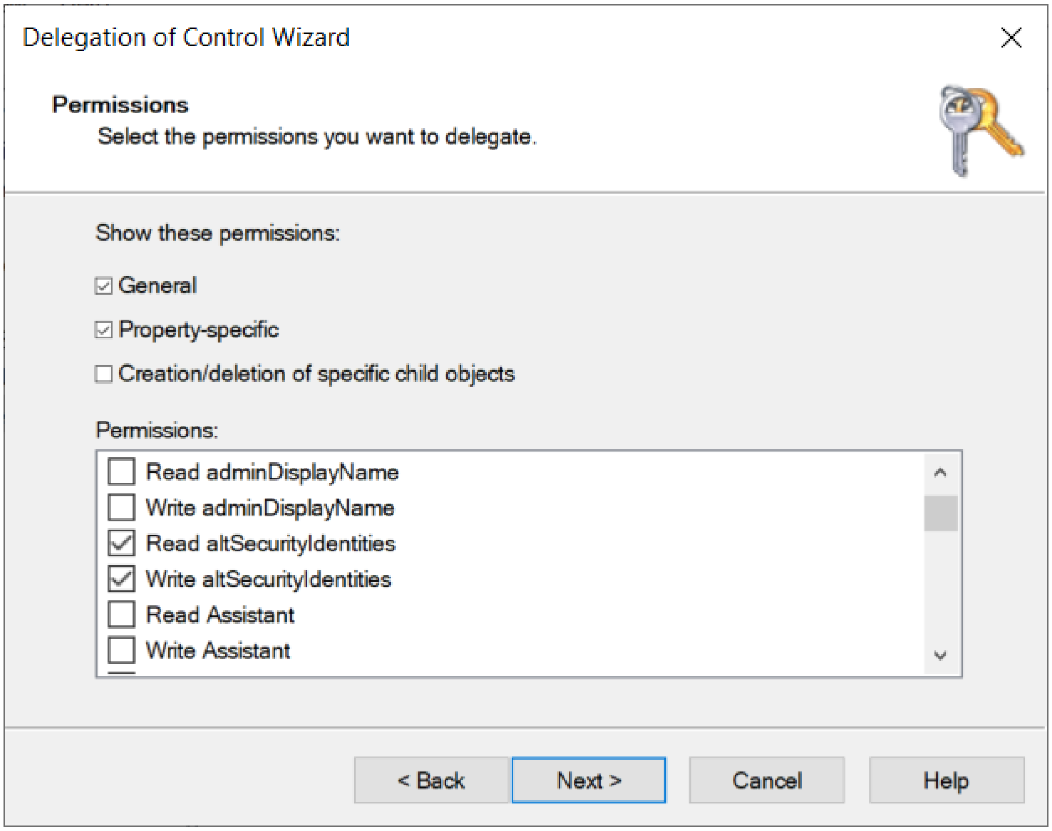
☑ General

☑ Property-specific

□ Creation/deletion of specific child objects

Permissions:

- □ Read All Properties
- □ Write All Properties
- ☑ Change password
- ☑ Reset password
- □ Send as
- □ Receive as



Delegation of Control Wizard

Permissions

Select the permissions you want to delegate.

Show these permissions:

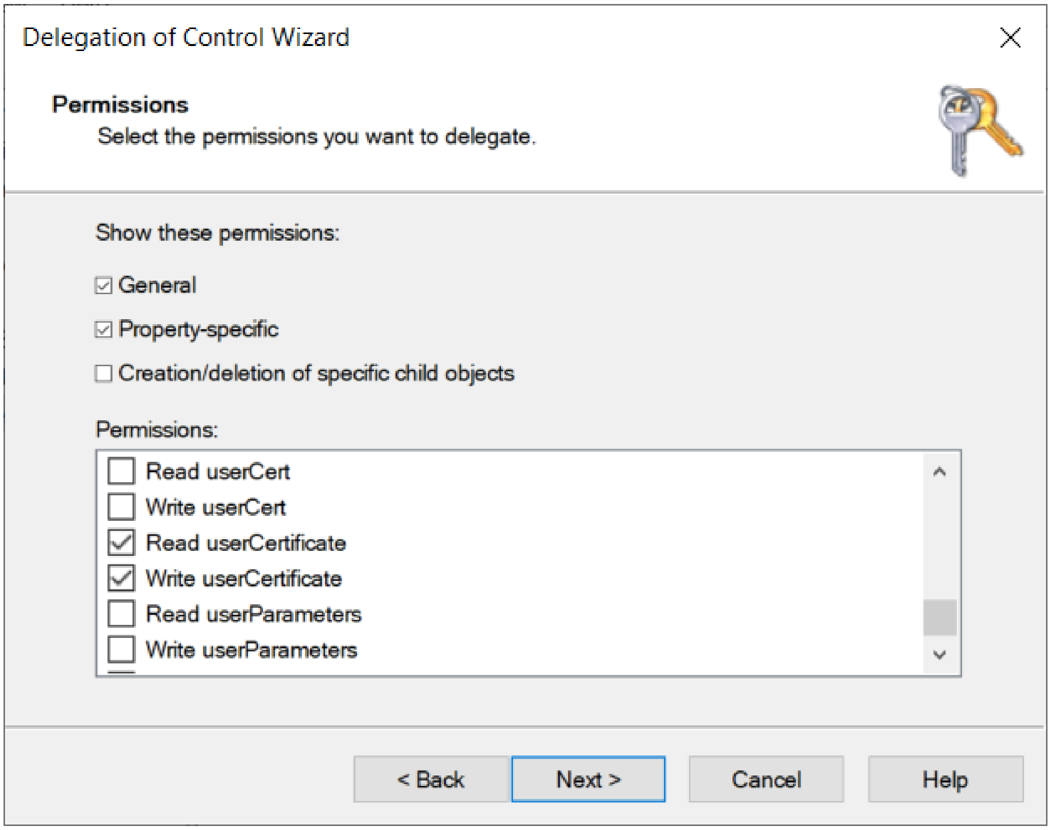
☑ General

☑ Property-specific

☐ Creation/deletion of specific child objects

Permissions:

☐ Read adminDisplayName
☐ Write adminDisplayName
☑ Read altSecurityIdentities
☑ Write altSecurityIdentities
☐ Read Assistant
☐ Write Assistant



Delegation of Control Wizard

Permissions

Select the permissions you want to delegate.

Show these permissions:

☑ General

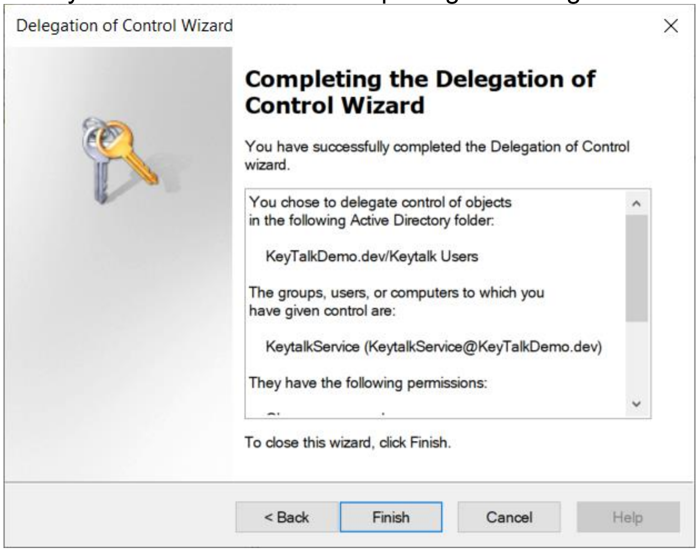
☑ Property-specific

☐ Creation/deletion of specific child objects

Permissions:

☐ Read userCert
☐ Write userCert
☑ Read userCertificate
☑ Write userCertificate
☐ Read userParameters
☐ Write userParameters

Finally click “Finish” in the “Completing the Delegation of Control Wizard” window.



Completing the Delegation of Control Wizard

You have successfully completed the Delegation of Control wizard.

You chose to delegate control of objects in the following Active Directory folder:

KeyTalkDemo.dev/Keytalk Users

The groups, users, or computers to which you have given control are:

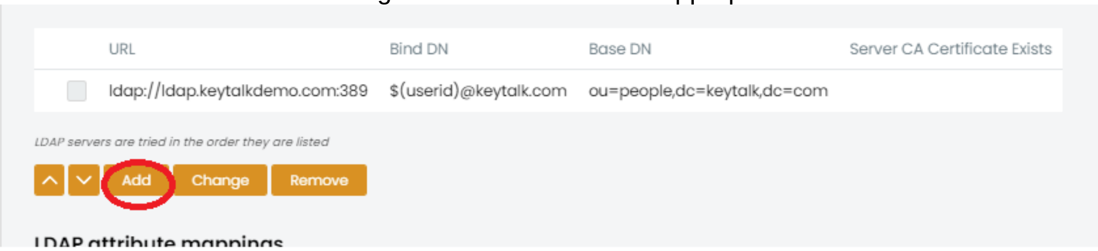
KeytalkService (KeytalkService@KeyTalkDemo.dev)

They have the following permissions:

To close this wizard, click Finish.

13.2.4 Advanced configuration steps: LDAP/AD clustering, Load Balancing

u) Now that a single LDAP/AD based working connector is in place, multiple LDAP/ADs can be added to KeyTalk using similar configuration steps as outlined above.
KeyTalk allows for each individual LDAP/AD to be added using:
Authentication -> LDAP -> Configure selected LDAP for appropriate TEMPLATE



	URL	Bind DN	Base DN	Server CA Certificate Exists
□	ldap://ldap.keytalkdemo.com:389	$(userid)@keytalk.com	ou=people,dc=keytalk,dc=com	

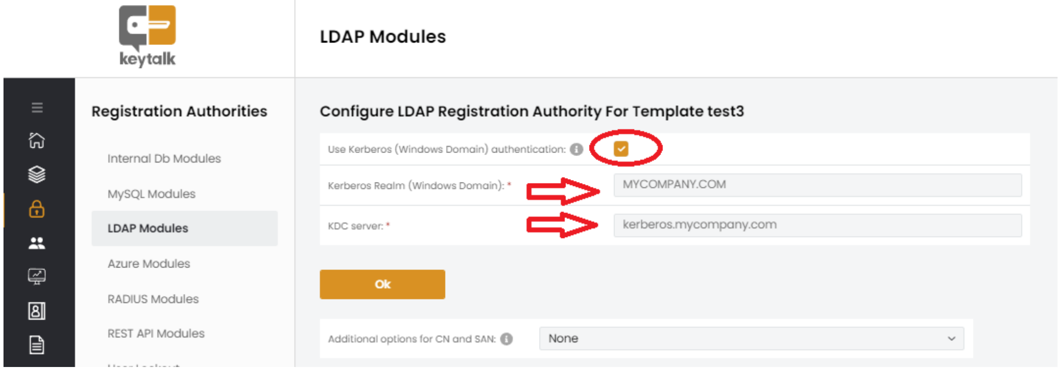
LDAP servers are tried in the order they are listed

Using the Up and Down arrows sets the order in which KeyTalk queries the individual LDAP/ADs in case of availability time-out

v) Instead of configuring each individual AD/LDAP ofcourse a LoadBalanced url can be used as well.
Do note that due to the use of LDAPS over TLS, your used LoadBalanced url or IP must be present in the LDAP/AD LDAPS certificate Subject Alternative Name. See item o).

13.2.5 Advanced configuration steps: Kerberos based authentication

With the AD connector now properly working, the KeyTalk AD connector can be enhanced by enabling Kerberos based authentication.
Provided an end-user’s Windows 7-10 device is domain joined, a clientless experience is possible as long as a valid Kerberos token is present.
To enable Kerberos simply activate it in the LDAP Authentication module :



Configure LDAP Registration Authority For Template test3

Use Kerberos (Windows Domain) authentication: ☑

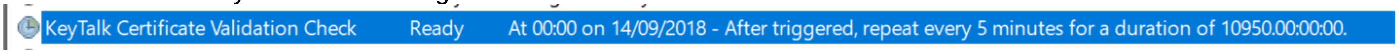
Kerberos Realm (Windows Domain): * MYCOMPANY.COM

KDC server: * kerberos.mycompany.com

Additional options for CN and SAN: None

If the Windows KeyTalk client is unable to authenticate using Kerberos is will fall back to regular username/password authentication.

The KeyTalk Windows agent adds a scheduled task to the Windows 10 Task Scheduler:



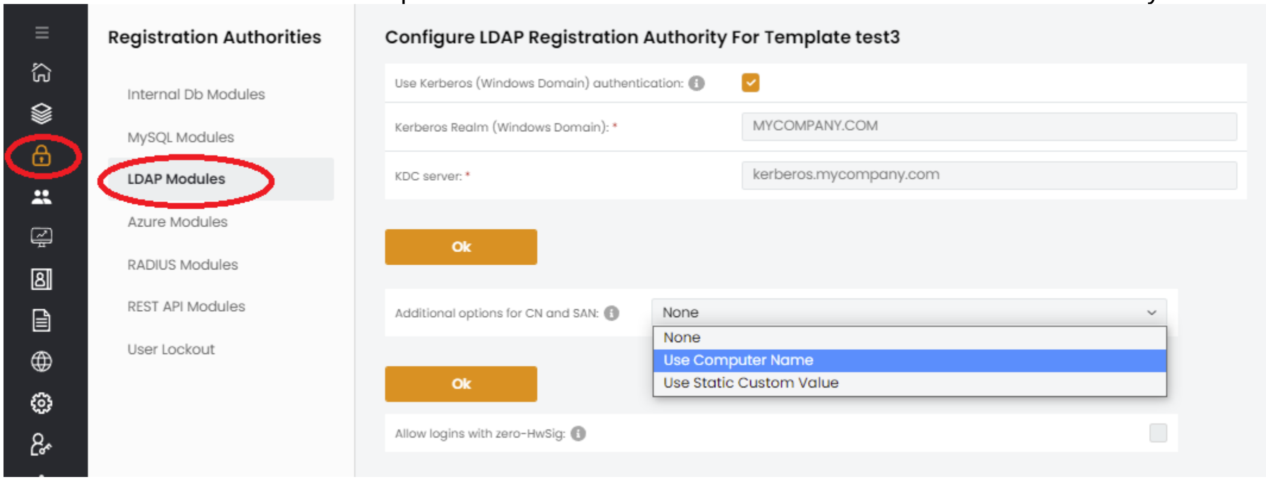
KeyTalk Certificate Validation Check Ready At 00:00 on 14/09/2018 - After triggered, repeat every 5 minutes for a duration of 10950.00:00:00.

This task triggers a certificate validation check every 5 minutes so see if the certificate has been revoked or is about to expire.
Should certificate renewal be triggered and Kerberos based authentication is enabled, the user will not see any visual cues that certificate renewal is taking place.
When a user triggers the KeyTalk client and Kerberos authentication is successful only a visible message will appear that the certificate was successfully obtained based on Kerberos authentication.

Note: Currently only Windows end points for clients are supported for the use of Kerberos authentication

13.2.6 Replacing AD user/hostname with machinename value or static value in the certificate CN

w) Many companies have not registered their user machinenames in their AD. But still want to be able to issue machine certificates, or other types of certificates based on a user's password or Kerberos authentication.
To support these use cases, the KeyTalk AD connector supports exchanging a user's AD username in the CN with either a static value , or with their machinename value. (currently only supported for Windows, contact sales@keytalk.com for support on other OS)
This alternative Common Name option also ensures that the set CN is added as a DNS SAN entry.



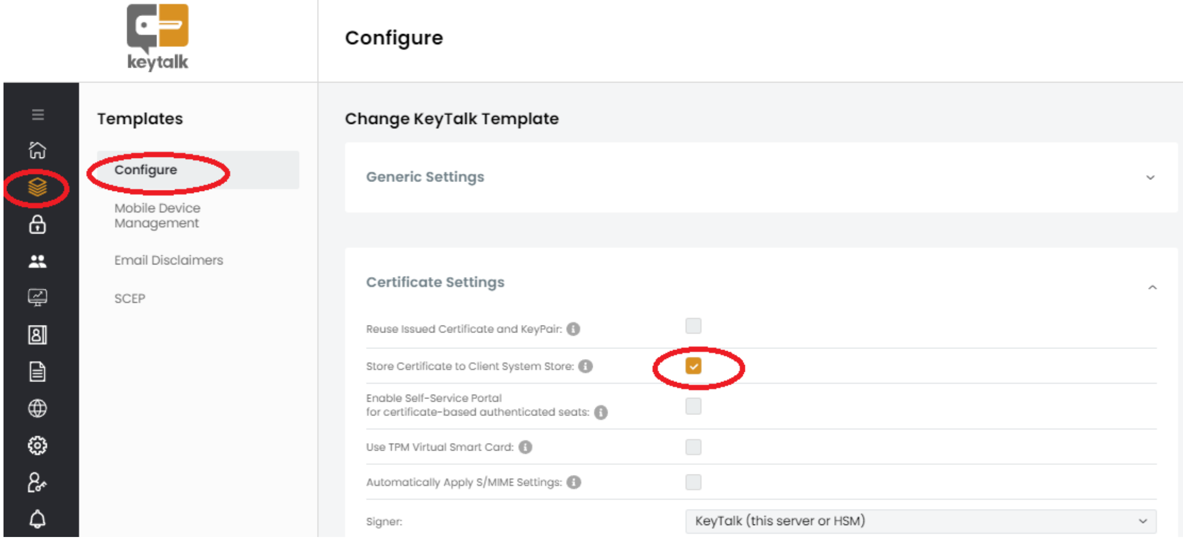
Configure LDAP Registration Authority For Template test3

Use Kerberos (Windows Domain) authentication: ☑
Kerberos Realm (Windows Domain): * MYCOMPANY.COM
KDC server: * kerberos.mycompany.com

Additional options for CN and SAN: None
- None
- Use Computer Name
- Use Static Custom Value

Allow logins with zero-HwSig: ☐

Since machine certificates often require that they reside in the Certificate Client System Store of Windows, ensure that this option is checks under the connected TEMPLATE:



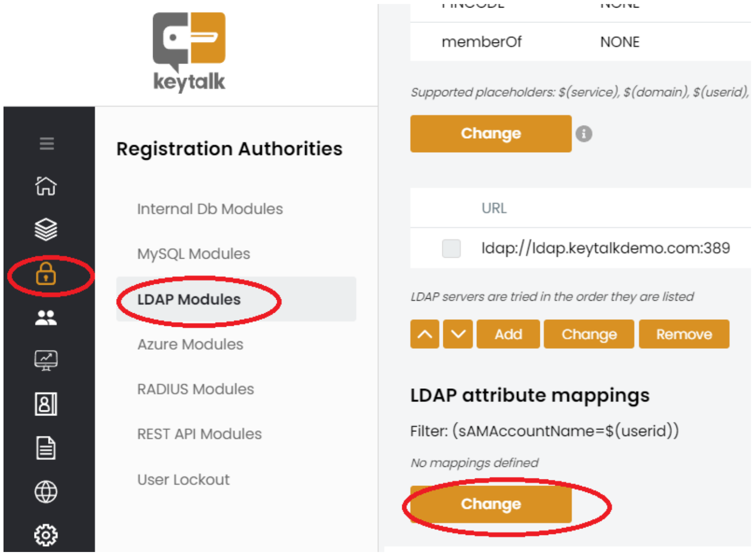
Change KeyTalk Template

Certificate Settings

Reuse Issued Certificate and KeyPair: ☐
Store Certificate to Client System Store: ☑
Enable Self-Service Portal for certificate-based authenticated seats: ☐
Use TPM Virtual Smart Card: ☐
Automatically Apply S/MIME Settings: ☐
Signer: KeyTalk (this server or HSM)

13.2.7 Overwriting default certificate template values with Active Directory Attribute values

No matter what DEFAULT certificate template settings you made (see chapter 12.1 and 12.2) , all these values (provided the CSP and 3rd party CA product allows for it), can be overwritten on a unique end-point username level, by making use of the advanced KeyTalk AUTHENTICATION feature:
<u>AUTHENTICATION MODULE CERTIFICATE ATTRIBUTE MAPPING for **LDAP/AD**</u>**:**



memberOf	NONE

Supported placeholders: $(service), $(domain), $(userid),

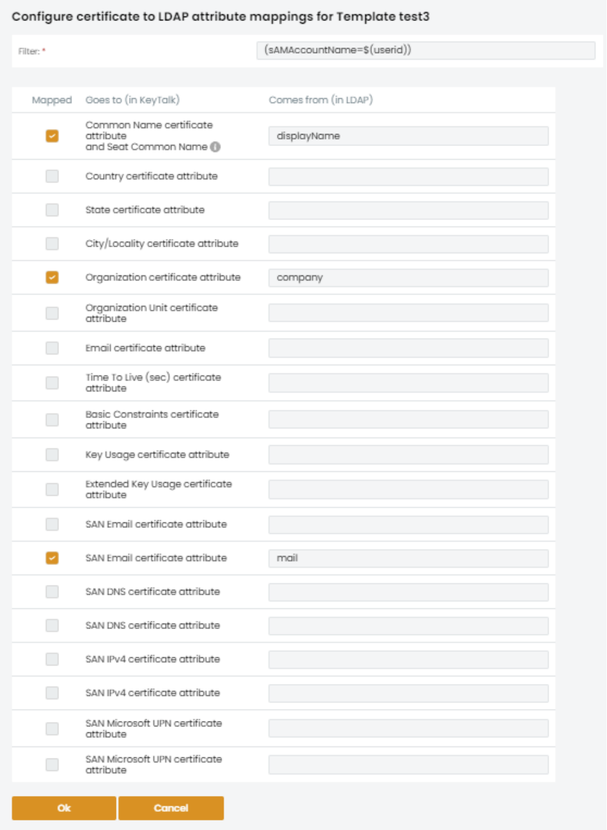
	URL
□	ldap://ldap.keytalkdemo.com:389

LDAP servers are tried in the order they are listed

LDAP attribute mappings

Filter: (sAMAccountName=$(userid))

No mappings defined



Configure certificate to LDAP attribute mappings for Template test3

Filter: * (sAMAccountName=$(userid))

Mapped	Goes to (in KeyTalk)	Comes from (in LDAP)
☑	Common Name certificate attribute and Seat Common Name	displayName
□	Country certificate attribute	
□	State certificate attribute	
□	City/Locality certificate attribute	
☑	Organization certificate attribute	company
□	Organization Unit certificate attribute	
□	Email certificate attribute	
□	Time To Live (sec) certificate attribute	
□	Basic Constraints certificate attribute	
□	Key Usage certificate attribute	
□	Extended Key Usage certificate attribute	
□	SAN Email certificate attribute	
☑	SAN Email certificate attribute	mail
□	SAN DNS certificate attribute	
□	SAN DNS certificate attribute	
□	SAN IPv4 certificate attribute	
□	SAN IPv4 certificate attribute	
□	SAN Microsoft UPN certificate attribute	
□	SAN Microsoft UPN certificate attribute	

In the above example my TEMPLATE default value for Organization Unit is overwritten (provided a valid value exists) with a value coming from my AD/LDAP attribute field “company” for any particular account positively authenticating the TEMPLATE’s RA.

Its likely that instead of using a **CN** based searchfilter, you may need **sAMAccountName** or **userPrincipalName**

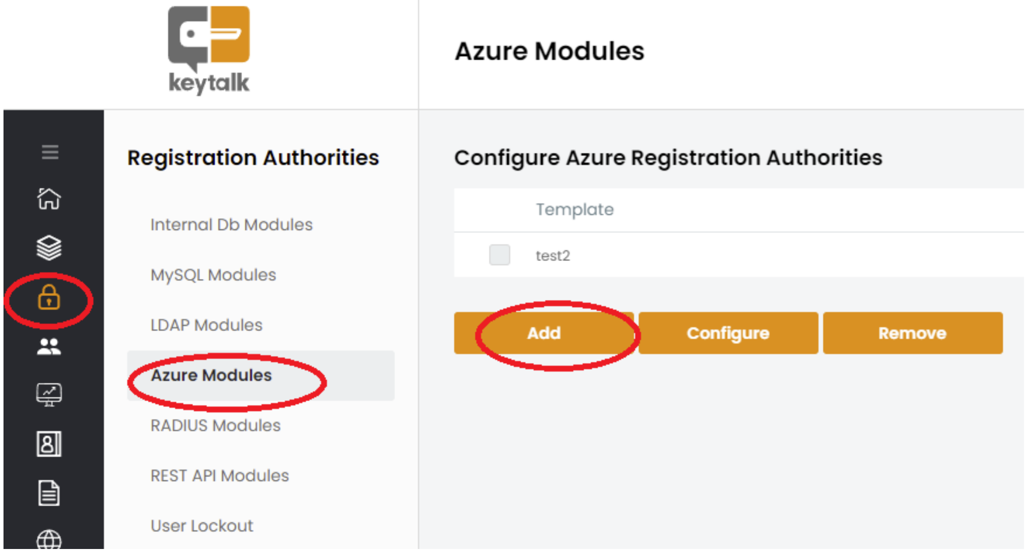
13.3 Azure Active Directory (AAD)

With Azure AD (AAD) username/password validation is supported. (MFA is not yet supported)

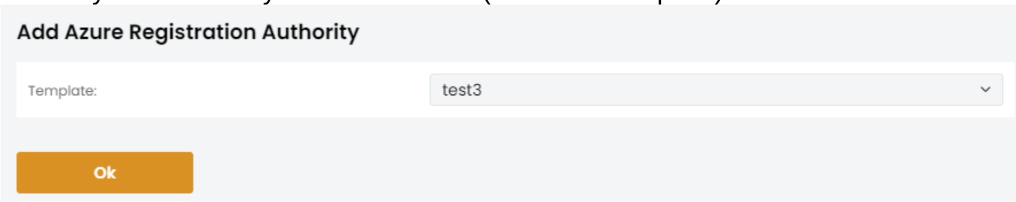
13.3.1 Add and configure a KeyTalk Authentication Azure Module connection

You must have an Office 365 Business in order to make the connection!!

Select: Registration Authorities -> Azure Modules -> Add

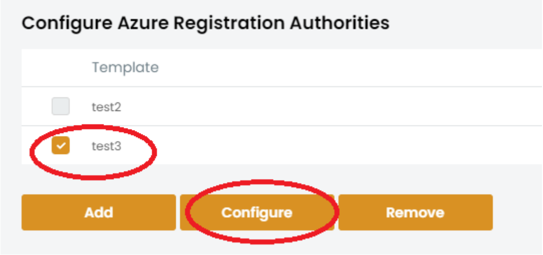


Ensure you have a KeyTalk TEMPLATE (certificate template) available to connect to the Azure module:

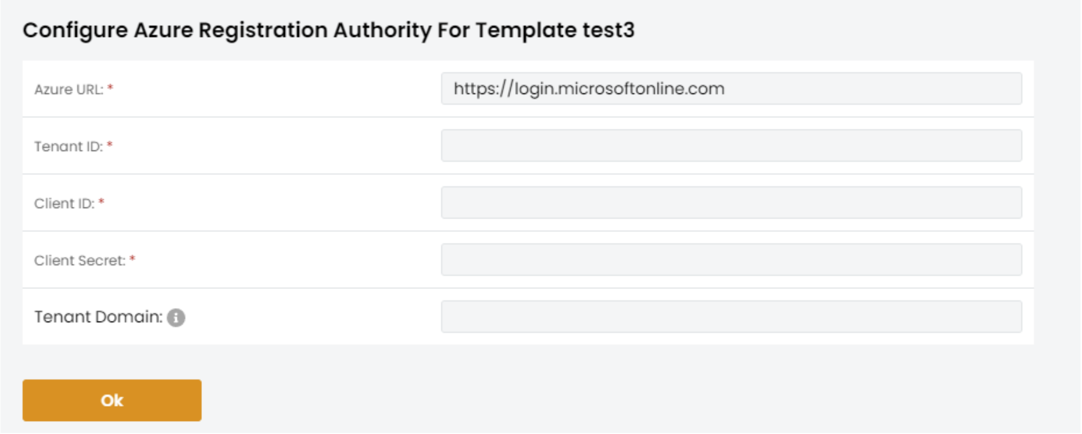

Add Azure Registration Authority

Template: test3

Now configure the Azure Module:



Enter the appropriate details to enable a connection your Azure AD. See chapter 13.3.2 to ensure you have the correct connection details:


Configure Azure Registration Authority For Template test3

Azure URL: *	https://login.microsoftonline.com
Tenant ID: *	
Client ID: *	
Client Secret: *	
Tenant Domain: ⓘ	

13.3.2 Configure Azure Active Directory for use with KeyTalk

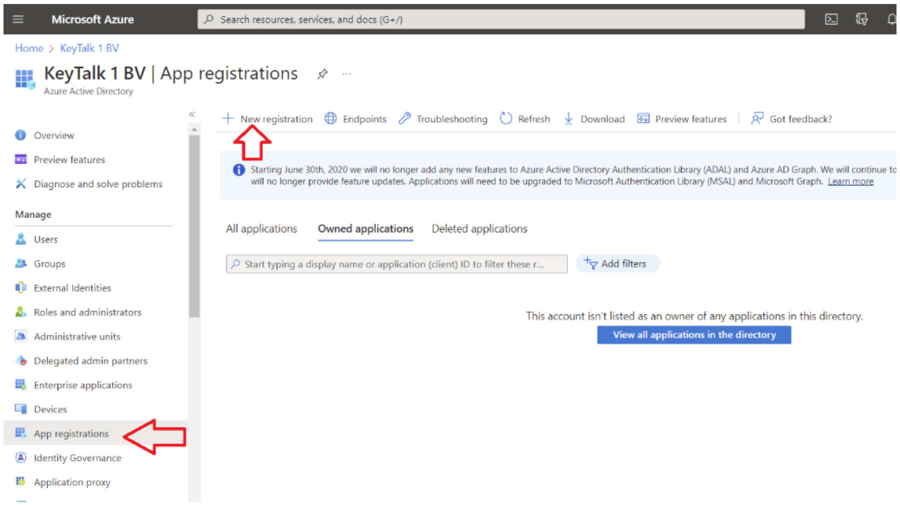
Azure AD allows non-Microsoft applications to make use of account authentication provided these applications have been registered within O365 Azure AD.

Once registered, the resulting credentials are used to configure the connector from KeyTalk Azure Module to AAD (see end of chapter 13.3.1)

Step 1: Login to https://portal.azure.com/

Step 2: Select Azure Active Directory

Step 3: Select app registration and select New registration:



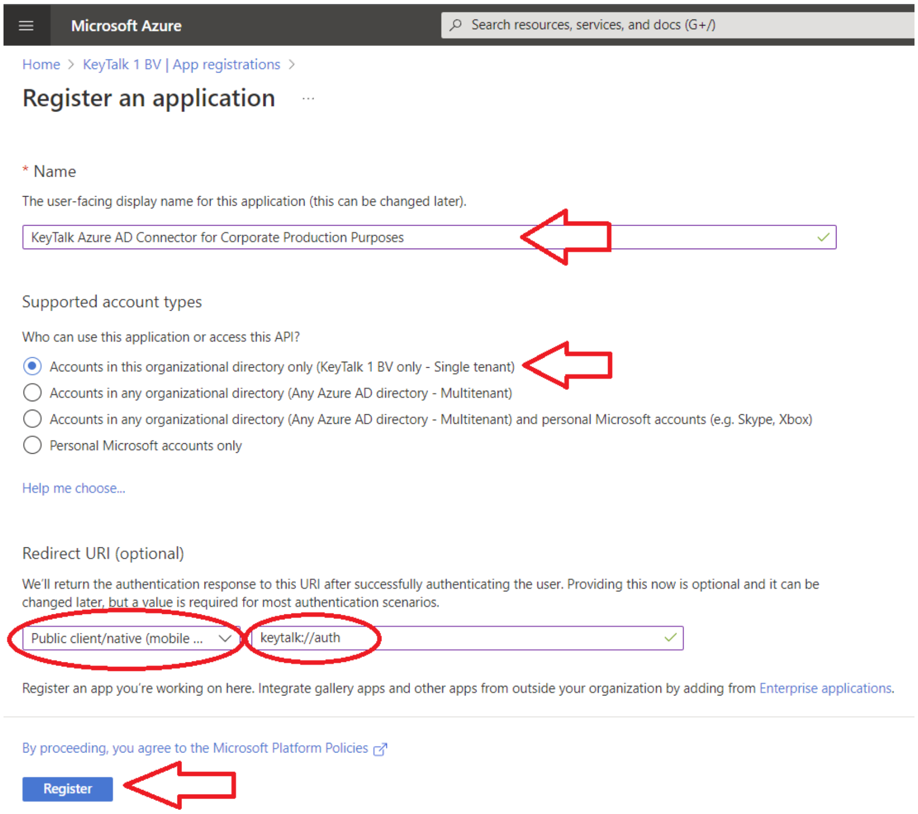
KeyTalk 1 BV | App registrations

Azure Active Directory

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to
will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more

This account isn't listed as an owner of any applications in this directory.

Step 4: Configure the app registration



Microsoft Azure
Search resources, services, and docs (G+/)
Home > KeyTalk 1 BV | App registrations >
Help me choose...
Enterprise applications
By proceeding, you agree to the Microsoft Platform Policies
Register

Register an application

* Name

The user-facing display name for this application (this can be changed later).

KeyTalk Azure AD Connector for Corporate Production Purposes

Supported account types

Who can use this application or access this API?

- ● Accounts in this organizational directory only (KeyTalk 1 BV only - Single tenant)
- ○ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- ○ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- ○ Personal Microsoft accounts only

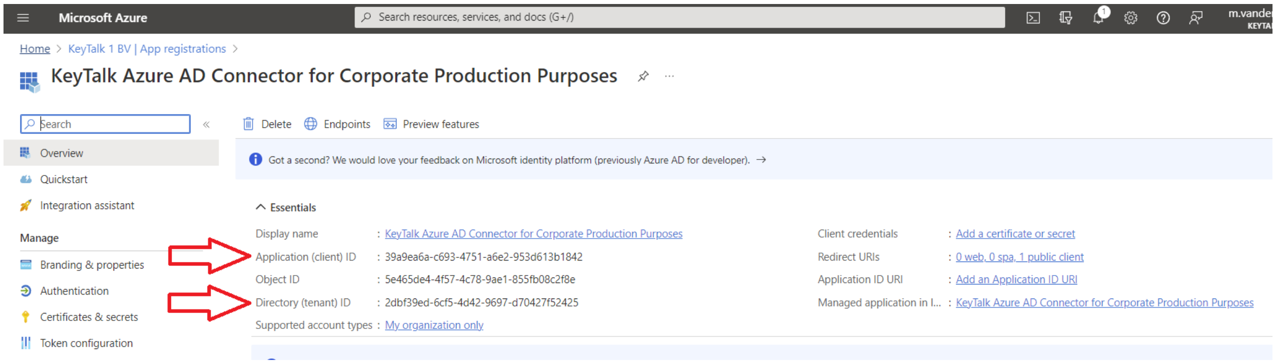
Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Public client/native (mobile ... | keytalk://auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from Enterprise applications.

Step 5: Write down the **client** and **tenant ID**



Home > KeyTalk 1 BV | App registrations >

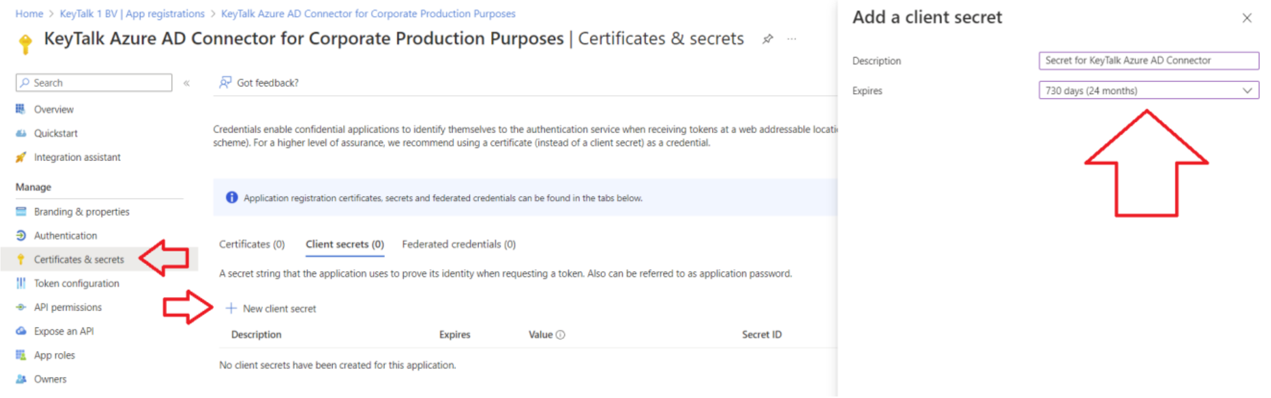
KeyTalk Azure AD Connector for Corporate Production Purposes

Got a second? We would love your feedback on Microsoft identity platform (previously Azure AD for developer). →

Essentials

Display name	: KeyTalk Azure AD Connector for Corporate Production Purposes	Client credentials	: Add a certificate or secret
Application (client) ID	: 39a9ea6a-c693-4751-a6e2-953d613b1842	Redirect URIs	: 0 web, 0 spa, 1 public client
Object ID	: 5e465de4-4f57-4c78-9ae1-855fb08c2f8e	Application ID URI	: Add an Application ID URI
Directory (tenant) ID	: 2dbf39ed-6cf5-4d42-9697-d70427f52425	Managed application in l...	: KeyTalk Azure AD Connector for Corporate Production Purposes
Supported account types	: My organization only		

Step 6: Configure a client secret and note the set client secret and expiration date (!!) and set a reminder to set the new client secret before it expires (!!)



Home > KeyTalk 1 BV | App registrations > KeyTalk Azure AD Connector for Corporate Production Purposes

KeyTalk Azure AD Connector for Corporate Production Purposes | Certificates & secrets

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable locati scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) | Client secrets (0) | Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

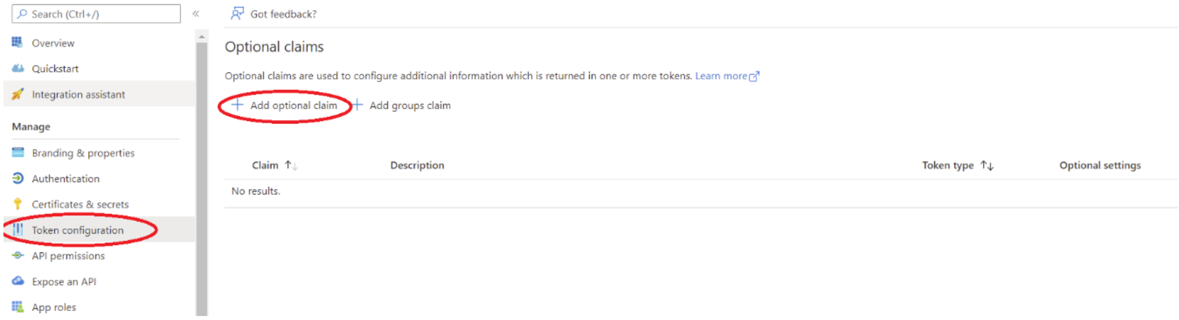
Description	Expires	Value	Secret ID

No client secrets have been created for this application.

Add a client secret

Description	Secret for KeyTalk Azure AD Connector
Expires	730 days (24 months)

Step 7: Configure an ACCESS token claim



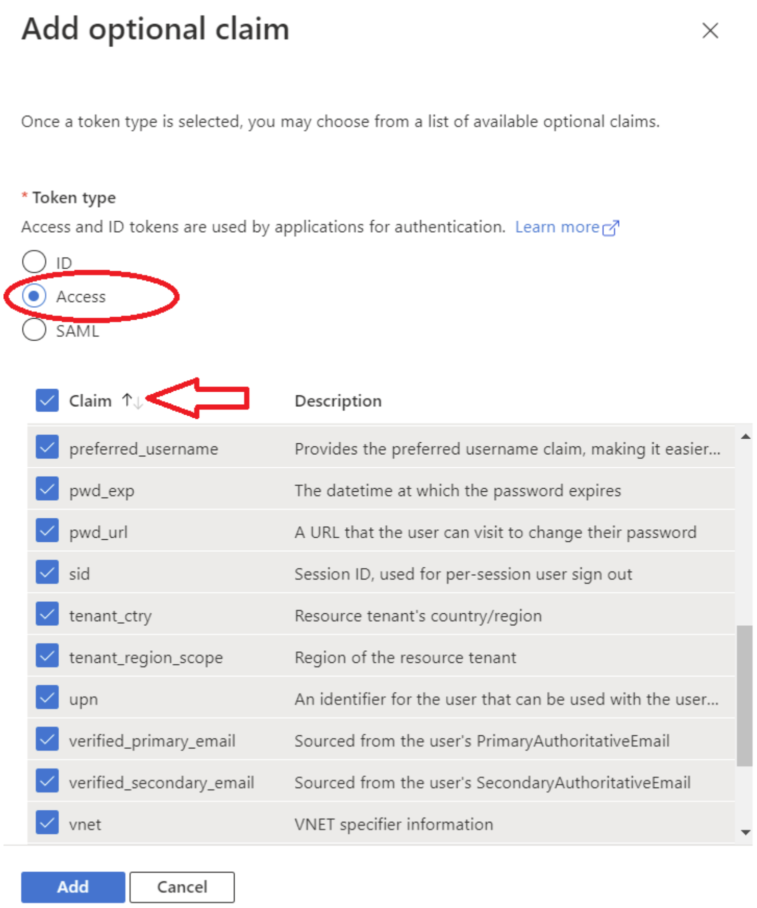
Optional claims

Optional claims are used to configure additional information which is returned in one or more tokens. Learn more

+ Add optional claim + Add groups claim

Claim ↑↓	Description	Token type ↑↓	Optional settings

No results.



Add optional claim

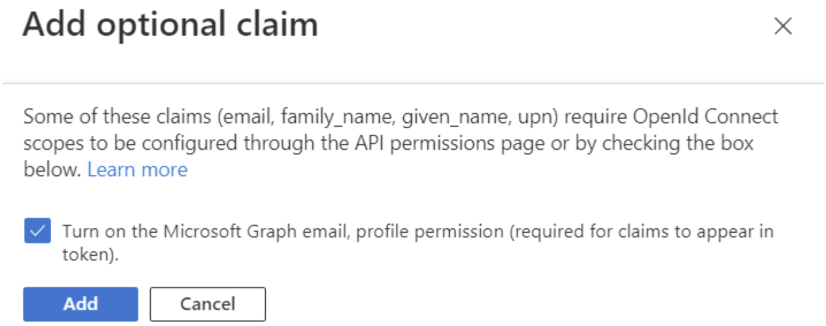
Once a token type is selected, you may choose from a list of available optional claims.

* Token type

Access and ID tokens are used by applications for authentication. Learn more

○ ID
● Access
○ SAML

	Claim ↑↓	Description
☑	preferred_username	Provides the preferred username claim, making it easier...
☑	pwd_exp	The datetime at which the password expires
☑	pwd_url	A URL that the user can visit to change their password
☑	sid	Session ID, used for per-session user sign out
☑	tenant_ctry	Resource tenant's country/region
☑	tenant_region_scope	Region of the resource tenant
☑	upn	An identifier for the user that can be used with the user...
☑	verified_primary_email	Sourced from the user's PrimaryAuthoritativeEmail
☑	verified_secondary_email	Sourced from the user's SecondaryAuthoritativeEmail
☑	vnet	VNET specifier information

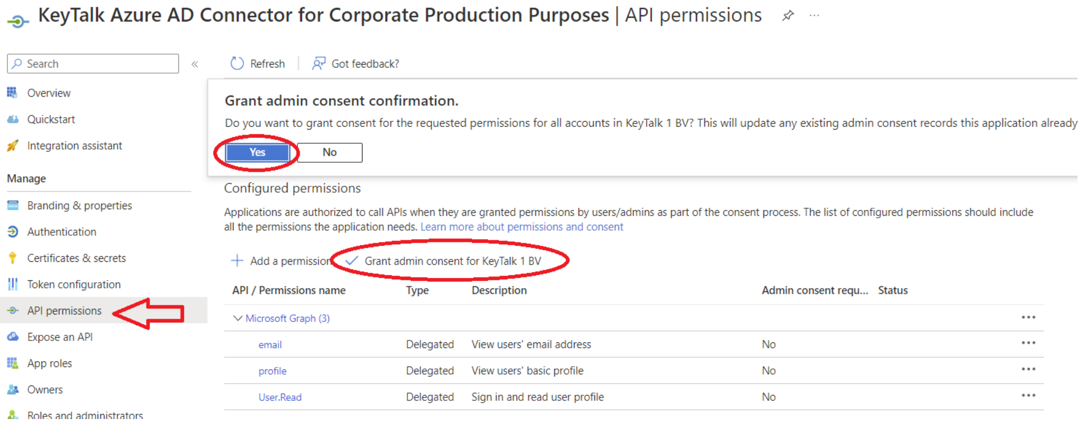


Add optional claim

Some of these claims (email, family_name, given_name, upn) require OpenId Connect scopes to be configured through the API permissions page or by checking the box below. Learn more

☑ Turn on the Microsoft Graph email, profile permission (required for claims to appear in token).

Step 7: Configure API permissions, Grant Admin consent:



KeyTalk Azure AD Connector for Corporate Production Purposes | API permissions

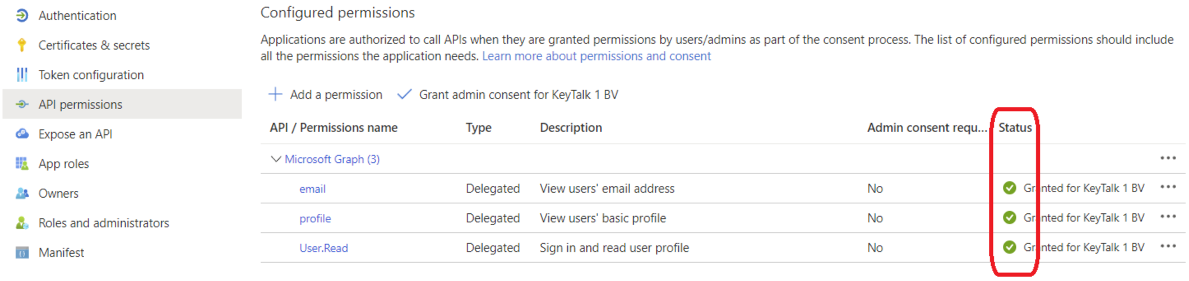
Grant admin consent confirmation.

Do you want to grant consent for the requested permissions for all accounts in KeyTalk 1 BV? This will update any existing admin consent records this application already

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (3)				
email	Delegated	View users' email address	No	
profile	Delegated	View users' basic profile	No	
User.Read	Delegated	Sign in and read user profile	No	



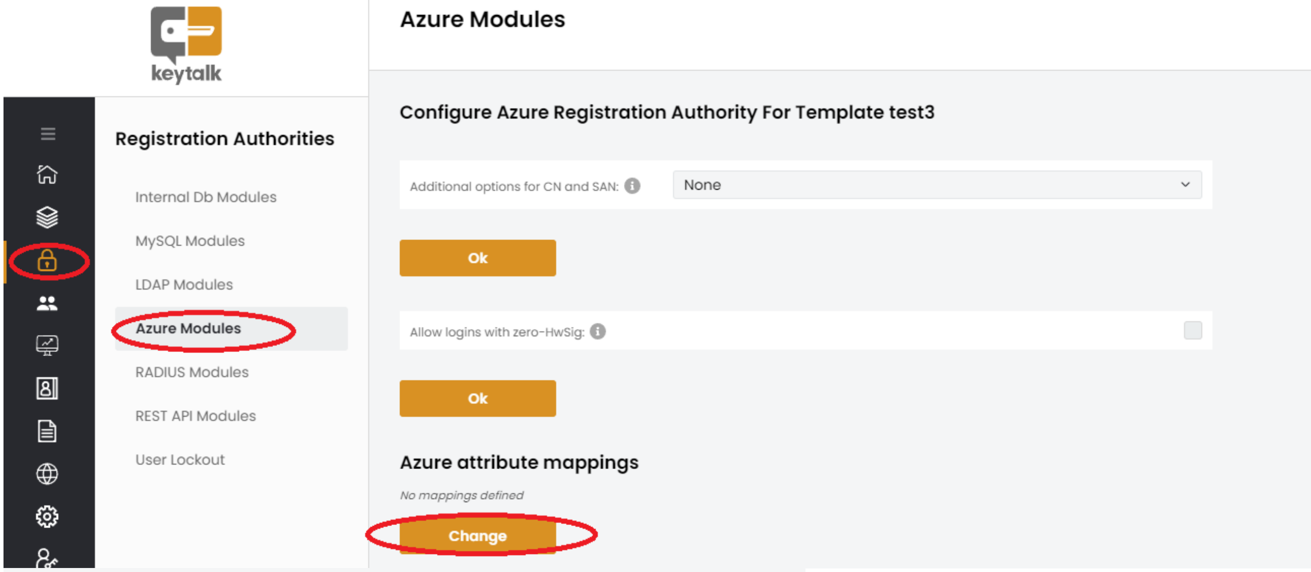
Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent requ...	Status	
Microsoft Graph (3)					•••
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV	•••
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV	•••
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV	•••

13.3.3 Configure Azure Active Directory to enable attribute mapping to certificate fields

Accounts in Azure Active Directory often contain a lot of information which are unique to the account and which needs to be available in an Azure AD based issued certificate.
For this purpose KeyTalk CKMS supports Azure AD attribute to certificate field mapping.



Azure Modules

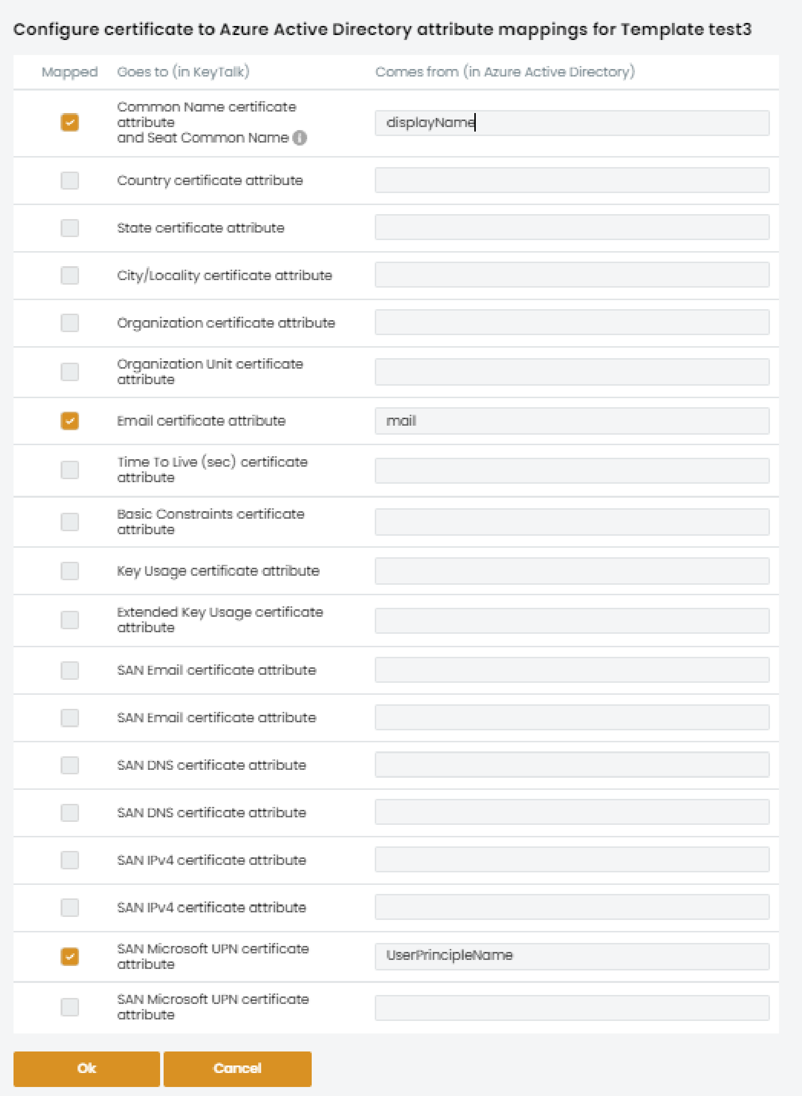
Configure Azure Registration Authority For Template test3

Additional options for CN and SAN: None

Allow logins with zero-HwSig: □

Azure attribute mappings

No mappings defined



Configure certificate to Azure Active Directory attribute mappings for Template test3

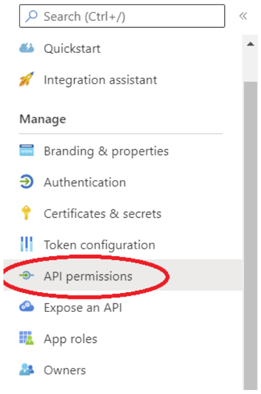
Mapped	Goes to (in KeyTalk)	Comes from (in Azure Active Directory)
☑	Common Name certificate attribute and Seat Common Name	displayName
□	Country certificate attribute	
□	State certificate attribute	
□	City/Locality certificate attribute	
□	Organization certificate attribute	
□	Organization Unit certificate attribute	
☑	Email certificate attribute	mail
□	Time To Live (sec) certificate attribute	
□	Basic Constraints certificate attribute	
□	Key Usage certificate attribute	
□	Extended Key Usage certificate attribute	
□	SAN Email certificate attribute	
□	SAN Email certificate attribute	
□	SAN DNS certificate attribute	
□	SAN DNS certificate attribute	
□	SAN IPv4 certificate attribute	
□	SAN IPv4 certificate attribute	
☑	SAN Microsoft UPN certificate attribute	UserPrincipleName
□	SAN Microsoft UPN certificate attribute	

In order to read these attributes, the KeyTalk CKMS must be authorized to read these attributes.
To enable the required authorization for reading the attributes, follow the following steps:

1. Log in to the Azure portal and navigate to Azure Active Directory

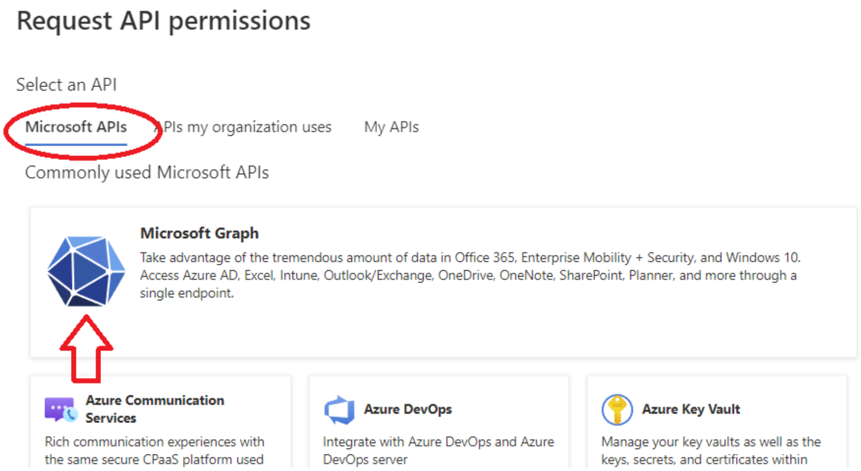
2. Select App Registrations in the Manage section and select the client app to use (or register a new app with ‘New Registration’)

3. From the app overview page select API Permissions in the Manage section or select ‘View API Permissions’



4. Ensure the following **Microsoft Graph API** permissions are present:
User.Read (**Delegated permissions**) , User.Read.All (**delegated permissions**) and User.Read.All (**Application permissions**)

When these permissions are missing, select ‘Add a Permission’



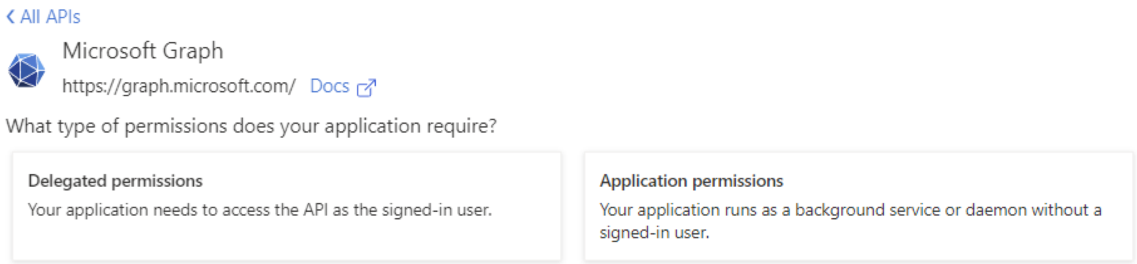
Request API permissions

Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

Azure Communication Services
Rich communication experiences with the same secure CPaaS platform used

Azure DevOps
Integrate with Azure DevOps and Azure DevOps server

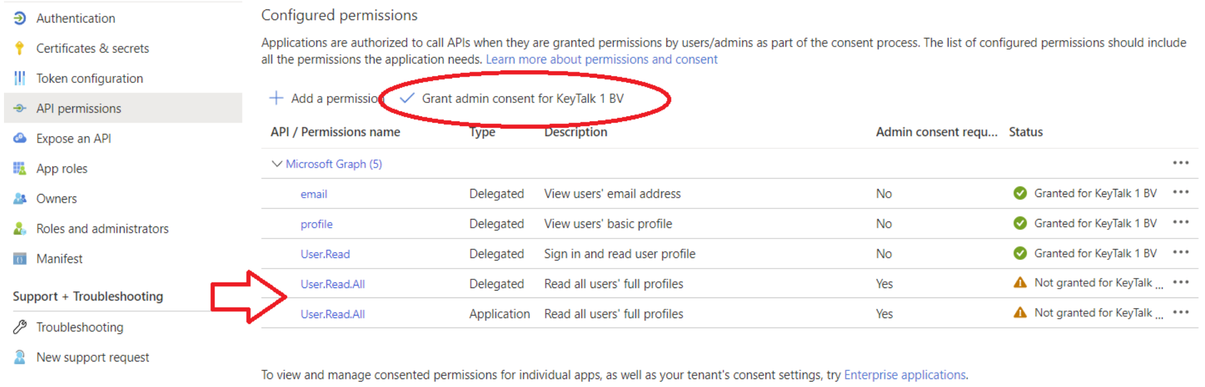
Azure Key Vault
Manage your key vaults as well as the keys, secrets, and certificates within



What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.



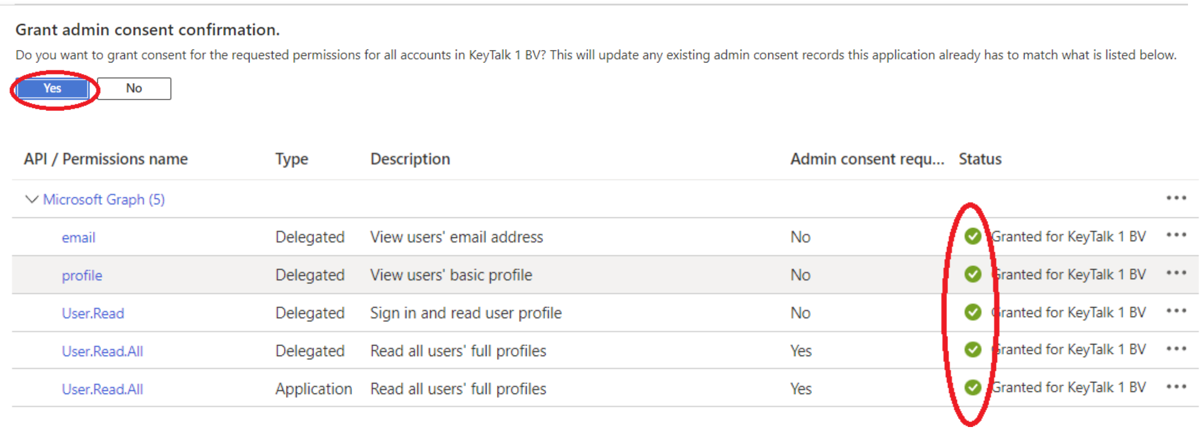
Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (5)				
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV
User.Read.All	Delegated	Read all users' full profiles	Yes	⚠ Not granted for KeyTalk ...
User.Read.All	Application	Read all users' full profiles	Yes	⚠ Not granted for KeyTalk ...

To view and manage consented permissions for individual apps, as well as your tenant's consent settings, try Enterprise applications.

After adding permissions, you need to grant admin consent for the app



Grant admin consent confirmation.

Do you want to grant consent for the requested permissions for all accounts in KeyTalk 1 BV? This will update any existing admin consent records this application already has to match what is listed below.

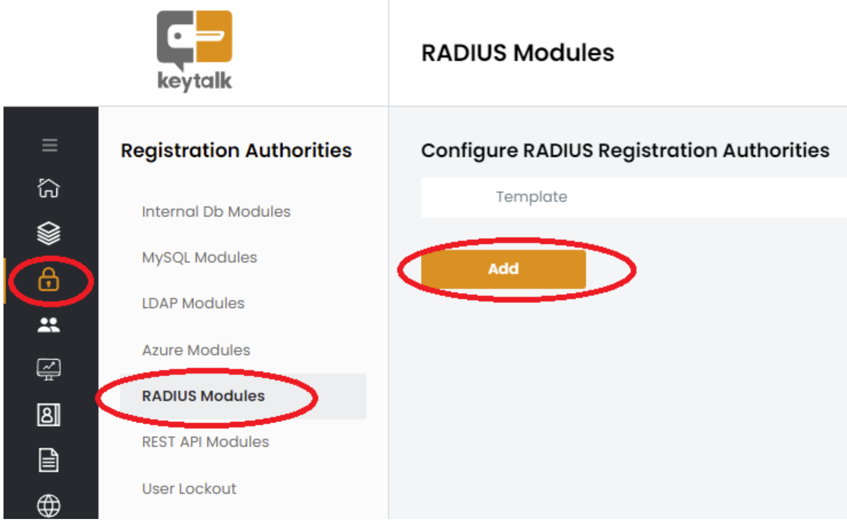
API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (5)				
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV
User.Read.All	Delegated	Read all users' full profiles	Yes	✓ Granted for KeyTalk 1 BV
User.Read.All	Application	Read all users' full profiles	Yes	✓ Granted for KeyTalk 1 BV

13.4 Radius: One Time Password tokens, Username/password, AKA SIM

Companies who deploy OTP either have these integrated into their AD, requiring KeyTalk to have its Authentication connector configured to LDAP/AD (see 13.2), or are likely to have either an on-premises or Cloud based Radius solution. Commonly used tokens are offered by RSA, Gemalto, Vasco and others.

Others might not have OTP implementations but use Radius for simple username/password purposes. Additionally, IoT providers may need AKA SIM based authentication which also relates to Radius.

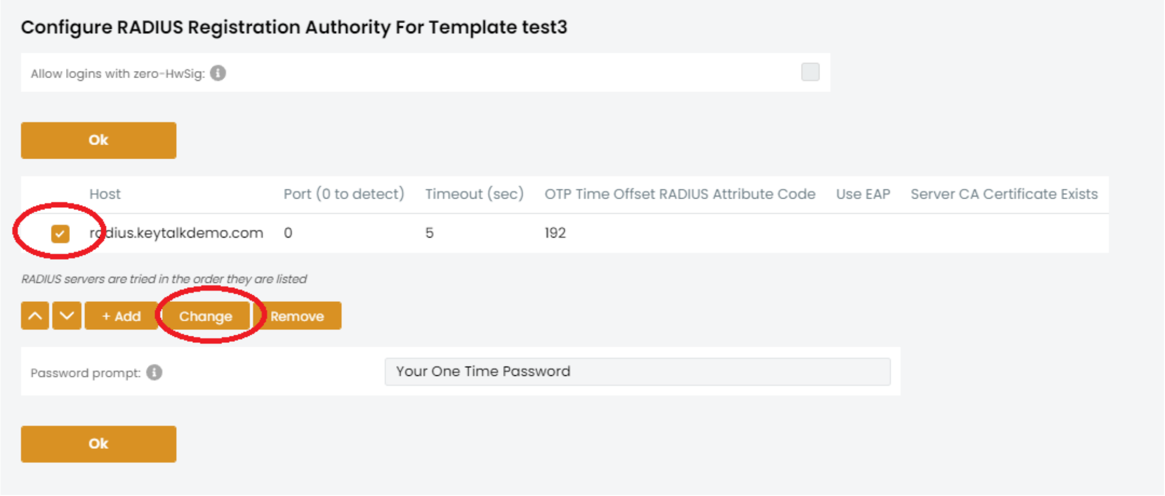
KeyTalk offers a Radius connector to serve all these use-cases.
Simply ADD a Radius Registration Authority module to a KeyTalk TEMPLATE:



The steps to configure the Radius connector are easy:

a) Ensure your firewall rules and when needed custom network routing allows the KeyTalk server to communicate with your Radius.
With some Cloud based token solutions you may need to specifically request a Radius agent url from your Cloud Token portal, and possibly whitelist your IP address(es).

b) Select the Radius server from your Authentication module:



Configure RADIUS Registration Authority For Template test3

Allow logins with zero-HwSig: □

Ok

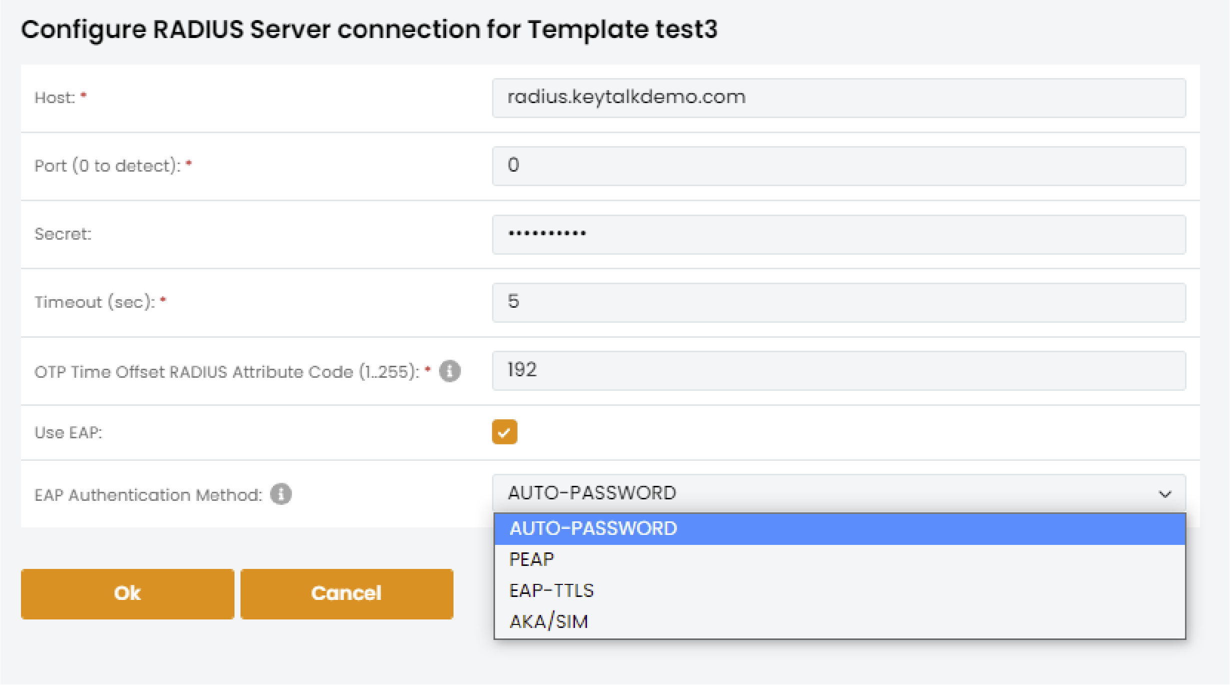
	Host	Port (0 to detect)	Timeout (sec)	OTP Time Offset RADIUS Attribute Code	Use EAP	Server CA Certificate Exists
☑	radius.keytalkdemo.com	0	5	192		

RADIUS servers are tried in the order they are listed

Password prompt: Your One Time Password

Ok

c) Enter your appropriate Radius authentication details



Configure RADIUS Server connection for Template test3

Host: *	radius.keytalkdemo.com
Port (0 to detect): *	0
Secret:	••••••••••
Timeout (sec): *	5
OTP Time Offset RADIUS Attribute Code (1..255): *	192
Use EAP:	☑
EAP Authentication Method:	AUTO-PASSWORD

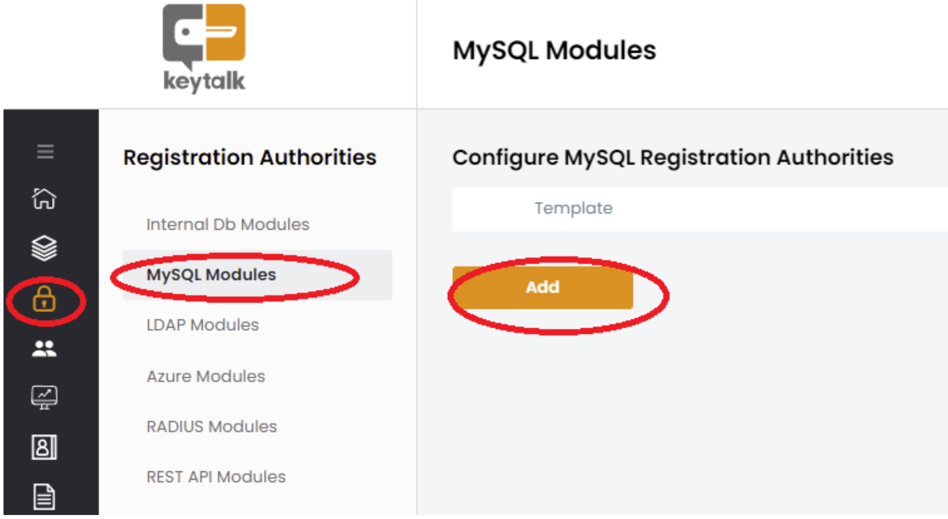
d) When enforcing EAP you must also upload in PEM format the intermediate certificate that issued your Radius SSL/TLS certificate.
Download this intermediate from your Radius server or request it with your Cloud Radius provider.
When you are unfamiliar with the conversion of an intermediate certificate into PEM contact your company Admin specialist, or convert it using OpenSSL, or send your certificate to support@keytalk.com

Now test your KeyTalk client with your Radius based authentication (see chapter 14/15)

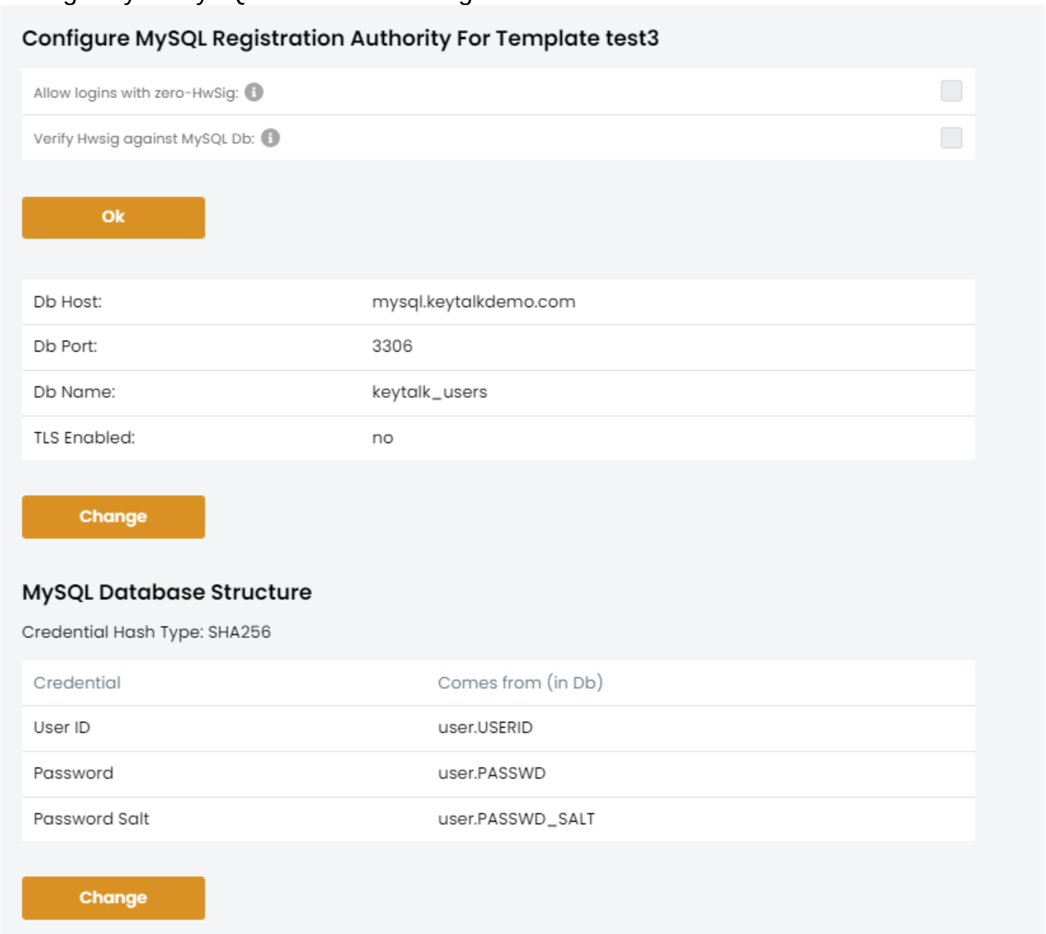
13.5 MySQL

Solution providers who make use of MySQL (or similar such as MariaDb) to authenticate their users and end-points can configure KeyTalk to connect to their MySQL or related Db infrastructure.

Go to: Registration Authorities - > MySQL -> ADD your KeyTalk TEMPLATE



Configure you MySQL connector settings:



Configure MySQL Registration Authority For Template test3

Allow logins with zero-HwSig: ℹ	□
Verify Hwsig against MySQL Db: ℹ	□

Ok

Db Host:	mysql.keytalkdemo.com
Db Port:	3306
Db Name:	keytalk_users
TLS Enabled:	no

Change

MySQL Database Structure

Credential Hash Type: SHA256

Credential	Comes from (in Db)
User ID	user.USERID
Password	user.PASSWD
Password Salt	user.PASSWD_SALT

Change

Depending on whether or not Certificate Based authentication is used against the MySQL Db, the configuration settings will vary

13.6 Other Identity Provider based authentication solutions

KeyTalk adds new authentication connectors depending on customer business-cases.

Should other connectors be required, please do not hesitate to contact your KeyTalk vendor or us directly at: sales@keytalk.com

14. Certificate renewal automation parameters

The KeyTalk app automates the certificate renewal on a just-on-time basis on servers and end-user devices.

Servers make use of a bash script on Linux, or a 1 minute timed scheduled task on Windows which are automatically added and activated once the KeyTalk client/app is activated on said server or user end-point.

With the KeyTalk solution able to automatically renew and activate new server or client certificates, this means that certificates can be effectively given a relatively short lifespan of a few seconds, up to several years.
Since Microsoft scheduled tasks can minimally be triggered once per minute, it is advisable to set certificate validity periods at least to 61 seconds, unless working with Linux where it is recommended to minimally use 3 second validity periods. For practical reasons most customers choose a minimal validity period of 5-10 minutes when using short lived certificates for transactional purposes, and use validity periods of minimally 10-12 hours for authentication purposes.

Most (Qualified) Trusted Certificate TEMPLATE Providers will enforce a certain certificate validity period such as 6 months or 1 year. The KeyTalk TEMPLATE CA selector will properly show what can and what cannot apply for the chosen CA provider.

So why are KeyTalk TEMPLATE (or the overwrite value coming from the connected Registration Authority) validity times so important?

KeyTalk client scripts trigger a renewal on one of 3 factors:

1) Revocation Pointers in the certificate indicate the certificate is no longer valid
2) A certificate validity before renewal hits its threshold
3) The certificate is missing

14.1 Validity replacement % (alternative to validity duration in absolute time)

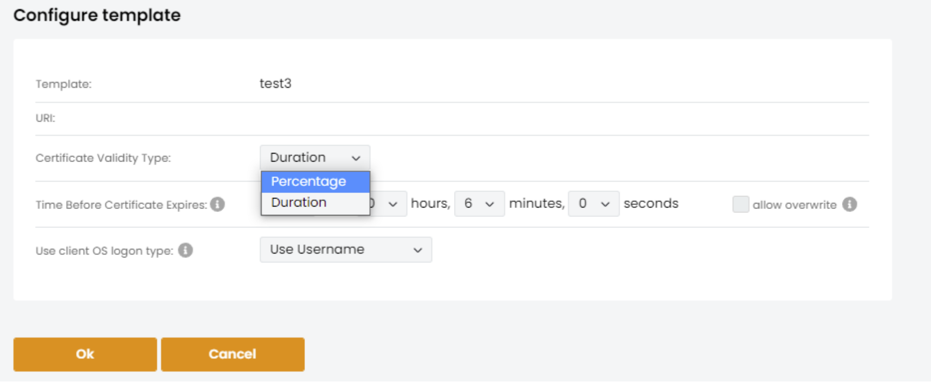
This feature is still used and supported but has been replaced as the default by absolute time (see chapter 14.2)

The certificate validity percentage is defined when creating the RCCD (the KeyTalk client configuration file).
It reflects when the renewal script must request a new certificate, based on the remaining percentage of time of the original total validity period of the certificate.

So, say your certificate is valid for 3153600 seconds, and you created your RCCD with a validity percentage of 10 (the default), then the KeyTalk script will trigger 3153600 seconds (36.5 days) BEFORE the certificate will expire.
In conclusion most admins will change the validity percentage to 1 in the RCCD when 1-year valid certificates are issued to ensure renewal doesn't happen until 3.65 days before expiry.

Say your certificate is valid for 36000 seconds, and you created your RCCD with a validity percentage of 10 (the default), then the KeyTalk script will trigger 3600 seconds (1 hour) BEFORE the certificate will expire.

The certificate validity percentage can be set under TEMPLATES -> CREATE RCCD -> EDIT



Configure template

Template: test3

URI:

Certificate Validity Type: Duration

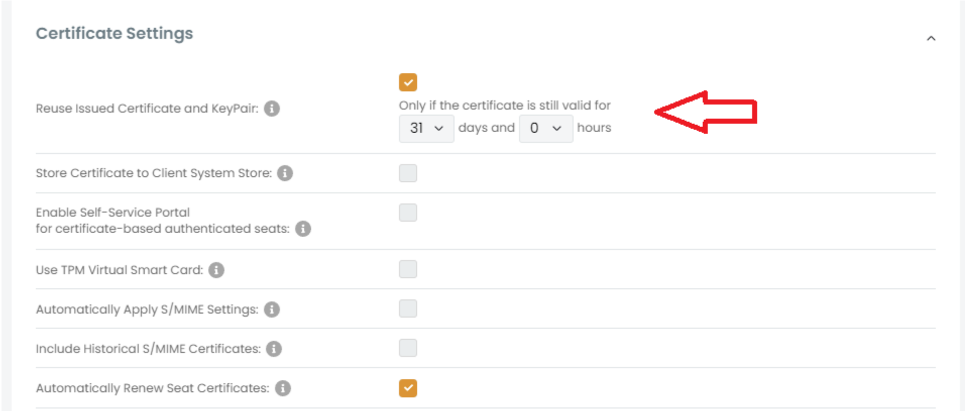
Time Before Certificate Expires:) hours, 6 minutes, 0 seconds □ allow overwrite

Use client OS logon type: Use Username

Should a KeyTalk agent be triggered to renew the certificate and for whatever reason fail its certificate request, an email is sent to the predefined admin email address provided this feature was configured during the client configuration.

14.2 Replacement based on duration in absolute time (the default, and alternative to validity %)

This feature has been updated as of KeyTalk agent 6.4.5 and KeyTalk firmware 6.4.2 with the ability to automatically fetch the renewal threshold from the Certificate TEMPLATE:



Certificate Settings

Reuse Issued Certificate and KeyPair: ☑ Only if the certificate is still valid for 31 days and 0 hours

Store Certificate to Client System Store: □

Enable Self-Service Portal for certificate-based authenticated seats: □

Use TPM Virtual Smart Card: □

Automatically Apply S/MIME Settings: □

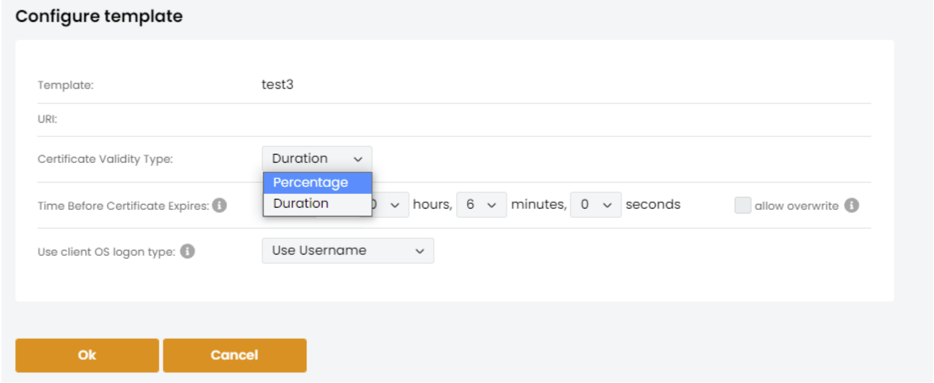
Include Historical S/MIME Certificates: □

Automatically Renew Seat Certificates: ☑

The certificate validity replacement in absolute time is defined when creating the RCCD (the KeyTalk client configuration file).
It reflects the time the renewal script must request a new certificate, based on the remaining actual time of the end date/time of the certificate.

Say you set validity replacement duration to 1 hour, then the certificate will be replaced 1 hour before it expired

The certificate validity duration can be set under TEMPLATES -> CREATE RCCD -> EDIT



Configure template

Template: test3

URI:

Certificate Validity Type: Duration

Time Before Certificate Expires: 0 hours, 6 minutes, 0 seconds □ allow overwrite

Use client OS logon type: Use Username

Should a KeyTalk agent be triggered to renew the certificate and for whatever reason fail its certificate request, an email is sent to the predefined admin email address provided this feature was configured during the client configuration.

14.3 KeyTalk client configuration file creation and advanced settings (Citrix/RD/Multi-User)

Each KeyTalk client/app/REST_API requires several minimal settings in order to be able to operate. These details are contained within the KeyTalk Real Client Configuration Data file, ie RCCD.

To generate an RCCD that contains at least 1, but possibly multiple configurations to fetch certificates for an end-point from multiple KeyTalk certificate template TEMPLATE configurations, simply checkmark each TEMPLATE you want included, provide the KeyTalk cluster name, ensure the url to the KeyTalk cluster is correct, and optionally upload a logo in PNG format sized 110x110 pixels.

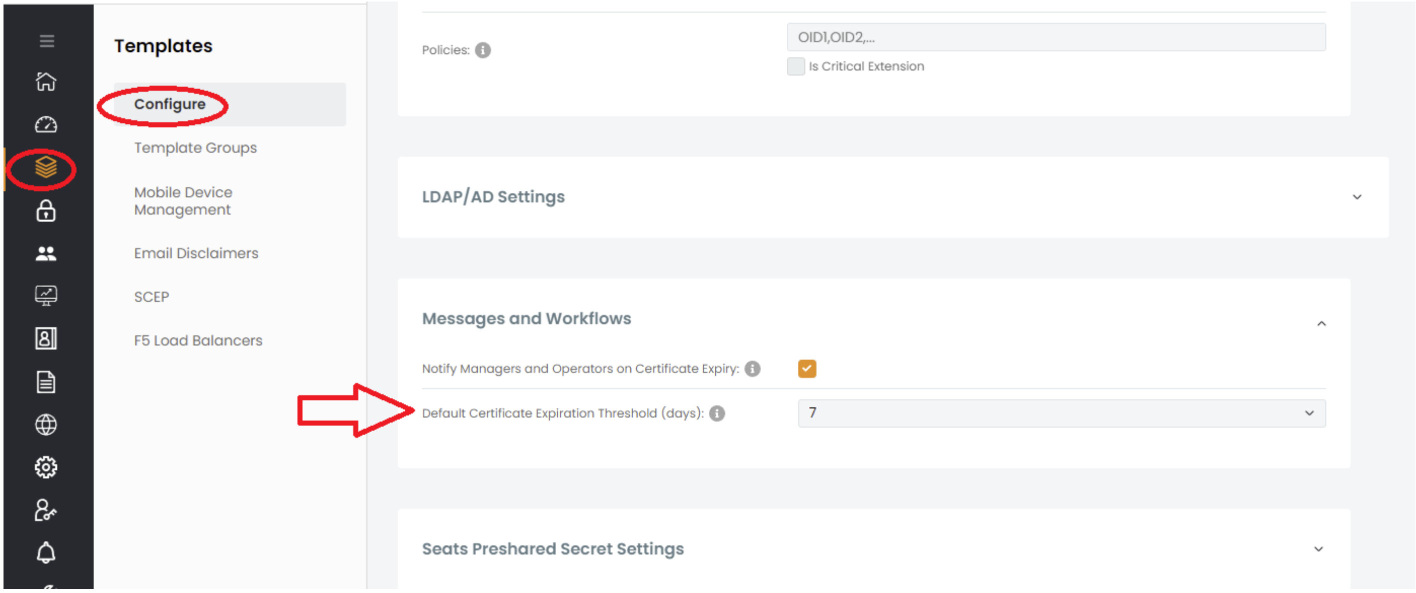
Before generating the RCCD file you can additionally edit the advanced settings.
The advanced settings allow you to enable/disable “allow overwrite” settings.
Disallowing at least 1 overwrite settings, triggers KeyTalk to include a MASTER config file which on windows machines is installed in the Windows public user directory.

Should a KeyTalk config file ever be missing, get corrupted, or contain values that minimally do not match with the MASTER config file, then this MASTER file will be fetched by the KeyTalk client to reset the configuration it minimally needs to operate.

Most customers who run a Citrix or Remote Desktop or simply a Windows multi user environment use this feature to ensure an installed KeyTalk client always has a proper config file for every user in the system.

14.4 Warning when failing to renew automatically

Each TEMPLATE contains a warning value used to warn the certificate KeyTalk OWNER and the Certificate end-point email address in case the certificate did not renew before the warning threshold was met:



Policies: OID1,OID2,...

□ Is Critical Extension

Notify Managers and Operators on Certificate Expiry: ☑

Default Certificate Expiration Threshold (days): 7

15.End-point user/device administration and hardware recognition

The KeyTalk solution will keep track of every attempt to request a certificate in the logfiles, and it also represents this information in its SEATS database on a TEMPLATE level.

When defining the TEMPLATE, the System Admin, or once the template has been created also the TEMPLATE Operator, can state that device end-points must be tracked per user account which successfully request a certificate under that TEMPLATE.

In KeyTalk currently a maximum of 10 device slots can be assigned to a single seat.
A seat uses 1 license in KeyTalk, and is defined as a unique positively authenticated username (a certificate Common Name) or anchorpoint, under a given TEMPLATE.

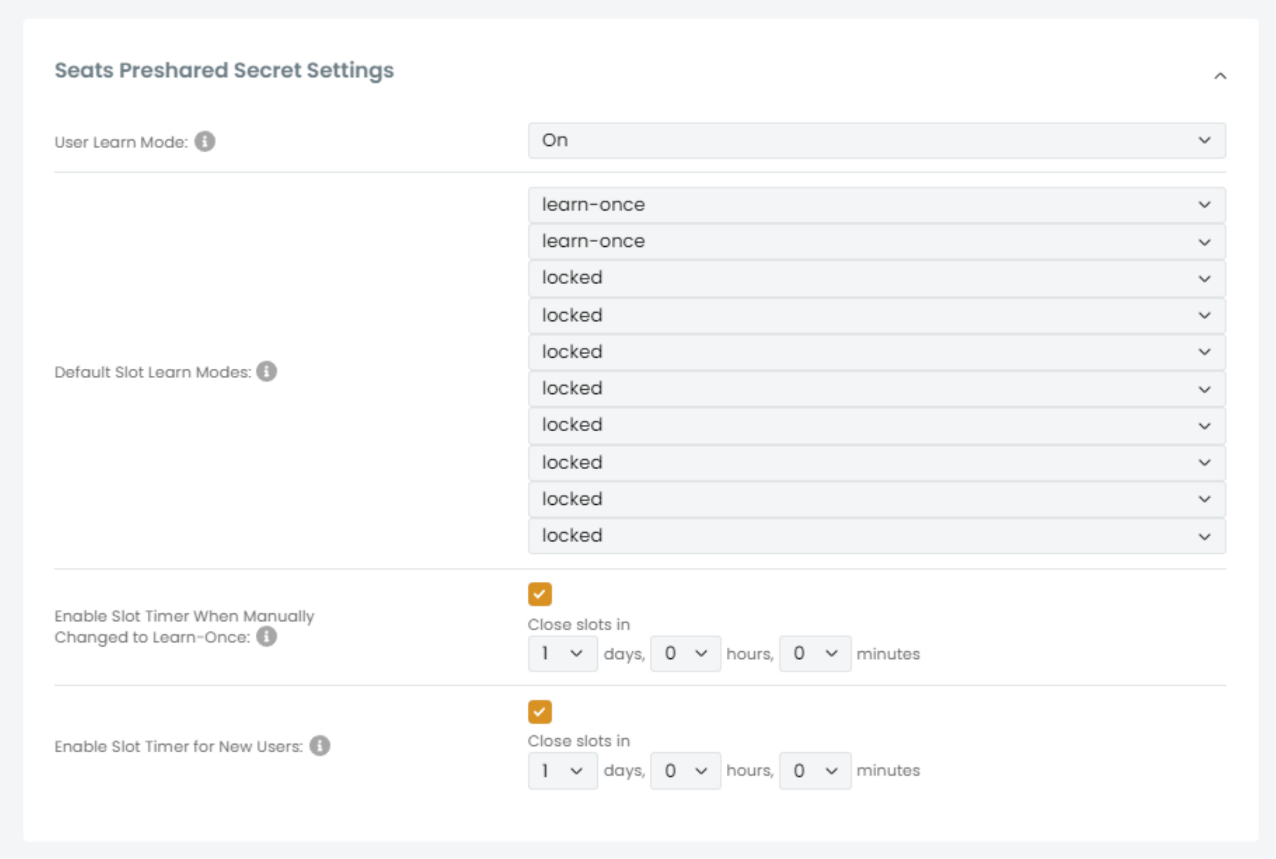
So, a seat in KeyTalk does not need to be a human, it can be a server or an IoT device as well. However typically a server or an IoT device will be assigned a single device slot in the TEMPLATE configuration, where-as a human can be assigned up to 10 device slots (in most cases 10 devices) without expending more than 1 KeyTalk license.

A KeyTalk TEMPLATE can be configured to automatically learn new seats (TEMPLATE learn mode=on), or enforce only manual new entries as trusted user/device combinations (TEMPLATE learn mode=off).

In learn mode=on, a username, or its equivalent anchorpoint, which doesn’t exist yet in the SEATS administration for a given TEMPLATE will be automatically added provided the authentication of that user was considered positive by the configured IDP (ie in KeyTalk the Authentication module connected to the TEMPLATE).

Once learned, a seat is assigned the device slots as configured in the TEMPLATE.

In the below example, the TEMPLATE assigns 2 LEARN-ONCE device slots to a newly learned seat, and disallows further devices to be learned (ie locked)



Seats Preshared Secret Settings

User Learn Mode: On

Default Slot Learn Modes:
- learn-once
- learn-once
- locked
- locked
- locked
- locked
- locked
- locked
- locked
- locked

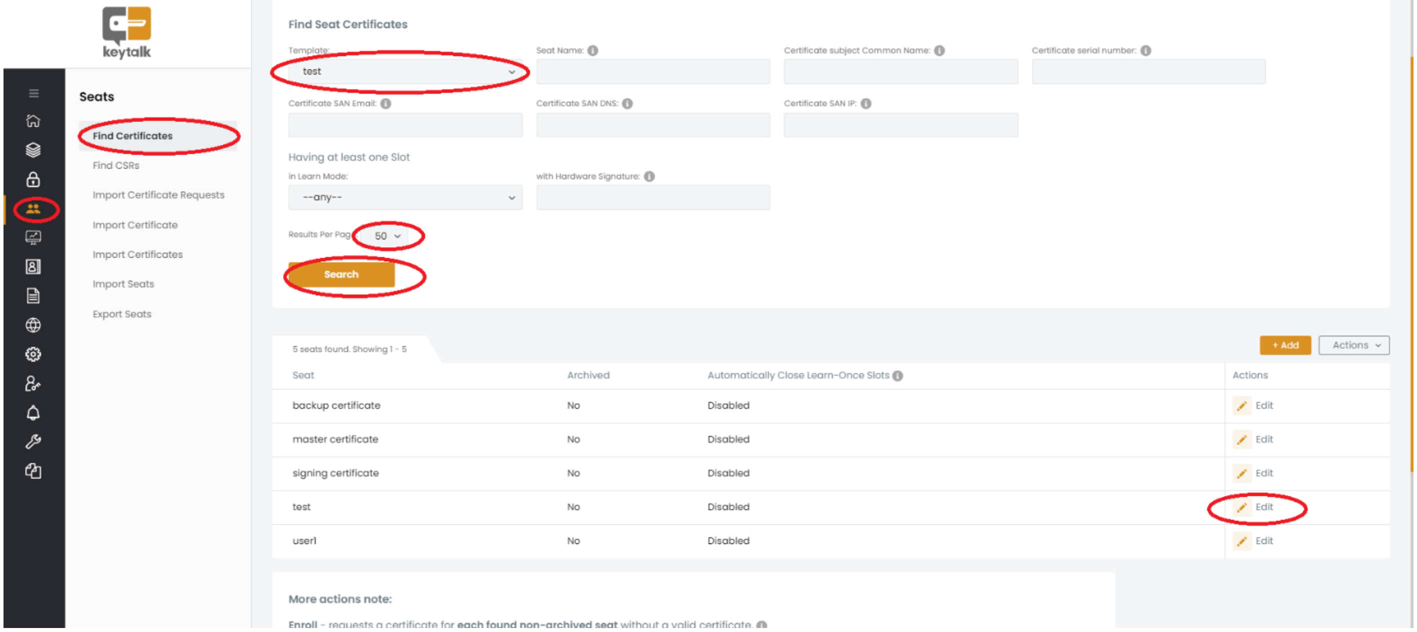
Enable Slot Timer When Manually Changed to Learn-Once: ☑ Close slots in 1 days, 0 hours, 0 minutes

Enable Slot Timer for New Users: ☑ Close slots in 1 days, 0 hours, 0 minutes

Furthermore the 2 configured timers allow for:

- an additional 1 hour for the seat to register his second device from the moment he first authenticated positively from his first device. When the second slot is not used within this 1 hour it automatically locks, preventing possible abuse.
- an automated 15-minute timed window of opportunity to register another device when someone (for example a TEMPLATE) manually sets one of the slots to learn-once.

When doing a seat lookup for a particular TEMPLATE under SEATS, certificate issuance specifics can be viewed, such as when was a device registered, or when it last authenticated positively and obtain as certificate.
But also revocation and pre-enrolments can be done.



Find Seat Certificates

Template: test
Seat Name:
Certificate subject Common Name:
Certificate serial number:
Certificate SAN Email:
Certificate SAN DNS:
Certificate SAN IP:

Having at least one Slot
in Learn Mode: --any--
with Hardware Signature:

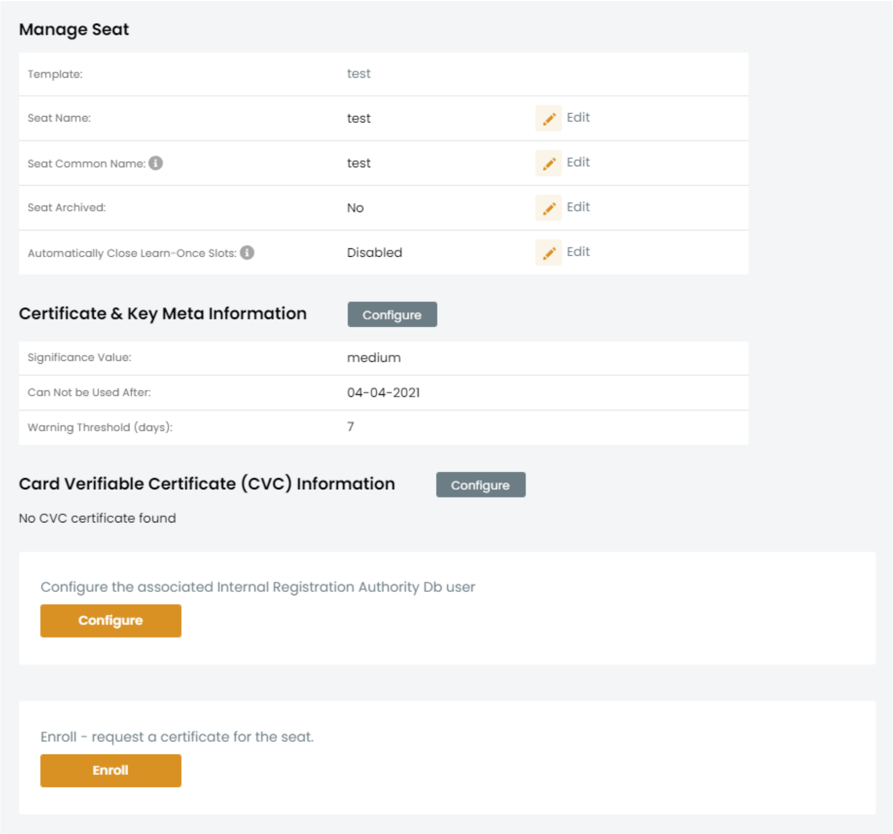
Results Per Pag 50

5 seats found. Showing 1 - 5

Seat	Archived	Automatically Close Learn-Once Slots	Actions
backup certificate	No	Disabled	Edit
master certificate	No	Disabled	Edit
signing certificate	No	Disabled	Edit
test	No	Disabled	Edit
user1	No	Disabled	Edit

More actions note:

Enroll - requests a certificate for **each found non-archived seat** without a valid certificate.



Manage Seat

Template:	test	
Seat Name:	test	Edit
Seat Common Name: ⓘ	test	Edit
Seat Archived:	No	Edit
Automatically Close Learn-Once Slots: ⓘ	Disabled	Edit

Certificate & Key Meta Information Configure

Significance Value:	medium
Can Not be Used After:	04-04-2021
Warning Threshold (days):	7

Card Verifiable Certificate (CVC) Information Configure

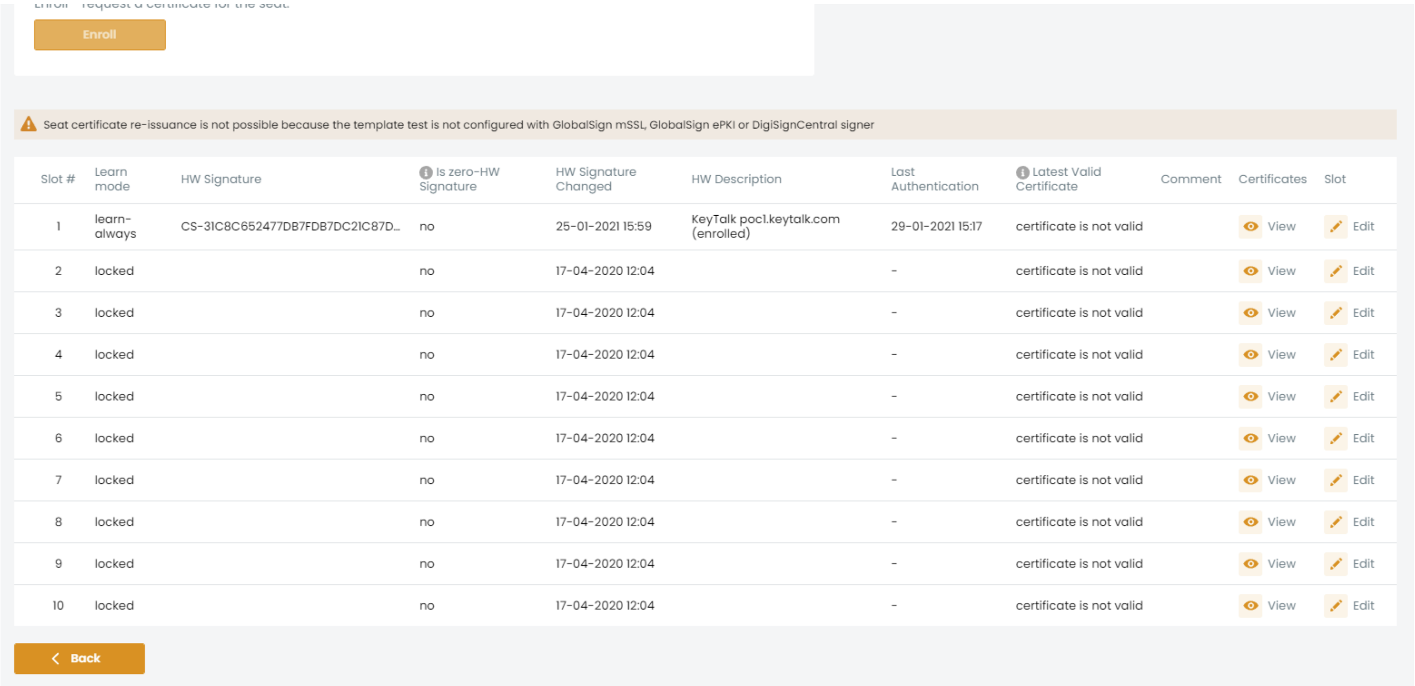
No CVC certificate found

Configure the associated Internal Registration Authority Db user

Configure

Enroll - request a certificate for the seat.

Enroll



Enroll - request a certificate for the seat.

Enroll

⚠ Seat certificate re-issuance is not possible because the template test is not configured with GlobalSign mSSL, GlobalSign ePKI or DigiSignCentral signer

Slot #	Learn mode	HW Signature	ⓘ Is zero-HW Signature	HW Signature Changed	HW Description	Last Authentication	ⓘ Latest Valid Certificate	Comment	Certificates	Slot
1	learn-always	CS-31C8C652477DB7FDB7DC21C87D...	no	25-01-2021 15:59	KeyTalk poc1.keytalk.com (enrolled)	29-01-2021 15:17	certificate is not valid		View	Edit
2	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
3	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
4	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
5	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
6	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
7	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
8	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
9	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
10	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit

< Back

Note that the HWSignature that is shown for each device slot is a SHA2 of the total, in order of request, device components as defined in the relevant TEMPLATE (provided a KeyTalk agent was used) or it's something generated by the software a third party built using the KeyTalk REST API.

With KeyTalk learning what devices belong to a user, the SEAT HardWare Signatures can optionally (arguably most customers use it) be used to enforce an additional authentication factor when requesting a certificate.

16. Certificate management : SEATS

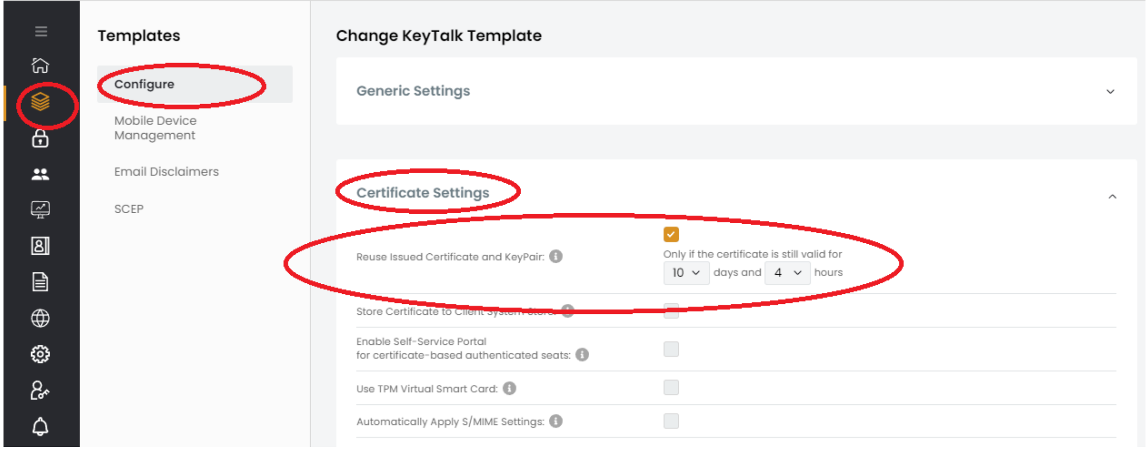
The SEATS administration keeps track on a TEMPLATE level of any certificate and corresponding primary certificate details per endpoint (user)name and unique certificate CommonName (CN) per assigned trusted device slot.

As a result, the SysAdmin, Cluster Admin, TEMPLATE Manager, TEMPLATE Operator, and Certificate OWNER, can lookup (in)active and archived certificates, as issued through KeyTalk based on:

1) Registration Authority based username
2) Certificate CommonName (CN)
3) Certificate serialnumber

16.1 Certificate and key roll-over, encrypting stored private keys

Each TEMPLATE allows for the option: cert+key reuse (certificate and key roll-over)
When activated, the DEVID USER administration will additionally store on a user level the keypair belonging to an issued certificate.



Templates

- Configure
- Mobile Device Management
- Email Disclaimers
- SCEP

Change KeyTalk Template

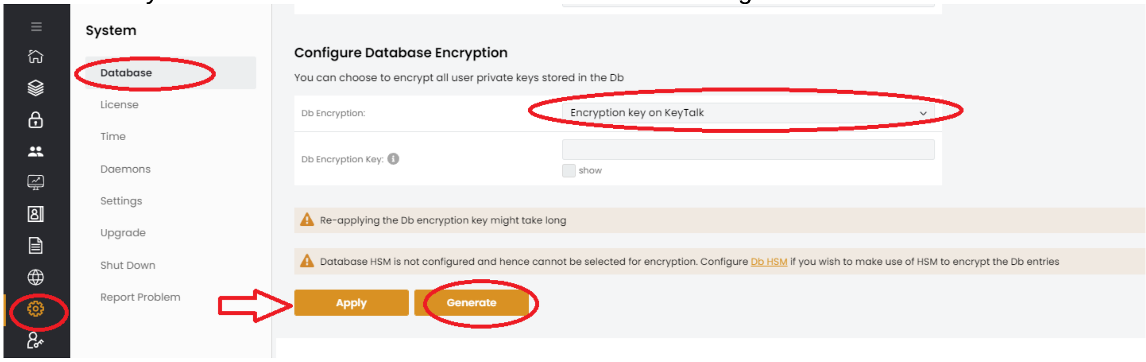
Generic Settings

Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ Only if the certificate is still valid for 10 days and 4 hours
Store Certificate to Client System Store:	□
Enable Self-Service Portal for certificate-based authenticated seats:	□
Use TPM Virtual Smart Card:	□
Automatically Apply S/MIME Settings:	□

Certificates are public, hence not stored in encrypted format, however KeyTalk allows for an AES 256 key to encrypt stored private keys of certificates on its MySQL Db.
This AES key is NOT shared through the MySQL database, and must be set individually per KeyTalk instance!!

The AES key can be set under SYSTEM -> Database -> Configure:



System

- Database
- License
- Time
- Daemons
- Settings
- Upgrade
- Shut Down
- Report Problem

Configure Database Encryption

You can choose to encrypt all user private keys stored in the Db

Db Encryption:	Encryption key on KeyTalk
Db Encryption Key:	□ show

Re-applying the Db encryption key might take long

Database HSM is not configured and hence cannot be selected for encryption. Configure Db HSM if you wish to make use of HSM to encrypt the Db entries

When a Database HSM has been configured, the option to use the HSM is shown as well.

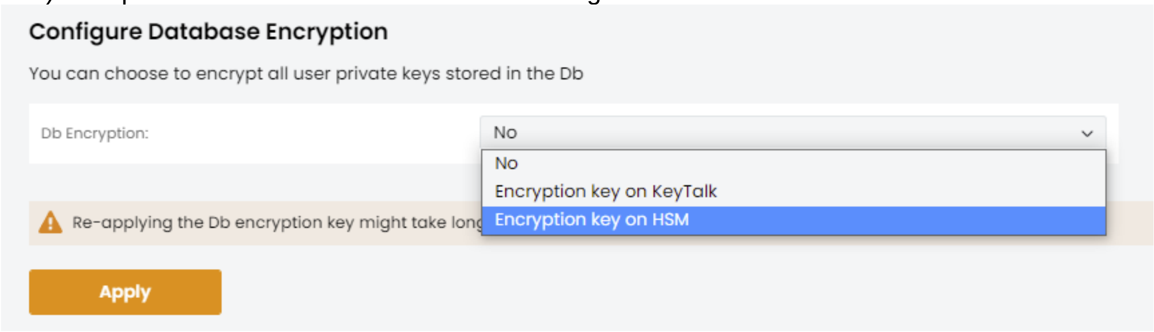
AES-256 CBC HMAC-SHA256 is used. Its main advantage is that it is widely supported across different versions of MySQL.

As a result of certificate and key roll-over, any allowed device slot requesting a certificate under the same TEMPLATE, will be given the same certificate and keypair as previously issued. Depending on the client/app requesting the certificate, KeyTalk will issue an appropriately converted certificate usable for the applicable end-point OS.

Make sure you do not loose this AES key or you will not be able to decrypt your sensitive data!!

16.1.1 HSM and MySQL Db encryption key

If an HSM has been setup to be used to store the AES256 MySQL Db encryption key (See chapter 11.1) an option is offered to make use of the configured HSM.

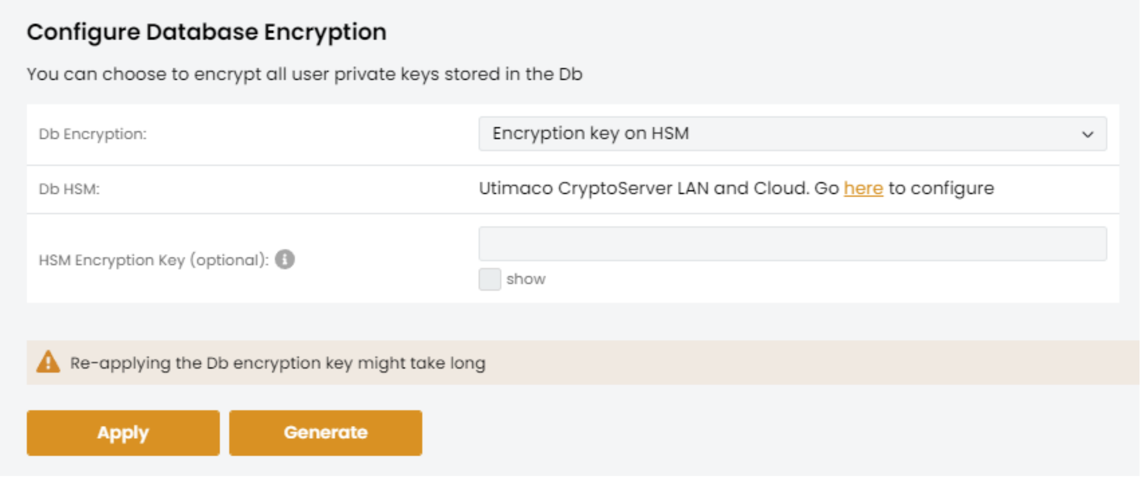


Configure Database Encryption

You can choose to encrypt all user private keys stored in the Db

Db Encryption: No

No
Encryption key on KeyTalk
Encryption key on HSM
Re-applying the Db encryption key might take long
Apply



Configure Database Encryption

You can choose to encrypt all user private keys stored in the Db

Db Encryption: Encryption key on HSM

Db HSM: Utimaco CryptoServer LAN and Cloud. Go here to configure

HSM Encryption Key (optional):

□ show

Re-applying the Db encryption key might take long

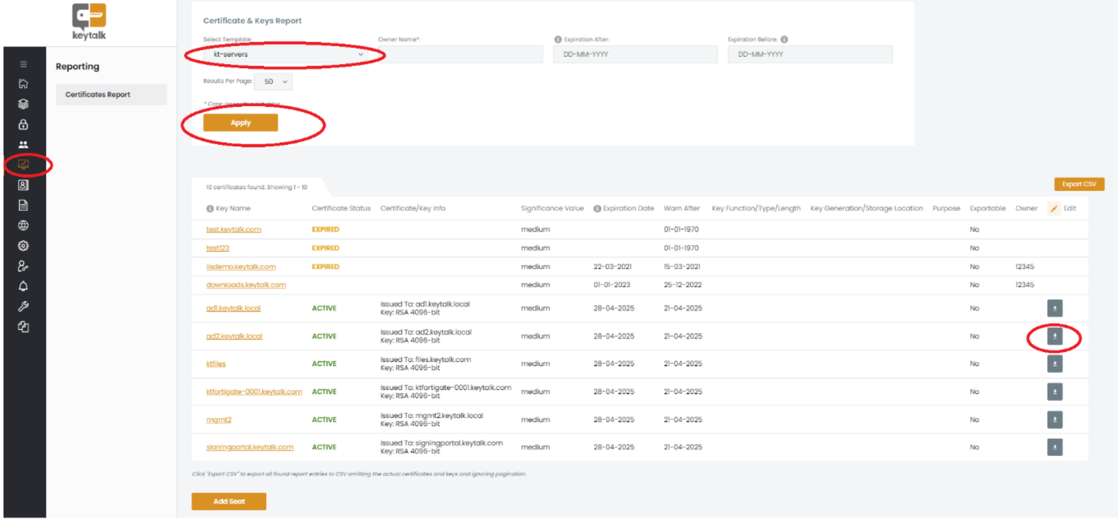
Two options now become available:

1) Leave the encryption key blank and select apply. This will result in the AES key being generated and getting stored on the HSM. Not even the KeyTalk Sys Admin will know the key.
2) Enter your own 64 byte (128 hex characters) AES key, or have KeyTalk generate it visibly for you. Now you can store the key for recovery purposes in an airgapped/offline location. When pressing apply KeyTalk will send the key to the HSM and remove it from its systems.

16.2 Historic certificate lookup

The SEATS administration will keep track all historic issued certificates and keys.

To view and/or download historic certificates belonging to an end-point, either go to: REPORTING And select the SEAT you wish the certificate and key to be downloaded for.



Certificate & Keys Report

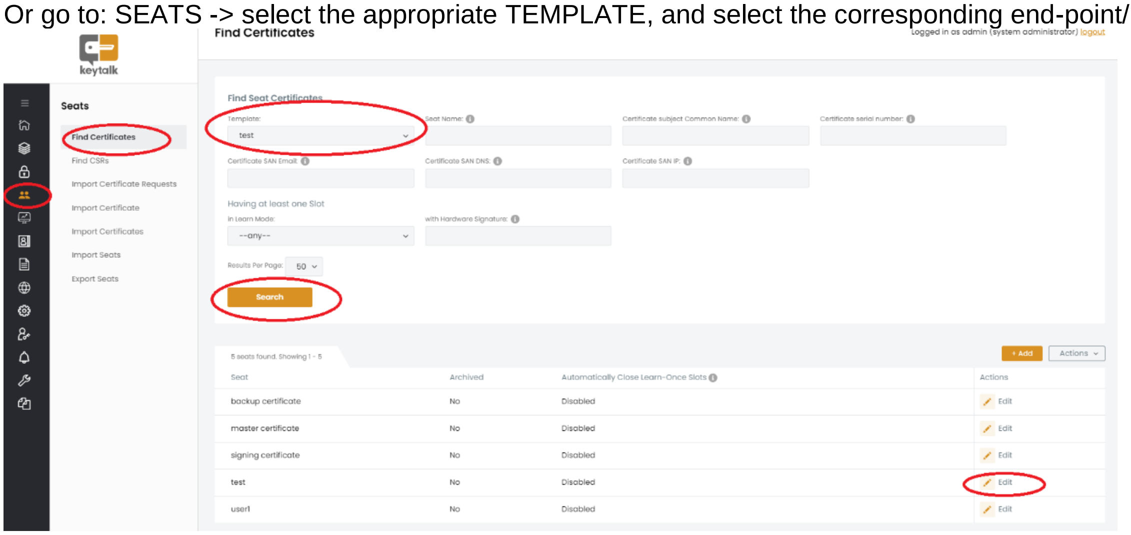
Select Template: kt-servers
Owner Name*:
Expiration After: DD-MM-YYYY
Expiration Before: DD-MM-YYYY
Results Per Page: 50

10 certificates found. Showing 1 - 10

Key Name	Certificate Status	Certificate/Key Info	Significance Value	Expiration Date	Warn After	Key Function/Type/Length	Key Generation/Storage Location	Purpose	Exportable	Owner	Edit
test.keytalk.com	EXPIRED		medium		01-01-1970				No		
test123	EXPIRED		medium		01-01-1970				No		
iisdemo.keytalk.com	EXPIRED		medium	22-03-2021	15-03-2021				No	12345	
downloads.keytalk.com			medium	01-01-2023	25-12-2022				No	12345	
ad1.keytalk.local	ACTIVE	Issued To: ad1.keytalk.local Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ad2.keytalk.local	ACTIVE	Issued To: ad2.keytalk.local Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ktfiles	ACTIVE	Issued To: files.keytalk.com Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ktfortigate-0001.keytalk.com	ACTIVE	Issued To: ktfortigate-0001.keytalk.com Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
mgmt2	ACTIVE	Issued To: mgmt2.keytalk.local Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
signingportal.keytalk.com	ACTIVE	Issued To: signingportal.keytalk.com Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇

Click "Export CSV" to export all found report entries to CSV omitting the actual certificates and keys and ignoring pagination.

Or go to: SEATS -> select the appropriate TEMPLATE, and select the corresponding end-point/user:



Find Certificates

Logged in as admin (system administrator) logout

Find Seat Certificates

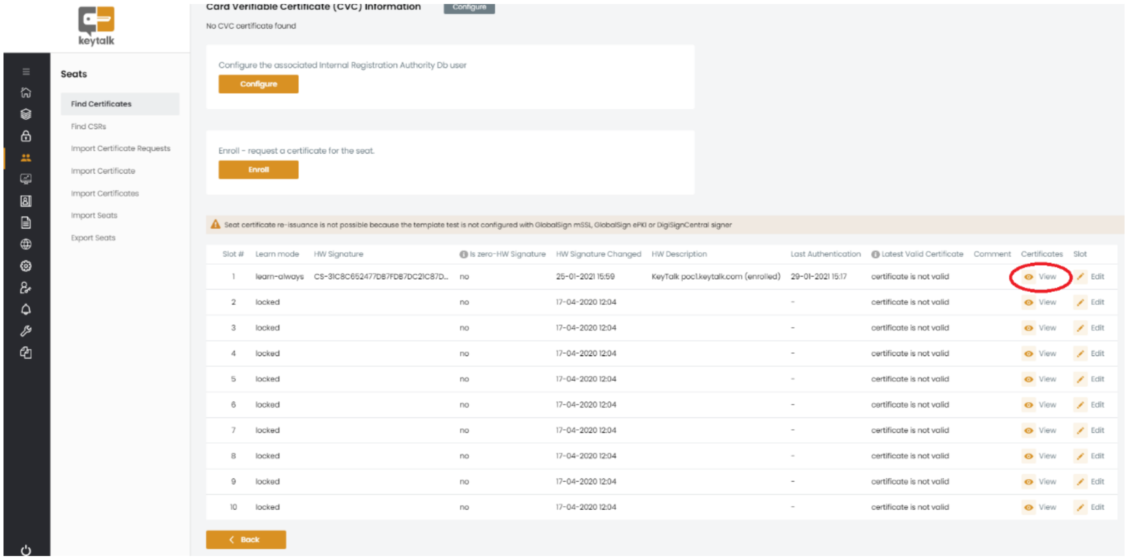
Template: test
Seat Name:
Certificate subject Common Name:
Certificate serial number:
Certificate SAN Email:
Certificate SAN DNS:
Certificate SAN IP:

Having at least one Slot

In Learn Mode: --any--
with Hardware Signature:
Results Per Page: 50

5 seats found. Showing 1 - 5

Seat	Archived	Automatically Close Learn-Once Slots	Actions
backup certificate	No	Disabled	Edit
master certificate	No	Disabled	Edit
signing certificate	No	Disabled	Edit
test	No	Disabled	Edit
user1	No	Disabled	Edit



Card Verifiable Certificate (CVC) Information

No CVC certificate found

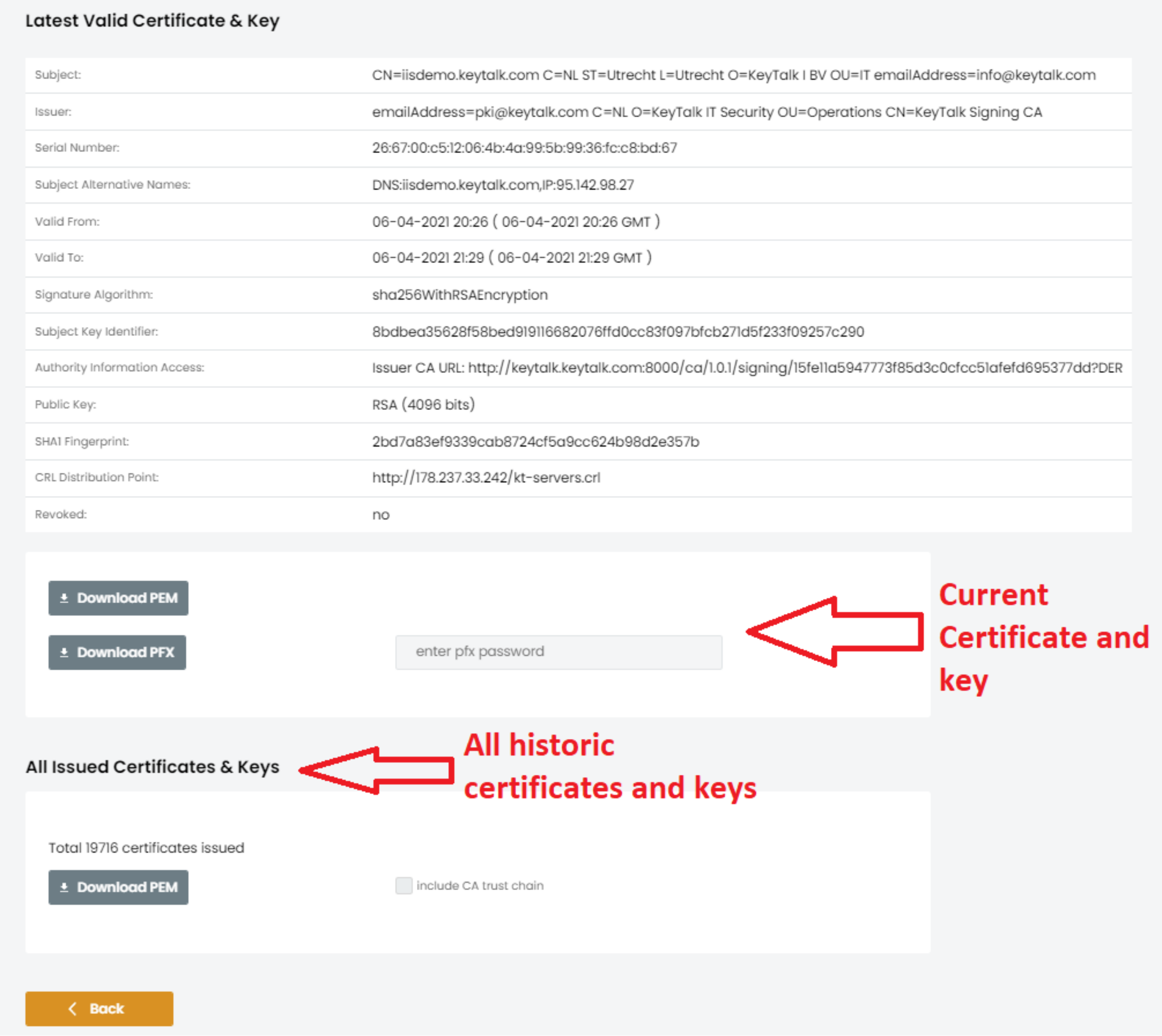
Configure the associated Internal Registration Authority Db user

Enroll - request a certificate for the seat.

⚠ Seat certificate re-issuance is not possible because the template test is not configured with GlobalSign mSSL, GlobalSign ePKI or DigiSignCentral signer

Slot #	Learn mode	HW Signature	Is zero-HW Signature	HW Signature Changed	HW Description	Last Authentication	Latest Valid Certificate	Comment	Certificates	Slot
1	learn-always	CS-31C8C652477DB7FDB7DC21C87D...	no	25-01-2021 15:59	KeyTalk poc1.keytalk.com (enrolled)	29-01-2021 15:17	certificate is not valid		View	Edit
2	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
3	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
4	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
5	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
6	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
7	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
8	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
9	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
10	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit

Now select either download all issued certificates (and keys), or select Action to view and download individual historic certificates (and keys):



Latest Valid Certificate & Key

Subject:	CN=iisdemo.keytalk.com C=NL ST=Utrecht L=Utrecht O=KeyTalk I BV OU=IT emailAddress=info@keytalk.com
Issuer:	emailAddress=pki@keytalk.com C=NL O=KeyTalk IT Security OU=Operations CN=KeyTalk Signing CA
Serial Number:	26:67:00:c5:12:06:4b:4a:99:5b:99:36:fc:c8:bd:67
Subject Alternative Names:	DNS:iisdemo.keytalk.com,IP:95.142.98.27
Valid From:	06-04-2021 20:26 (06-04-2021 20:26 GMT)
Valid To:	06-04-2021 21:29 (06-04-2021 21:29 GMT)
Signature Algorithm:	sha256WithRSAEncryption
Subject Key Identifier:	8bdbea35628f58bed919116682076ffd0cc83f097bfcb271d5f233f09257c290
Authority Information Access:	Issuer CA URL: http://keytalk.keytalk.com:8000/ca/1.0.1/signing/15fe11a5947773f85d3c0cfcc51afefd695377dd?DER
Public Key:	RSA (4096 bits)
SHA1 Fingerprint:	2bd7a83ef9339cab8724cf5a9cc624b98d2e357b
CRL Distribution Point:	http://178.237.33.242/kt-servers.crl
Revoked:	no

Current Certificate and key

All Issued Certificates & Keys

All historic certificates and keys

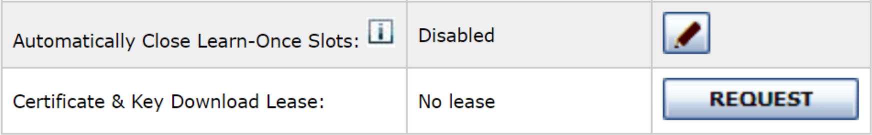
Total 19716 certificates issued

16.3 4-eye download principle for private keys

End-points/seats may only obtain certificates and keys when positively authenticated through KeyTalk and as approved by the set Registration Authority.

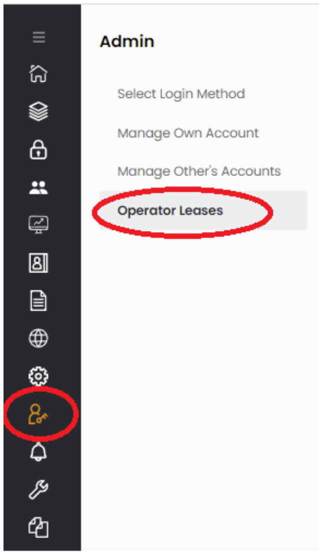
However, for administrative purposes, a support employee (TEMPLATE Operator or Certificate OWNER), might need access to the certificate and keypair for corporate reasons, and thus require a download of both the certificate and key-pair in an appropriate format.

From the SEATS administration panel, a TEMPLATE Operator or OWNER can request a LEASE APPROVAL to download the certificate and keys of a SEAT for a specific period of time.



Automatically Close Learn-Once Slots: ⓘ	Disabled	✎
Certificate & Key Download Lease:	No lease	REQUEST

The download request will result in a request showing up for an appropriate TEMPLATE ADMIN and CLUSTER ADMIN, and SYS ADMIN where he or she can see who requested the download, the reason why and a decline or approve button.
These requests for approval are found under : Admin -> TEMPLATE Operator Leases



When approved, the TEMPLATE Operator is notified that his/her download request was approved or declined. When approved the TEMPLATE Operator has a timed window of opportunity to download the certificate and key.

TEMPLATE ADMINs and SYS/CLUSTER ADMINs can download certificates and keys without 4-eye principle as they are in the top of the authorization hierarchy.

NOTE: When a client generates the Certificate Signing Request offline (ie not on the KeyTalk CKMS), , the private key is NOT shared with the KeyTalk virtual appliance, thus resulting in the inability for any management role to download the certificate and key-pair.

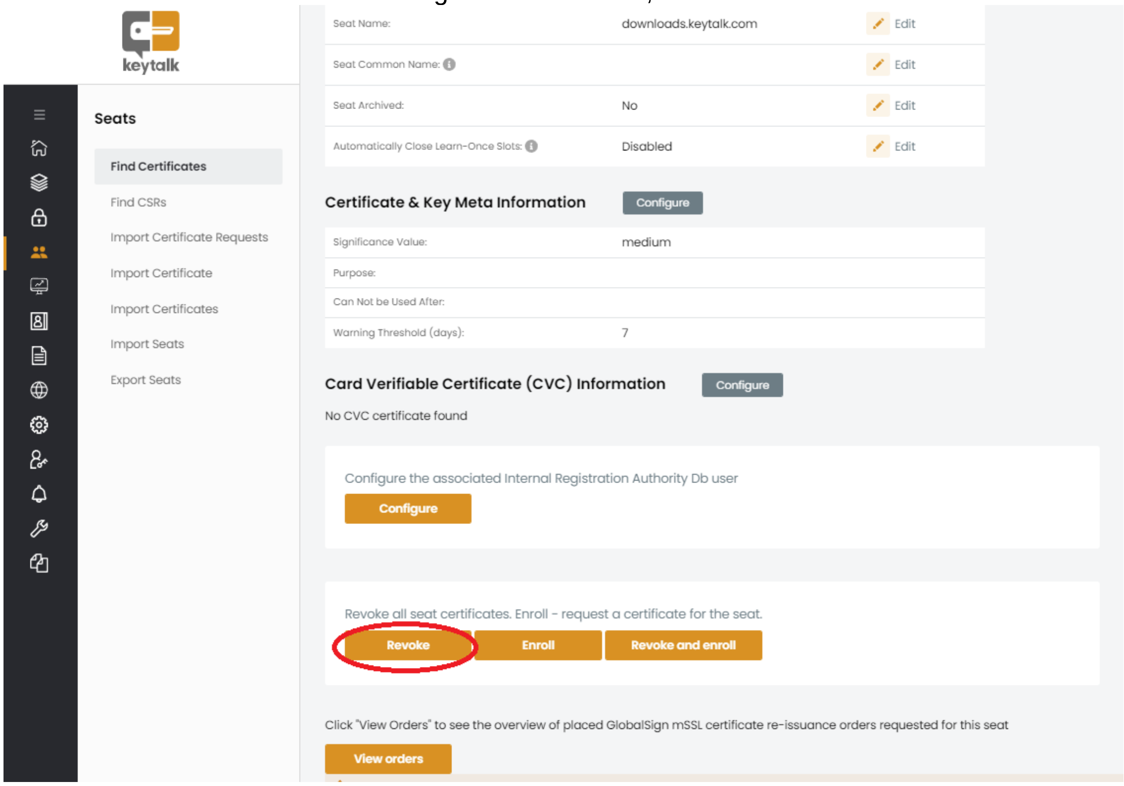
16.4 Revocation of certificates

Many customers use KeyTalk's private CA to issue short lived certificates to end-points for authentication purposes.
According to KeyTalk: short lived certificates have a validity of 12 hours or less.

Since CRLs are typically updated once every 12-24 hours, this means that a Network Admin does not need to propagate the CRL.
Some will claim that OCSP would make sense in these cases, but practically it takes on average more than 24 hours before a certificate is reported as needing to be revoked, hence making the intended use of OCSP for short lived certificates obsolete.

The KeyTalk private CA does of course support long lived certificates of nearly any validity period.

To revoke for an individual seat of a given TEMPLATE, select the seat and select REVOKE



Seat Name:	downloads.keytalk.com	Edit
Seat Common Name:		Edit
Seat Archived:	No	Edit
Automatically Close Learn-Once Slots:	Disabled	Edit

Certificate & Key Meta Information

Significance Value:	medium
Purpose:	
Can Not be Used After:	
Warning Threshold (days):	7

Card Verifiable Certificate (CVC) Information

No CVC certificate found

Configure the associated Internal Registration Authority Db user

Revoke all seat certificates. Enroll - request a certificate for the seat.

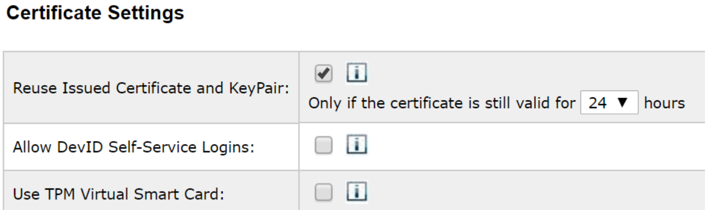
Click "View Orders" to see the overview of placed GlobalSign mSSL certificate re-issuance orders requested for this seat

Alternatively REVOKE AND ENROLL can be chosen which results in the user's revoked certificate immediately being replaced.
This function is only available when REUSE CERTIFICATE is enabled in the TEMPLATE.

16.5 Certificate pre-enrolment / enrolment

When issuing for example S/MIME certificates, there will likely be a need to have the required certificates already enrolled so the Admin can prepare his systems, such as an LDAP or AD address book directory.

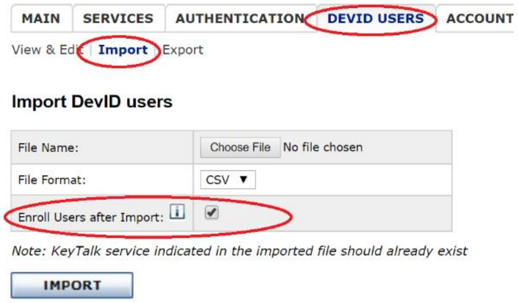
Pre-enrolment and enrolment are only supported in KeyTalk when in the TEMPLATES certificate template REUSE CERTIFICATE is enabled



Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ ⓘ Only if the certificate is still valid for 24 ▼ hours
Allow DevID Self-Service Logins:	☐ ⓘ
Use TPM Virtual Smart Card:	☐ ⓘ

To pre-enroll for X amount of end-points/users, ensure you have a CSC or LDIF or XML file with the required information and upload it into KeyTalk under DEVID USERS -> import



Import DevID users

File Name:	Choose File No file chosen
File Format:	CSV ▼
Enroll Users after Import: ⓘ	☑

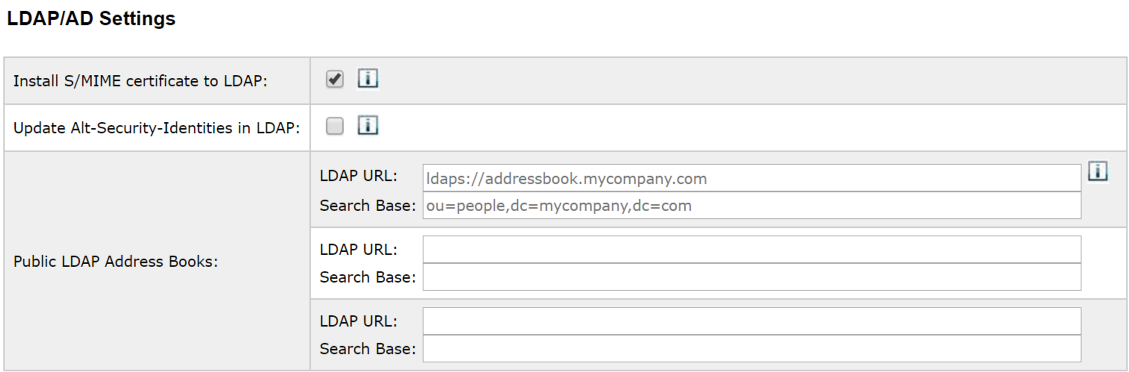
Note: KeyTalk service indicated in the imported file should already exist

IMPORTANT: Importing a SEAT set for a given TEMPLATE-group results in previous SEAT data for that TEMPLATE being overwritten.
AD Attribute mapping to certificate field with pre-enrolment issuance of certificates **requires** an AD Service Account to be configured in the connected KeyTalk LDAP Registration Authority module!

16.6 Certificate propagation: S/MIME email address book directory

KeyTalk makes use of LDAP based BIND to sends certificate updates to a target AD/LDAP and/or an additional optional AD/LDAP source that can act as a public S/MIME directory.
This allows any party using their AD, or 3rd party who simply needs access to appropriate certificate details for S/MIME purposes, to add the certificate to its private and/or public repository as an address-book url. Typically, an Active directory is used, and/or an LDAP.

To publish S/MIME details of an X.509 certificate to a target AD or LDAP, activate this feature on the TEMPLATE level:



LDAP/AD Settings

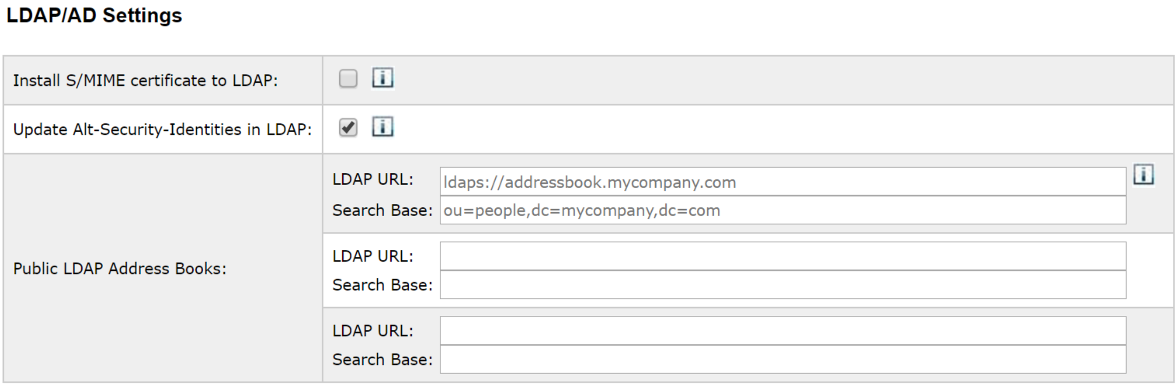
Install S/MIME certificate to LDAP:	☑	
Update Alt-Security-Identities in LDAP:	☐	
Public LDAP Address Books:	LDAP URL: Search Base:	ldaps://addressbook.mycompany.com ou=people,dc=mycompany,dc=com
	LDAP URL: Search Base:	
	LDAP URL: Search Base:	

As a result, the target configured AD or LDAP will be provided with the relevant S/MIME details in its appropriate native format, whereby attribute "userCertificate" is used

To **additionally** write the S/MIME details to a (semi)public LDAP or AD based address-book, ensure that this LDAP/AD is configured under the LDAP/AD Authentication module as an (additional) LDAP server, and ensure you checkmark Address Book Only.

16.7 Certificate propagation: AD AltSecurityIdentities

Microsoft solutions allow for Single-Sign-On based on client certificates.
This requires the Active Directory on a user account level to be mapped to the issued user-certificate using the attribute AltSecurityIdentities.
KeyTalk can write the issued certificate to this attribute when this function is activated under the applicable TEMPLATE and provided the TEMPLATE is connected to an LDAP/AD authentication module, AND provided that a TEMPLATE User is defined in the LDAP/AD authentication module connector.



LDAP/AD Settings

Install S/MIME certificate to LDAP:	☐	
Update Alt-Security-Identities in LDAP:	☑	
Public LDAP Address Books:	LDAP URL: Search Base:	ldaps://addressbook.mycompany.com ou=people,dc=mycompany,dc=com
	LDAP URL: Search Base:	
	LDAP URL: Search Base:	

Note: When an updated certificate is issued the previous entry in the AD is updated only, no new entry is added.

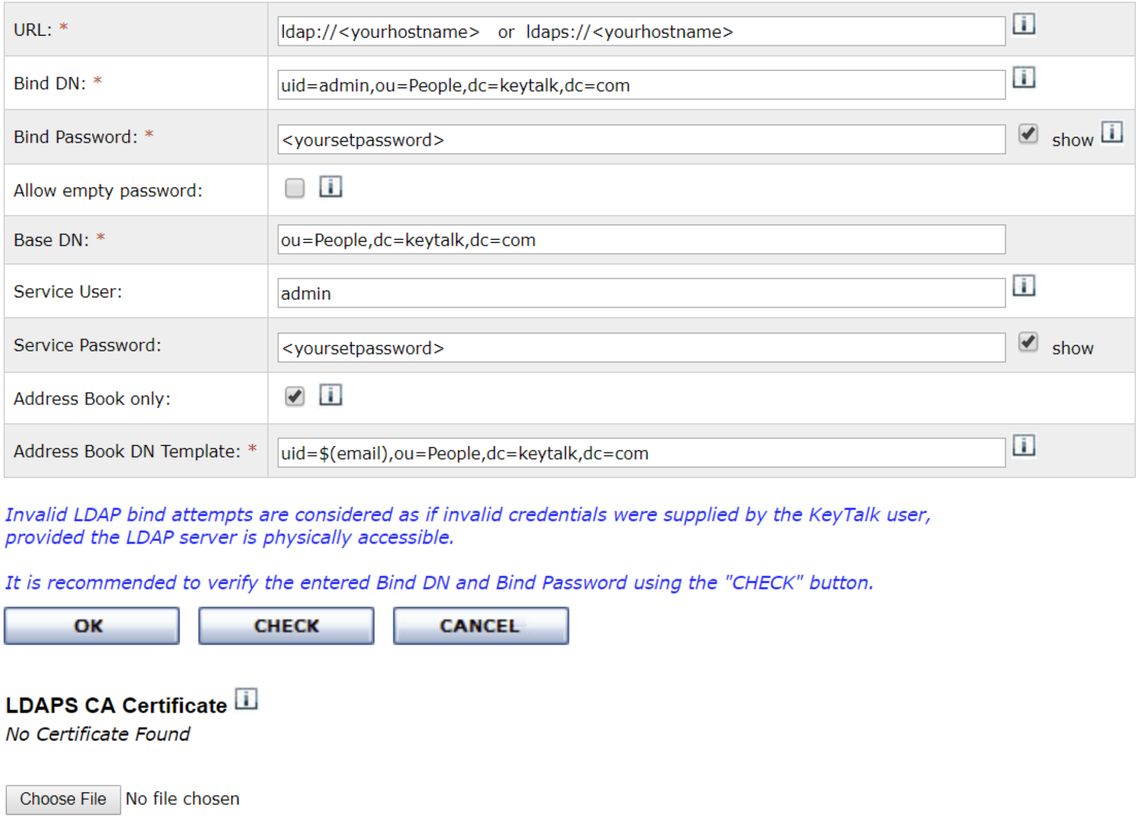
Alternatively for Admins who don't want KeyTalk to write to their AD, they can use a Powershell script to update their AD under their own manual control, as found under https://keytalk.com/support

16.8 KeyTalk’s LDAP S/MIME secure email address book directory

Many customer’s do not wish to fiddle with their own LDAP or AD to make available their own secure public address book for S/MIME detail lookup purposes.

KeyTalk offers as part of its certificate and key management solution a hardened (Open)LDAP for this sole purpose.
This LDAP comes as a virtual appliance and allows for regular LDAP based address book lookups in commonly used mail clients, but also includes a browser-based lookup function.

The following settings apply under KeyTalk’s Authentication LDAP module



URL: *	ldap://<yourhostname> or ldaps://<yourhostname>
Bind DN: *	uid=admin,ou=People,dc=keytalk,dc=com
Bind Password: *	<yoursetpassword> ☑ show
Allow empty password:	☐
Base DN: *	ou=People,dc=keytalk,dc=com
Service User:	admin
Service Password:	<yoursetpassword> ☑ show
Address Book only:	☑
Address Book DN Template: *	uid=$(email),ou=People,dc=keytalk,dc=com

Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible.

It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

LDAPS CA Certificate
No Certificate Found

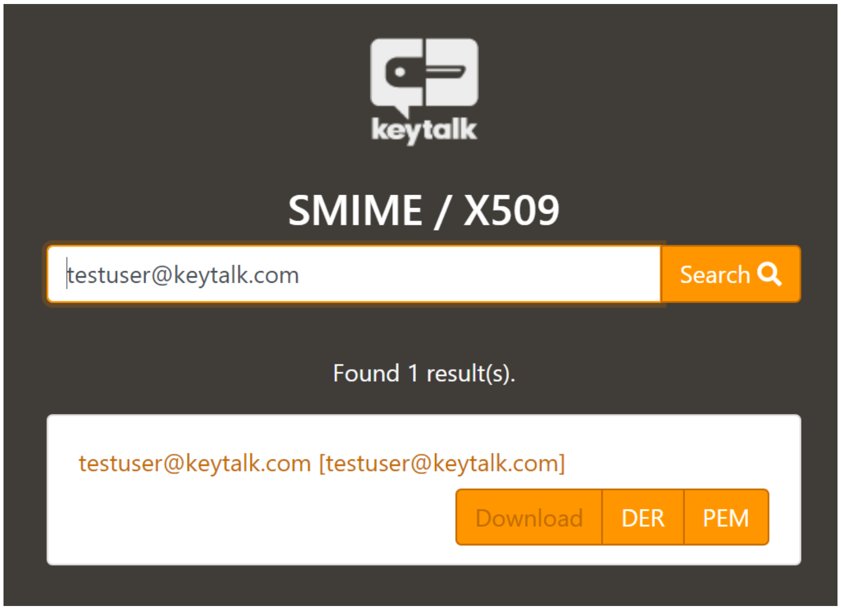
NOTE: Make sure to follow these exact settings (other than the LDAP/LDAPS url settings), as the custom LDAP schema has been configured to only work with these configurations. Ensure you check Address Book only!!

The default password is :   Change!

You can download this virtual appliance SMIME LDAP secure email address book from the KeyTalk website: https://www.keytalk.com/download#bas
The web-interface listens by default on http://<sethostname>

In order to use https://<sethostname> , http://<setipaddress> or https://<setipaddress> a configuration change is required as explained in the setup manual provided with the S/MIME LDAP download.

The HTTP(S) based S/MIME address search allows for exact match only lookups of S/MIME public key and certificate information in PEM and DER format.



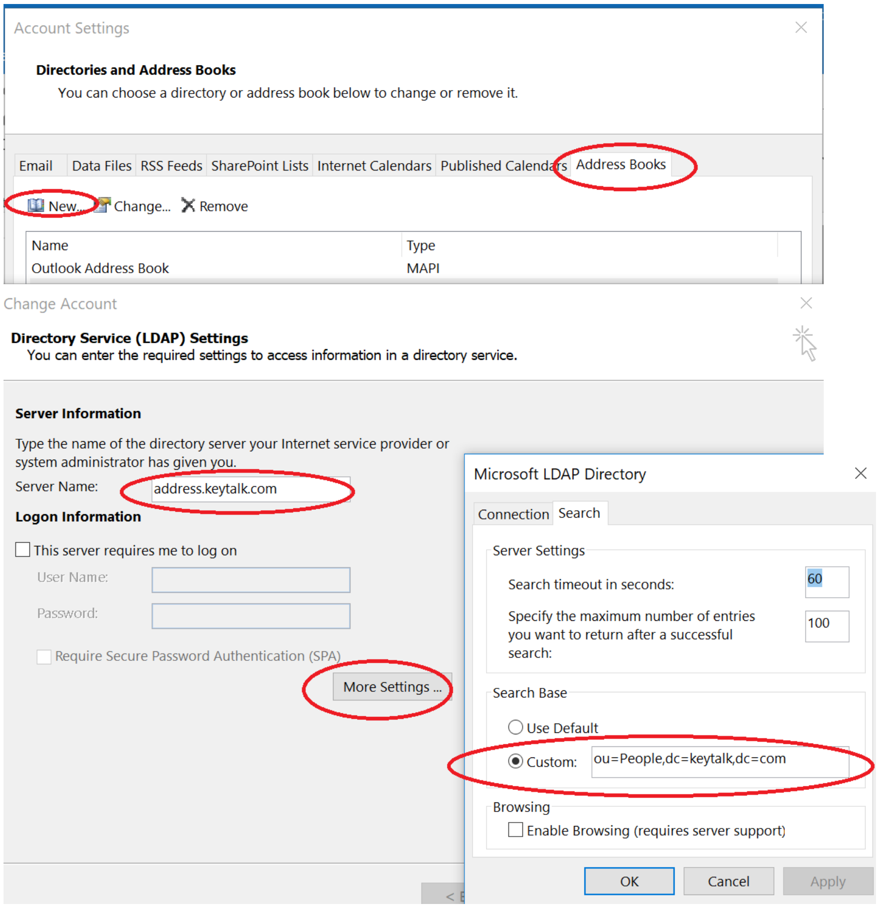
SMIME / X509

testuser@keytalk.com

Found 1 result(s).

testuser@keytalk.com [testuser@keytalk.com]

You can also use this LDAP as your mail client’s address book, by adding it manually to your mail client, (or have KeyTalk client 5.5.5 auto-configure this for your user’s Outlook and MacMail):



Account Settings

Directories and Address Books

You can choose a directory or address book below to change or remove it.

Name	Type
Outlook Address Book	MAPI

Change Account

Directory Service (LDAP) Settings

You can enter the required settings to access information in a directory service.

Server Information

Type the name of the directory server your Internet service provider or system administrator has given you.

Server Name: address.keytalk.com

Logon Information

□ This server requires me to log on

User Name:

Password:

□ Require Secure Password Authentication (SPA)

Microsoft LDAP Directory

Server Settings

Search timeout in seconds: 60

Specify the maximum number of entries you want to return after a successful search: 100

Search Base

○ Use Default

● Custom: ou=People,dc=keytalk,dc=com

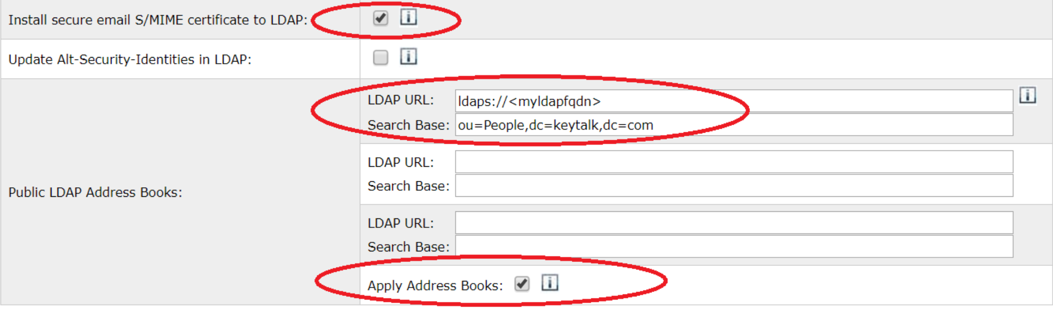
Browsing

□ Enable Browsing (requires server support)

16.8.1 REST API and KeyTalk client LDAP S/MIME address-book client auto configure

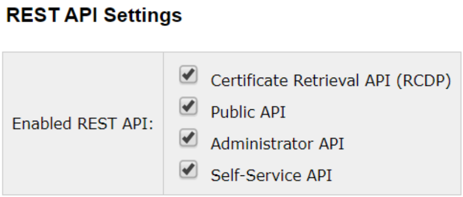
Under KeyTalk TEMPLATES an Admin can set the LDAP S/MIME server address-book details, the KeyTalk client 5.5.5+ can auto-configure a client present email program (currently Outlook and MacMail only), or the current REST-API can fetch these details.

Set it in the KeyTalk virtual appliance under the appropriate TEMPLATE as:



Install secure email S/MIME certificate to LDAP:	☑
Update Alt-Security-Identities in LDAP:	☐
Public LDAP Address Books:	LDAP URL: ldaps://<myldapfqdn> Search Base: ou=People,dc=keytalk,dc=com
	LDAP URL: Search Base:
	LDAP URL: Search Base:
	Apply Address Books: ☑

Each KeyTalk certificate template TEMPLATE can enable and disable specific external REST API calls to said template. This allows for full control on whether manual only or external scripting is allowed.



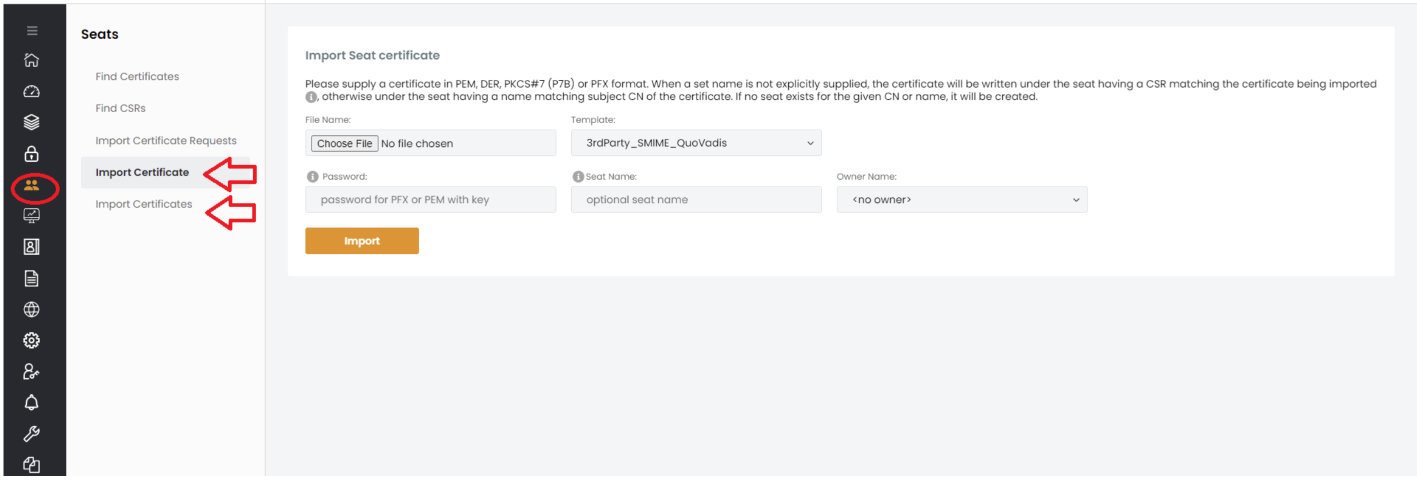
REST API Settings

Enabled REST API:	☑ Certificate Retrieval API (RCDP) ☑ Public API ☑ Administrator API ☑ Self-Service API

16.9 *Importing existing and new certificates*

Many customers already have server and client certificates (and private keys) in their organization.
So it makes sense that these can be imported into the KeyTalk solution in order for KeyTalk to manage and (re)issue these.

KeyTalk supports the importing of a single certificate (and private key), and requires a username/anchorpoint to be defined. If no username/anchorpoint is given, then the non-blank CN value of the imported certificate is used.
The username/anchorpoint usually matches the value used by the user or end-point to authenticate itself with (such as AD username or hostname)

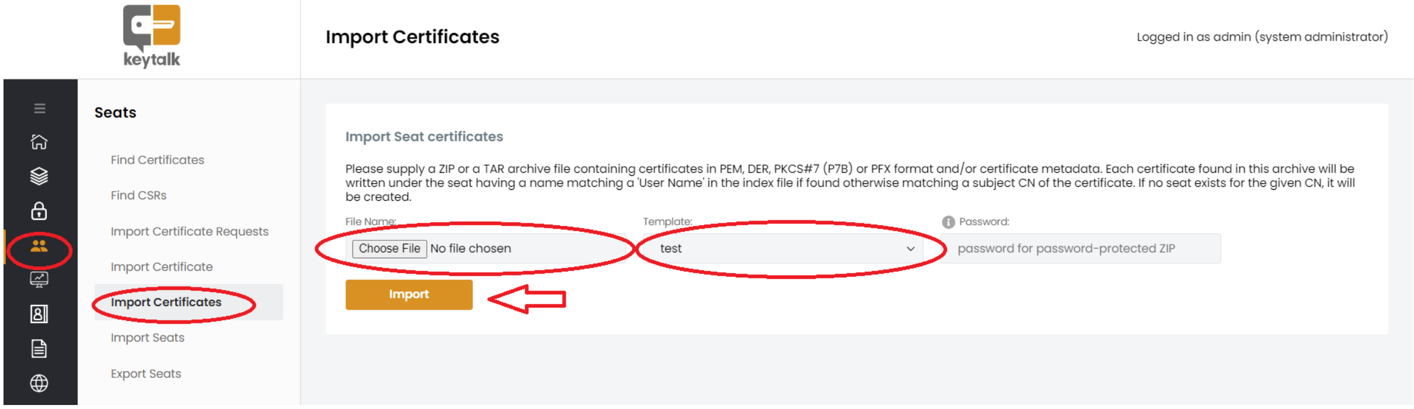


Import Seat certificate

Please supply a certificate in PEM, DER, PKCS#7 (P7B) or PFX format. When a set name is not explicitly supplied, the certificate will be written under the seat having a CSR matching the certificate being imported ⓘ, otherwise under the seat having a name matching subject CN of the certificate. If no seat exists for the given CN or name, it will be created.

File Name: Choose File No file chosen	Template: 3rdParty_SMIME_QuoVadis	
Password: password for PFX or PEM with key	Seat Name: optional seat name	Owner Name: <no owner>

KeyTalk also supports the mass importing certificates (and their private key):



Import Certificates

Logged in as admin (system administrator)

Import Seat certificates

Please supply a ZIP or a TAR archive file containing certificates in PEM, DER, PKCS#7 (P7B) or PFX format and/or certificate metadata. Each certificate found in this archive will be written under the seat having a name matching a 'User Name' in the index file if found otherwise matching a subject CN of the certificate. If no seat exists for the given CN, it will be created.

File Name: Choose File No file chosen

Template: test

Password: password for password-protected ZIP

ZIP archives might optionally be protected with a password. Only *ZipCrypto* encryption is supported for ZIP (not e.g. *AES256*)

An archive MUST contain an index file called index.csv

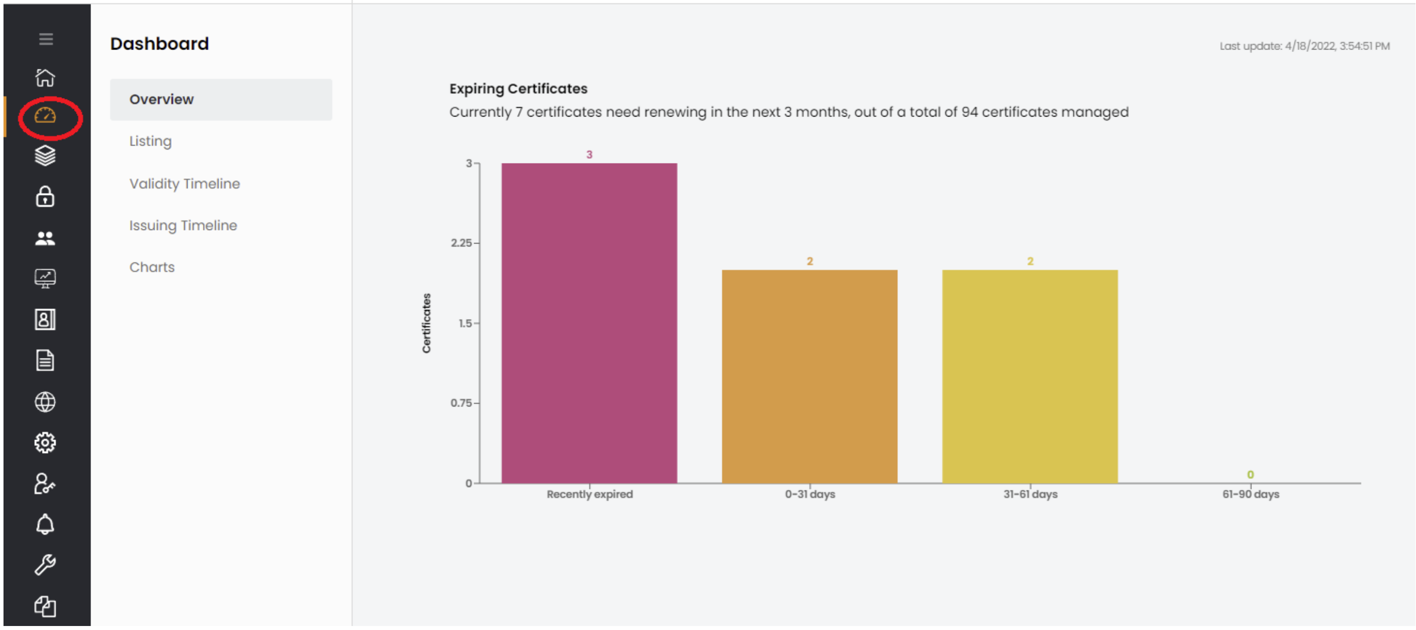
- the values are separated by commas
- values containing blanks or commas should be enclosed in double quotes (Excel takes care of it)
- values with double quote should be escaped with a double quote (Excel takes care of it)
- index.csv MUST contain 3 columns separated by commas named: File Name, Password and User Name respectively.
- the first row is a header and contains row names above
- a file can be in PEM, DER or PFX format; PEM must contain cert , and may contain a cert with a key
- Password is either Pfx password or a password for the encrypted key in PEM
- User Name, if non-blank, specifies a KeyTalk DevID user name to use for the cert/key
- any error during processing of an individual certificate references in the index file are tolerated (e.g. file does not exist, password is not correct)

A sample mass import zip file can be found here:
https://downloads.keytalk.com/downloads/samples/sample_mass-certificate-import.zip

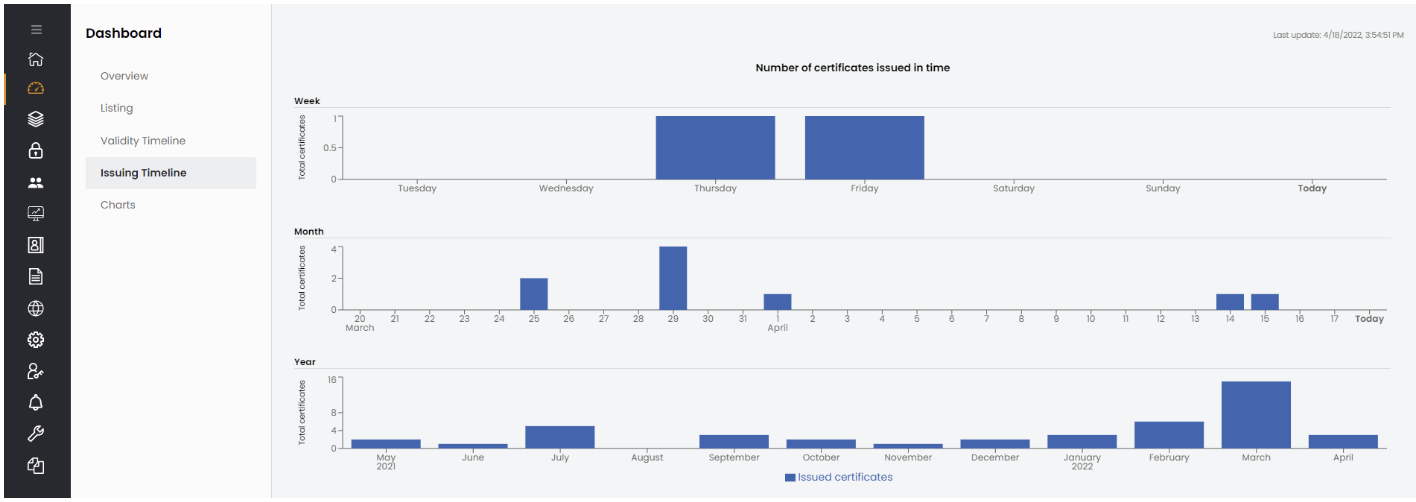
17 Certificate reporting and management meta-data

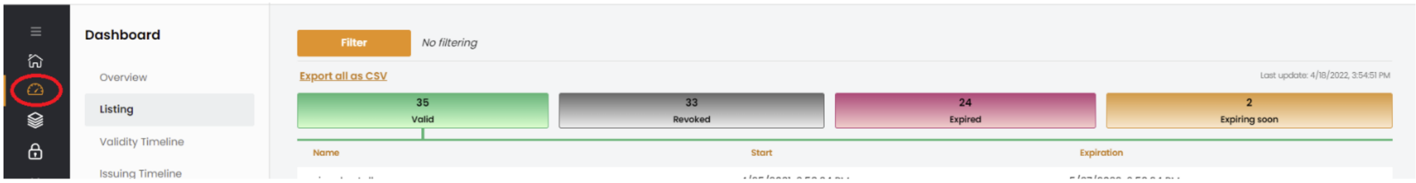
The KeyTalk CKMS provides reporting on managed certificates and keys.
As of firmware 6.4.2 Dashboard reporting got introduced, and will eventually replace the previous implementation of reporting.

17.1 Dashboard reporting



Currently 7 certificates need renewing in the next 3 months, out of a total of 94 certificates managed



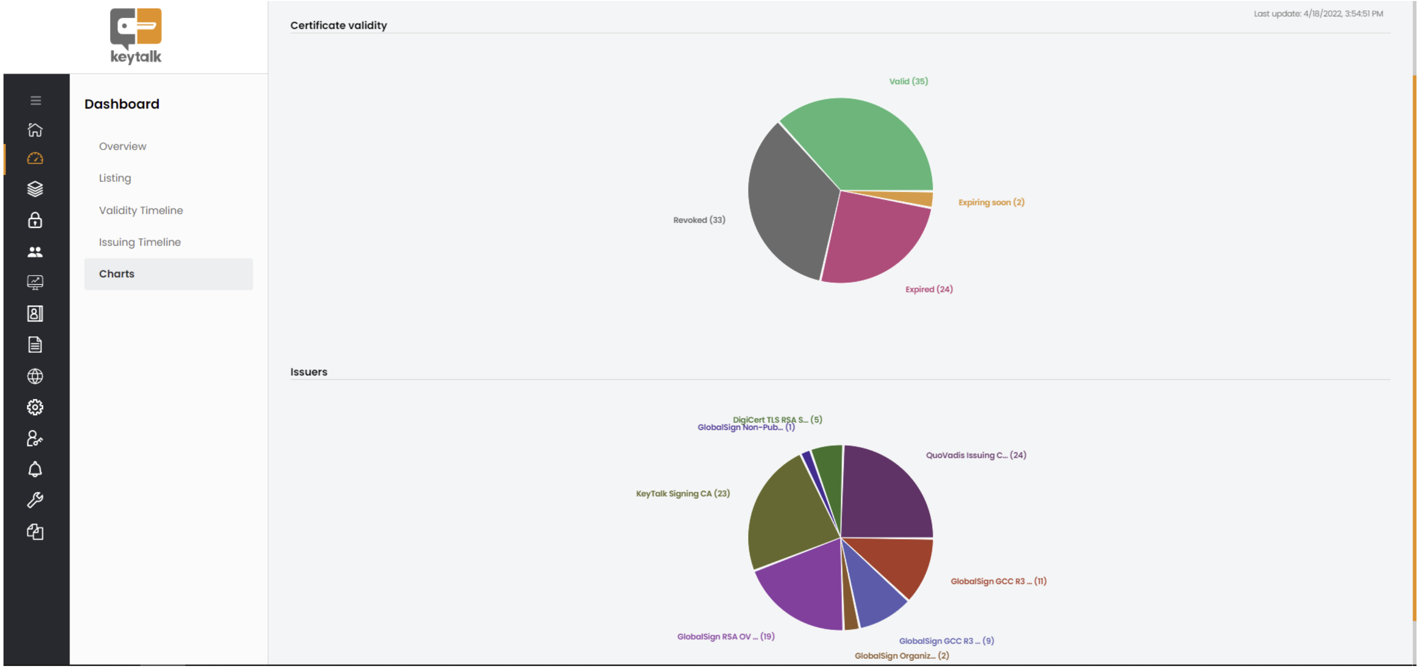


No filtering

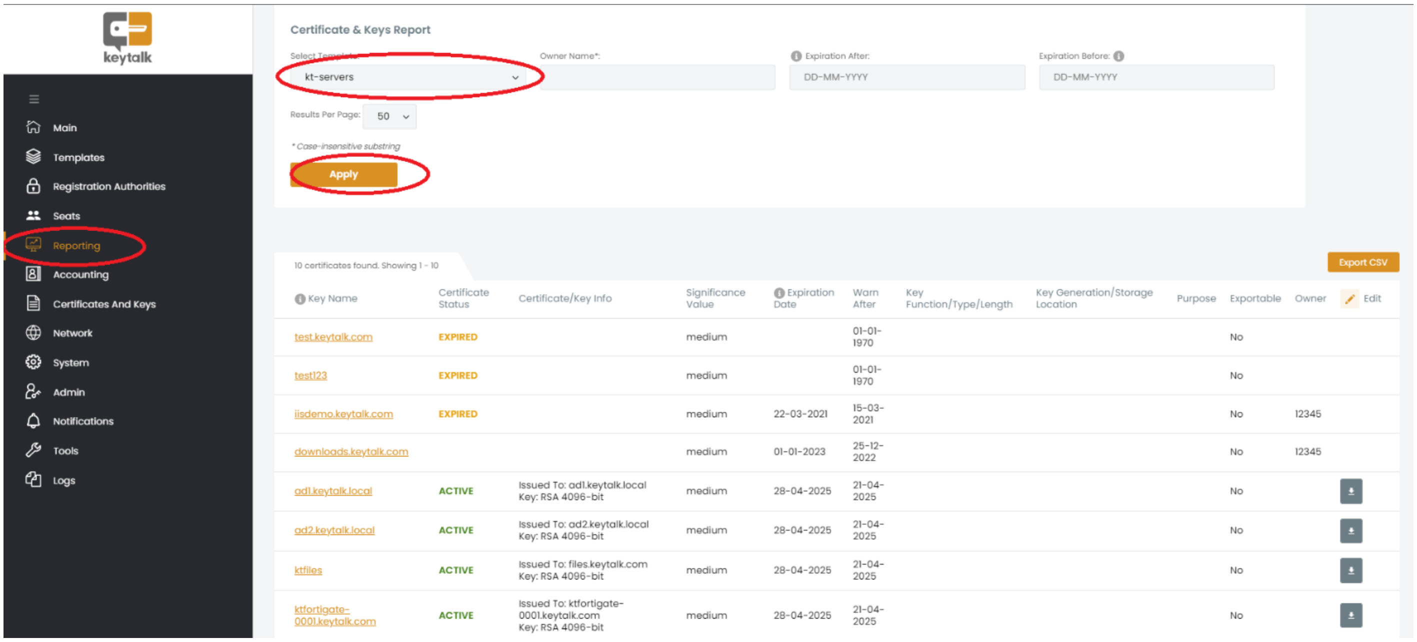
Last update: 4/18/2022, 3:54:51 PM

35 Valid	33 Revoked	24 Expired	2 Expiring soon

Name	Start	Expiration



17.2 Legacy reporting



Certificate & Keys Report

Select Template: kt-servers

Owner Name*:

Expiration After: DD-MM-YYYY

Expiration Before: DD-MM-YYYY

Results Per Page: 50

* Case-insensitive substring

Apply

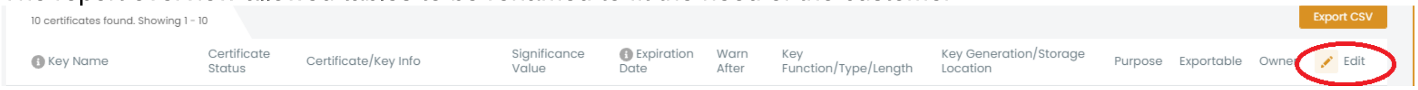
10 certificates found. Showing 1 - 10

Export CSV

Key Name	Certificate Status	Certificate/Key Info	Significance Value	Expiration Date	Warn After	Key Function/Type/Length	Key Generation/Storage Location	Purpose	Exportable	Owner	Edit
test.keytalk.com	EXPIRED		medium		01-01-1970				No		
test123	EXPIRED		medium		01-01-1970				No		
iisdemo.keytalk.com	EXPIRED		medium	22-03-2021	15-03-2021				No	12345	
downloads.keytalk.com			medium	01-01-2023	25-12-2022				No	12345	
ad1.keytalk.local	ACTIVE	Issued To: ad1.keytalk.local Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ad2.keytalk.local	ACTIVE	Issued To: ad2.keytalk.local Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ktfiles	ACTIVE	Issued To: files.keytalk.com Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇
ktfortigate-0001.keytalk.com	ACTIVE	Issued To: ktfortigate-0001.keytalk.com Key: RSA 4096-bit	medium	28-04-2025	21-04-2025				No		⬇

The report results in an easy to understand overview of managed certificates and keys, and shows relevant meta data as (additionally) provided

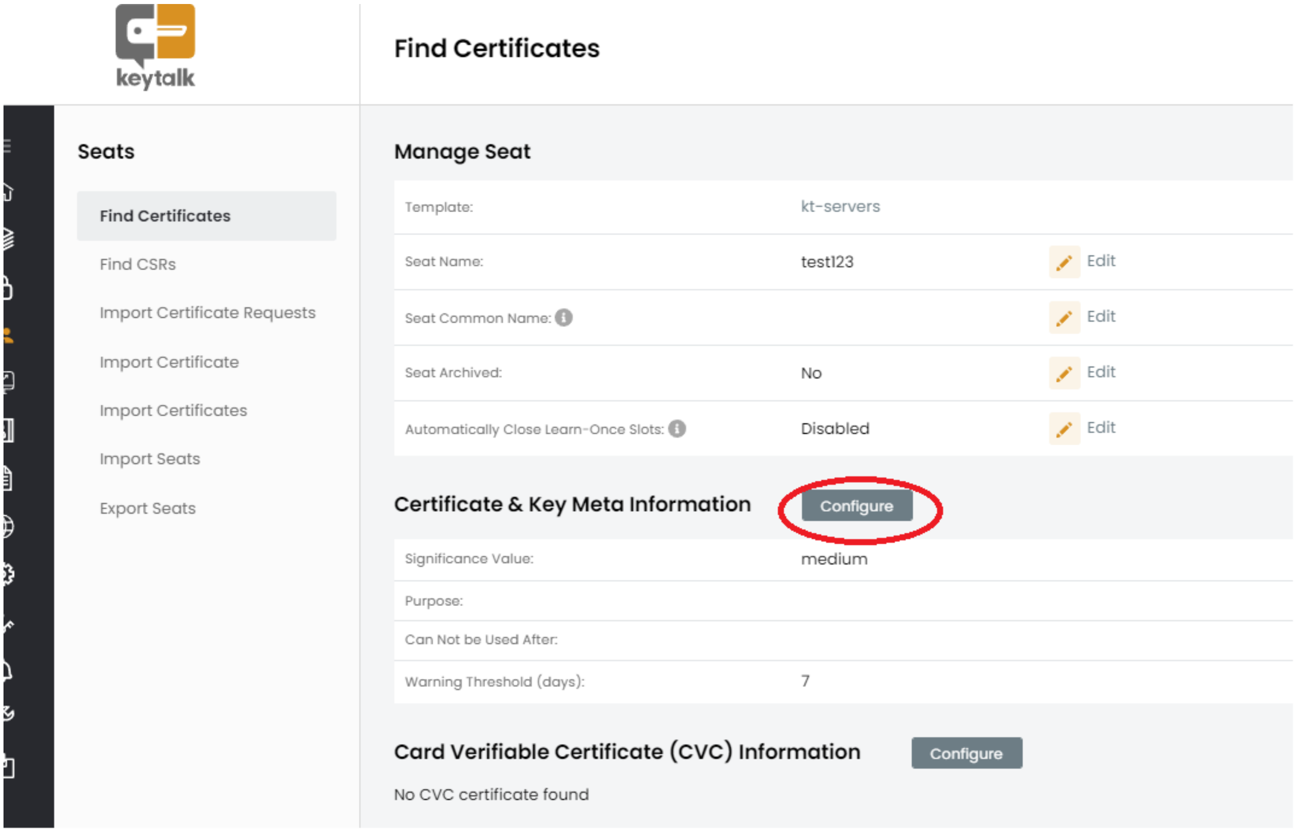
The report overview allowed tables to be renamed to fit the need of the customer.



10 certificates found. Showing 1 - 10

Export CSV

Key Name	Certificate Status	Certificate/Key Info	Significance Value	Expiration Date	Warn After	Key Function/Type/Length	Key Generation/Storage Location	Purpose	Exportable	Owne	Edit



Find Certificates

Manage Seat

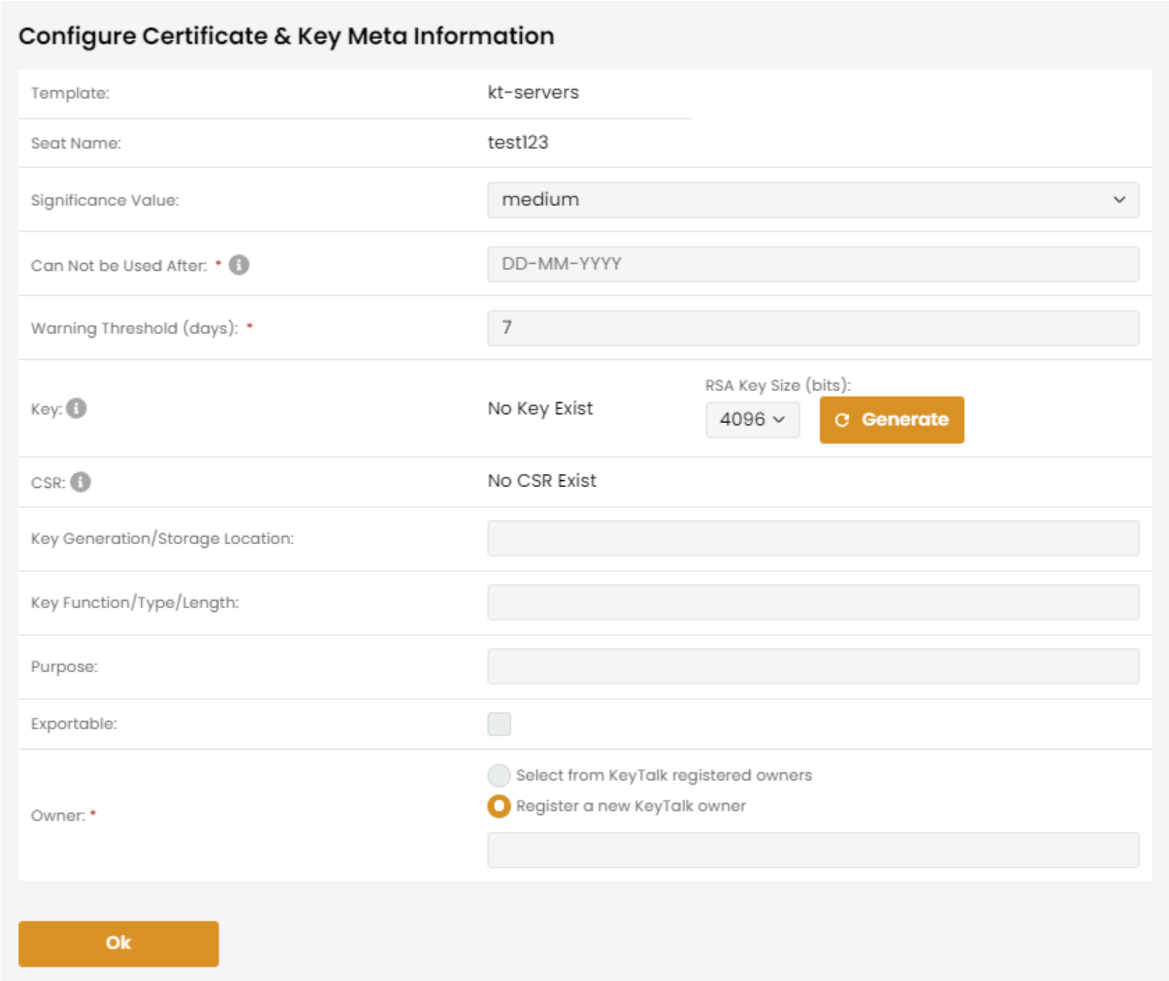
Template:	kt-servers	
Seat Name:	test123	Edit
Seat Common Name:		Edit
Seat Archived:	No	Edit
Automatically Close Learn-Once Slots:	Disabled	Edit

Certificate & Key Meta Information

Significance Value:	medium
Purpose:	
Can Not be Used After:	
Warning Threshold (days):	7

Card Verifiable Certificate (CVC) Information

No CVC certificate found



Configure Certificate & Key Meta Information

Template:	kt-servers
Seat Name:	test123
Significance Value:	medium
Can Not be Used After: *	DD-MM-YYYY
Warning Threshold (days): *	7
Key:	No Key Exist RSA Key Size (bits): 4096 Generate
CSR:	No CSR Exist
Key Generation/Storage Location:	
Key Function/Type/Length:	
Purpose:	
Exportable:	□
Owner: *	○ Select from KeyTalk registered owners ● Register a new KeyTalk owner

18 User browser based self-service portal

Any end-point users who has a KeyTalk agent, can obtain a valid certificate from a given KeyTalk TEMPLATE

Most organizations have an IT TEMPLATE desk to help end-point users, but some organizations choose to give limited control to an end-point user.

For this purpose, KeyTalk offers its webbased Self-TEMPLATE portal from any modern browser, reachable from: https://<ip_address>:3000 or https://<FQDN>:3000

The portal allows an end-point user **with a KeyTalk virtual appliance issued trusted valid client authentication certificate**, to Single-Sign-On authenticate using 2-way SSL over TLS to the KeyTalk Self-TEMPLATE portal.

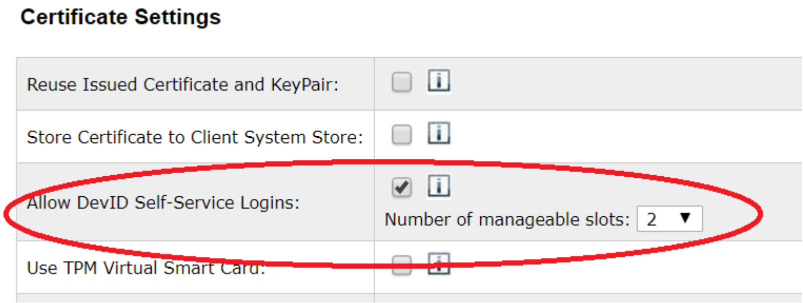
The end-point user can:

- ✓ View his known devices
- ✓ Revoke his active certificates
- ✓ See active certificate details
- ✓ Manage within Admin set limitation new and existing trusted device slots

End-point users can NOT reset their IDP passwords, as the KeyTalk virtual application server generally does not control their IDP.

Each KeyTalk client DOES have the ability to change a password when it's about to expire, or when it has expired, using the KeyTalk client, and provided appropriate configuration settings in KeyTalk support this.

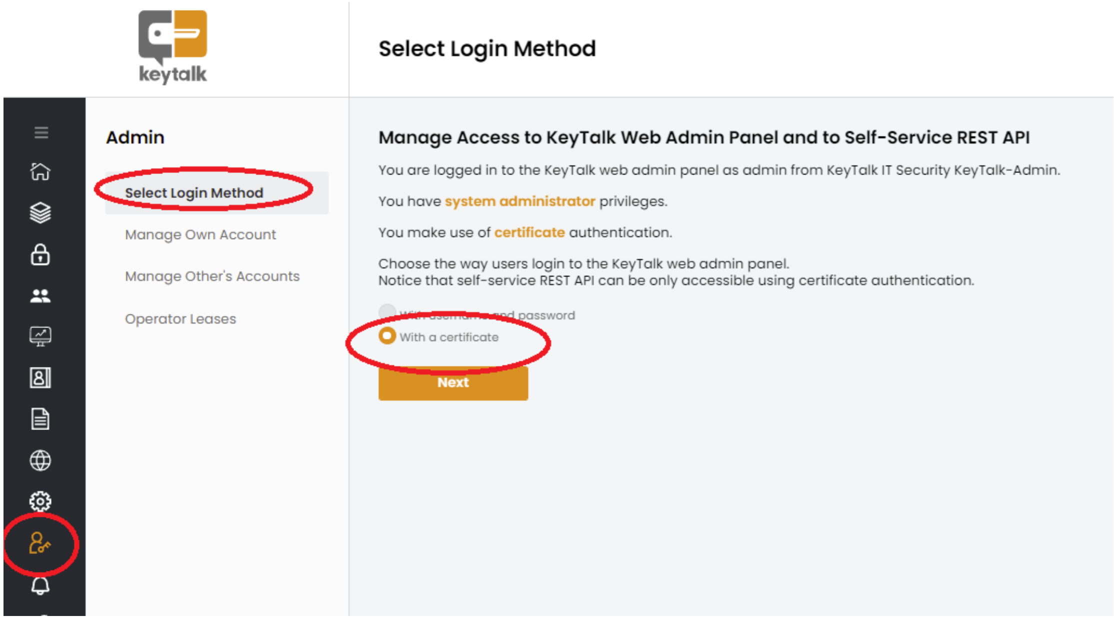
To enable the self-TEMPLATE webbased portal, select its enablement for the appropriate KeyTalk TEMPLATE:



Certificate Settings

Reuse Issued Certificate and KeyPair:	☐ ⓘ
Store Certificate to Client System Store:	☐ ⓘ
Allow DevID Self-Service Logins:	☑ ⓘ Number of manageable slots: 2 ▼
Use TPM Virtual Smart Card:	☐ ⓘ

Additionally, ensure that the KeyTalk virtual appliance enforces client certificate based login:



Manage Access to KeyTalk Web Admin Panel and to Self-Service REST API

You are logged in to the KeyTalk web admin panel as admin from KeyTalk IT Security KeyTalk-Admin.

You have system administrator privileges.

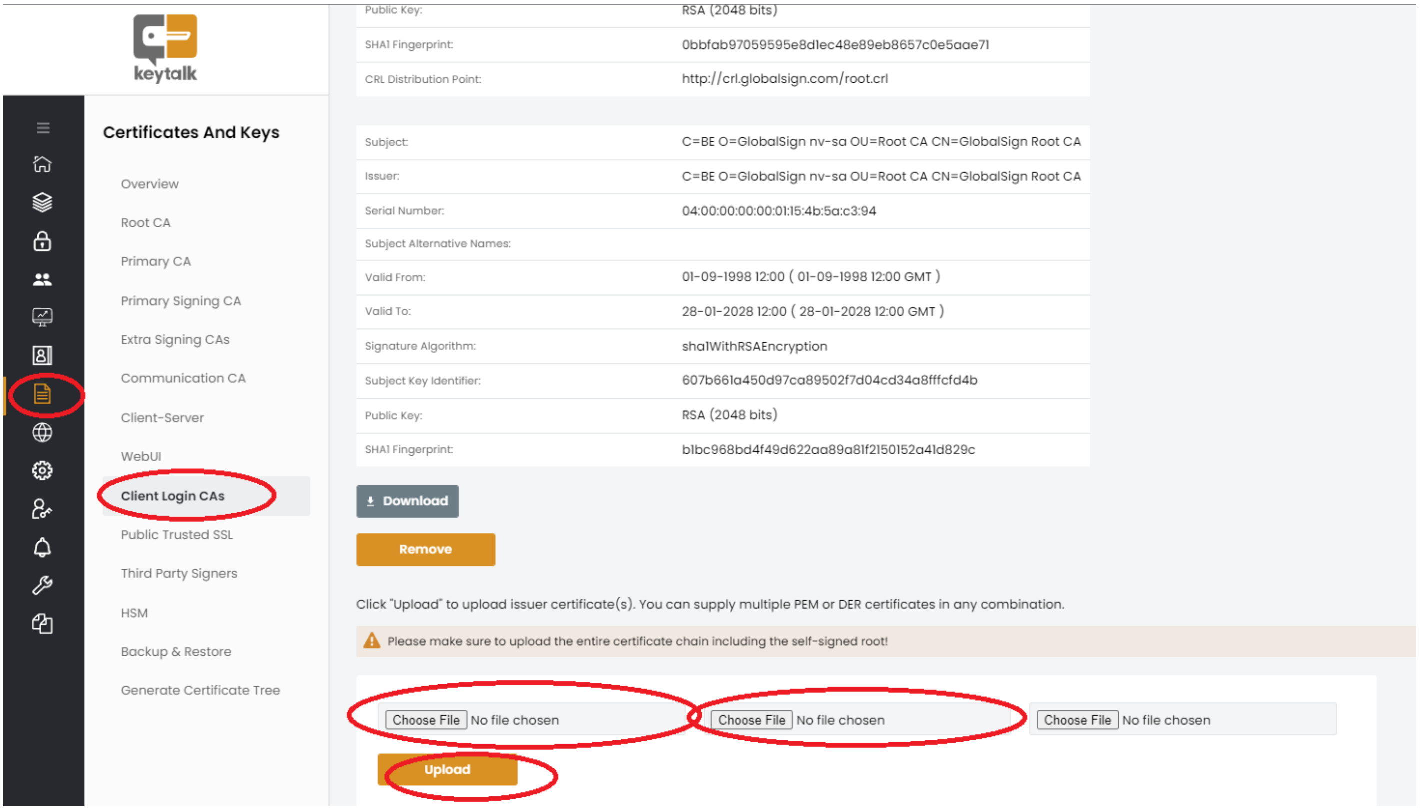
You make use of certificate authentication.

Choose the way users login to the KeyTalk web admin panel.
Notice that self-service REST API can be only accessible using certificate authentication.

○ With username and password
● With a certificate

Next

Other CA sources can be set as trusted as well for login purposes both for the self-TEMPLATE portal and management UI access. Set the additional trust here:



Public Key:	RSA (2048 bits)
SHA1 Fingerprint:	0bbfab97059595e8d1ec48e89eb8657c0e5aae71
CRL Distribution Point:	http://crl.globalsign.com/root.crl

Subject:	C=BE O=GlobalSign nv-sa OU=Root CA CN=GlobalSign Root CA
Issuer:	C=BE O=GlobalSign nv-sa OU=Root CA CN=GlobalSign Root CA
Serial Number:	04:00:00:00:00:01:15:4b:5a:c3:94
Subject Alternative Names:	
Valid From:	01-09-1998 12:00 (01-09-1998 12:00 GMT)
Valid To:	28-01-2028 12:00 (28-01-2028 12:00 GMT)
Signature Algorithm:	sha1WithRSAEncryption
Subject Key Identifier:	607b661a450d97ca89502f7d04cd34a8fffcfd4b
Public Key:	RSA (2048 bits)
SHA1 Fingerprint:	b1bc968bd4f49d622aa89a81f2150152a41d829c

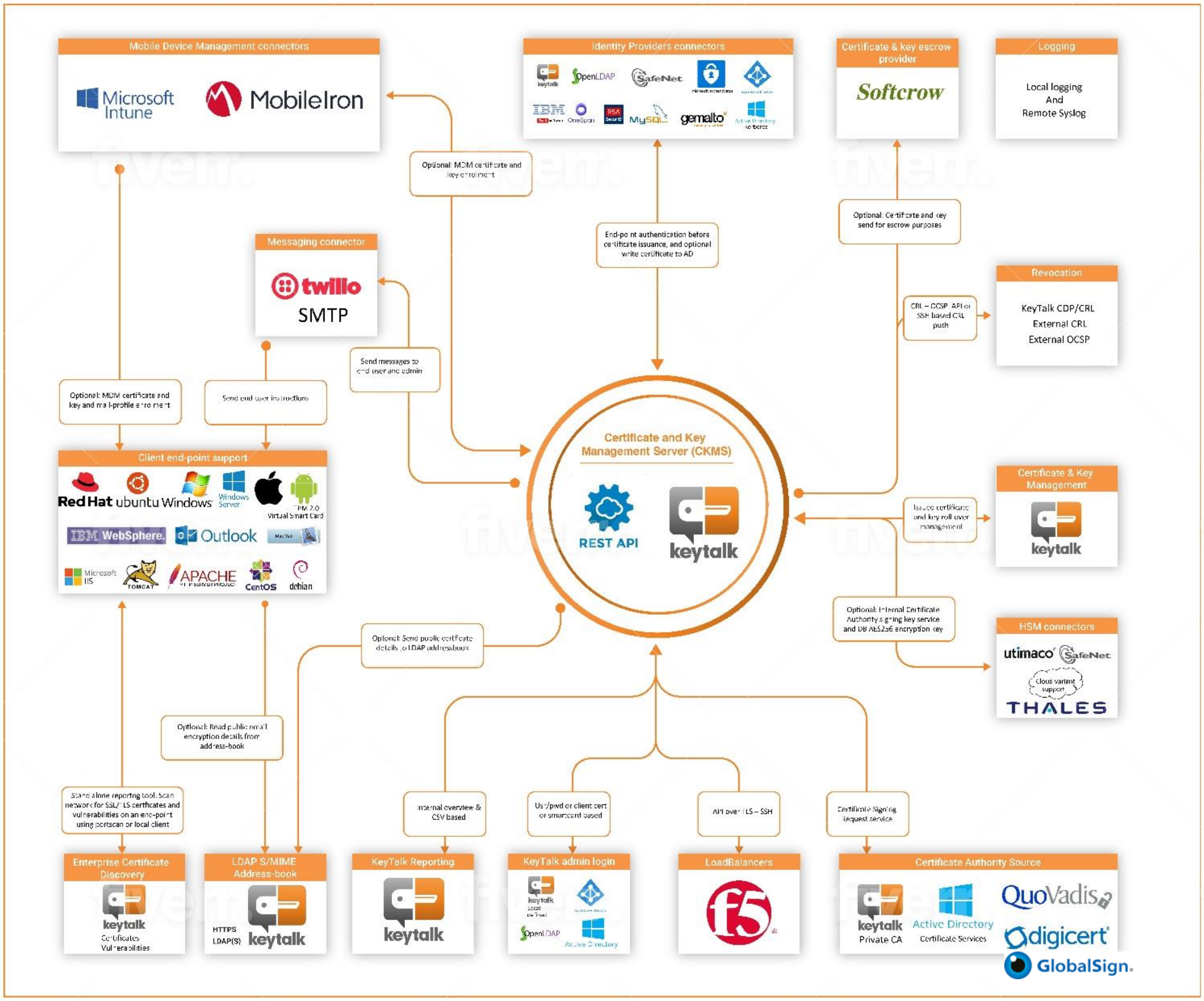
Click "Upload" to upload issuer certificate(s). You can supply multiple PEM or DER certificates in any combination.

⚠ Please make sure to upload the entire certificate chain including the self-signed root!

19 Network architecture: KeyTalk in a network and HA

While in its most simple form a single KeyTalk application server is used with an internal database user entry, and just a client and/or server end-point, most production environments will run KeyTalk in a High Availability setup with multiple connected components.

It can look like: https://downloads.keytalk.com/images/ktoverview.jpg



Mobile Device Management connectors
Microsoft Intune
MobileIron
Identity Providers connectors
keytalk
OpenLDAP
SafeNet
IBM
OneSpan
RSA SecurID
MySQL
gemalto
Active Directory Kerberos
Certificate & key escrow provider
Softcrow
Logging
Local logging And Remote Syslog
Optional: MDM certificate and key enrolment
End-point authentication before certificate issuance, and optional write certificate to AD
Optional: Certificate and key send for escrow purposes
Messaging connector
twilio
SMTP
Revocation
CRL – OCSP API or SSH based CRL push
KeyTalk CDP/CRL
External CRL
External OCSP
Send messages to end user and admin
Optional: MDM certificate and key and mail-profile enrolment
Send end user instructions
Client end-point support
Red Hat
ubuntu
Windows
Windows Server
TPM 2.0 Virtual Smart Card
IBM WebSphere.
Outlook
Microsoft IIS
TOMCAT
APACHE
CentOS
debian
Certificate and Key Management Server (CKMS)
REST API
keytalk
Certificate & Key Management
Issued certificate and key roll over management
keytalk
Optional: Internal Certificate Authority signing key service and DB AES256 encryption key
HSM connectors
utimaco
SafeNet
Cloud variant support
THALES
Optional: Send public certificate details to LDAP addressbook
Optional: Read public email encryption details from address-book
Stand alone reporting tool: Scan network for SSL/TLS certficates and vulnerabilities on an end-point using portscan or local client
Internal overview & CSV based
Usr/pwd or client cert or smartcard based
API over TLS – SSH
Certificate Signing Request service
Enterprise Certificate Discovery
keytalk
Certificates Vulnerabilities
LDAP S/MIME Address-book
HTTPS
LDAP(S)
keytalk
KeyTalk Reporting
keytalk
KeyTalk admin login
keytalk Local defined
OpenLDAP
Active Directory
LoadBalancers
f5
Certificate Authority Source
keytalk Private CA
Active Directory Certificate Services
QuoVadis
digicert
GlobalSign.

The amount of KeyTalk application servers needed in a given network environment strongly depends on many factors, most notably the required certificate key-length when KeyTalk is required to generate the key-pair on-demand, and the amount of certificates that need to be issued per second.

Most customers choose to LoadBalance minimally 2 KeyTalk virtual appliances, which all get connected to a MySQL cluster.
Other customers simply choose NOT to setup KeyTalk in HA and rely on a single KeyTalk instance which get auto backup-ed several times a day. A single KeyTalk instance can support, based on its internal database, up to 400.00 certificates (and keys).

Most customer choose to run the KeyTalk virtual application server either in their DMZ, or in a separate VLAN.

NOTE: **When using an LB, ensure that IP affinity is configured and HTTP passthrough is used. Its possible to terminate TLS traffic however this will result in issues when using client certificate based authentication.**

19.1 High Availability setup

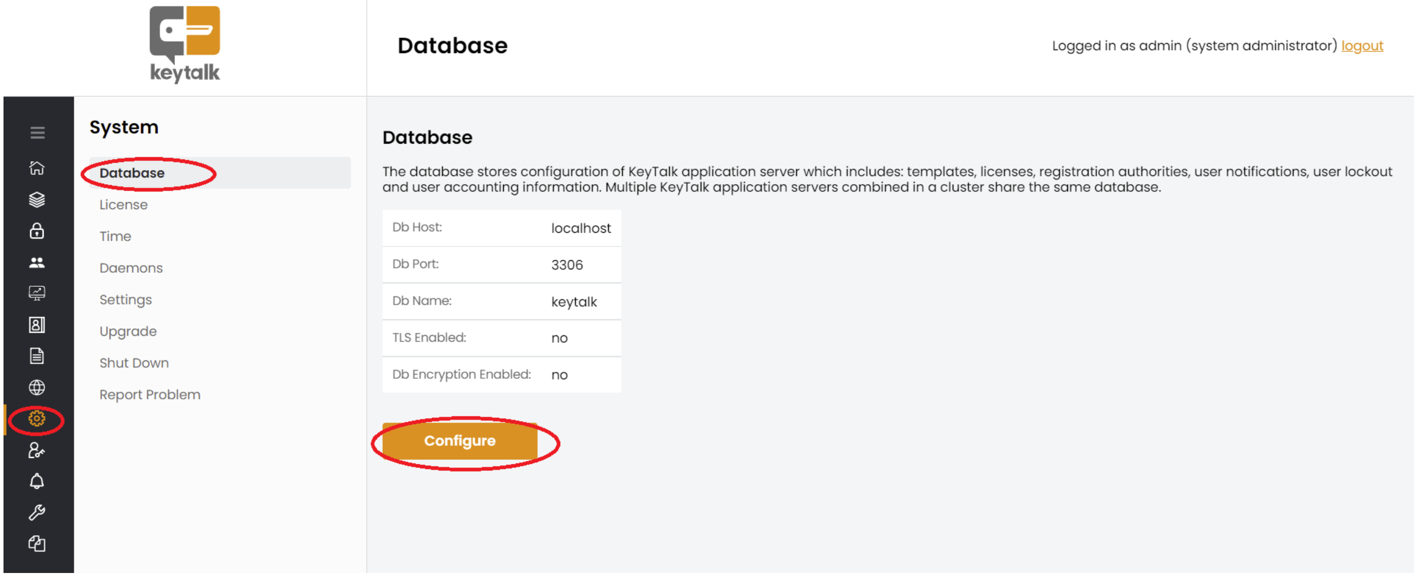
When HA is required, the KeyTalk virtual appliance functionality can be separated from the database used to store shared data. Shared data includes issued certificates and keys, certificate template information (TEMPLATES), authentication connectors and much more.

As a result multiple KeyTalk virtual appliances can be configured behind a Load Balancer, and connected to a single or clustered MySQL Db.

The MySQL Db must be minimally version 5.7.8, though most customer run version 8
The KeyTalk virtual appliance download, contains a preconfigured single instance of a MySQL Db.

For HA it is advisable to setup a MySQL cluster.

To configure the external MySQL, use:



Logged in as admin (system administrator) logout

Database

The database stores configuration of KeyTalk application server which includes: templates, licenses, registration authorities, user notifications, user lockout and user accounting information. Multiple KeyTalk application servers combined in a cluster share the same database.

Db Host:	localhost
Db Port:	3306
Db Name:	keytalk
TLS Enabled:	no
Db Encryption Enabled:	no

A preconfigured SSL certificate is shipped with the MySQL instance, but should always be replaced with a production version since the private key is considered compromised and the SAN likely wont match.

19.2 High Availability and firmware upgrades

When HA is setup, and a firmware update is performed, the KeyTalk virtual appliance will share the firmware update with the connected MySQL Db, which in turn shares it with all connected KeyTalk virtual appliances.

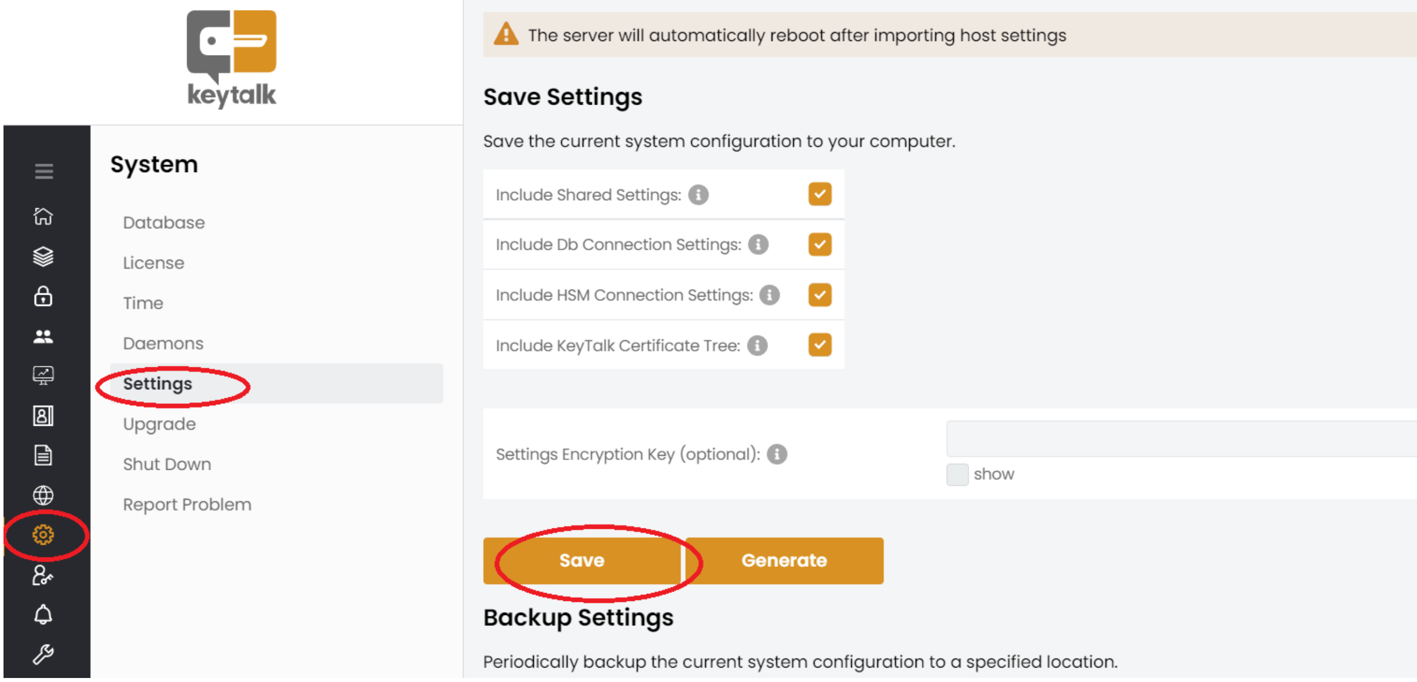
As a result, you can upgrade your firmware in a KeyTalk cluster within seconds.

The firmware upgrade will also update the MySQL schema to match new functionality and corresponding shared data fields.

Should you ever find yourself in a situation where the MySQL Db schema is lower than that of your KeyTalk virtual appliance, the MySQL Db will not be able to connect to the higher version of the KeyTalk virtual appliance, and the KeyTalk virtual appliance will enforce the use of its internal Db.

If this ever happens, the KeyTalk virtual appliance will inform the (network) Admin and provide an update button to force update the MySQL schema to match that of the KeyTalk virtual appliance firmware.

If you also ever get into a situation where your local Db settings need to be moved to the remote MySQL Db then, simply backup your local Db using:



The server will automatically reboot after importing host settings

Save Settings

Save the current system configuration to your computer.

Include Shared Settings:	☑
Include Db Connection Settings:	☑
Include HSM Connection Settings:	☑
Include KeyTalk Certificate Tree:	☑

Settings Encryption Key (optional):

□ show

Backup Settings

Periodically backup the current system configuration to a specified location.

Then connect the external MySQL, and import your backup settings using the Load Settings features.

20 KeyTalk webUI management roles and strong authentication

The KeyTalk application server comes with 1 single local SysAdmin username and password.

All other KeyTalk management role settings and accounts are synchronized across all KeyTalk virtual appliances, provided they are connected using the same MySQL (cluster).

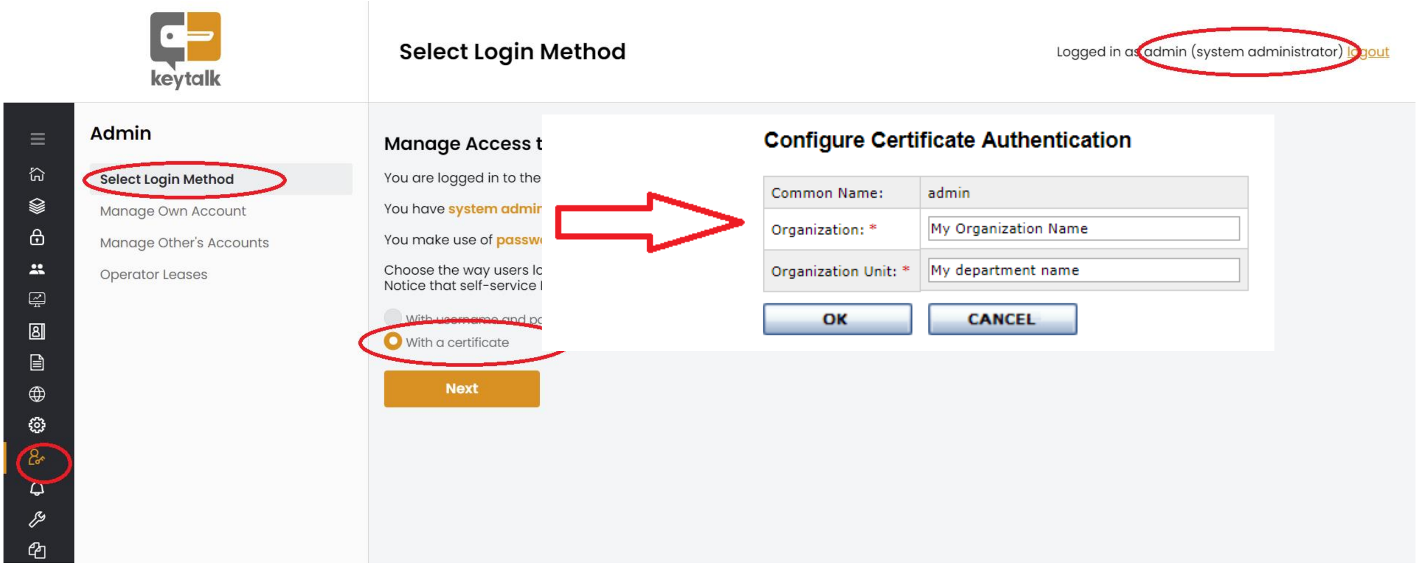
KeyTalk currently differentiates the following roles:

Role	Authorization
Local System Administrator	Full read/write access (account settings NOT shared to other KeyTalk instances)
Cluster Administrator	Full read/write access for shared data (settings shared to other KeyTalk instances)
Auditor	Full read only access (except for passwords and private keys)
Network Administrator	Read/write access to Network, System, and local logs
TEMPLATE Admin	Read/write access to designated TEMPLATES, Authentication Modules and corresponding DevID user administration. At least 1 TEMPLATE must be assigned, or the account is considered inactive. (Mass-)enrolment and (mass-)revocation Can approve 4-eye principle download requests of TEMPLATE Operators
TEMPLATE Operator	Read/write access to designated DevID user administration. At least 1 TEMPLATE must be assigned, or the account is considered inactive. A 4-eye principle applies to downloading of active and historic certificates and keys. Single certificate enrolment and singe certificate revocation Read access to Authentication module settings and TEMPLATE template settings (except for passwords and private keys)
SEAT Owner	Same as TEMPLATE Operator except for individually assigned seats. Owners are informed about “their certificate” or “their key”, but cannot login into the KeyTalk management environment
Self TEMPLATE User	Only active if the Self TEMPLATE portal on a TEMPLATE level is enabled. Read access to his/her own account within the limitation of the set amount of device slots Write access to managing his/her own device slots, provided that this feature was enabled by the Admin Write access to request third party certificates, provided that this feature has been enabled.
SSL Discovery Manager	Used as the authentication account to submit KeyTalk SSL Scanner results to a designated templated as authorized under the account.

While username/password based authentication is a basic necessity for initial setup, it is strongly advised to enable strong authentication to the webUI for all roles.

KeyTalk can enforce valid client certificate-based authentication issued **under the KeyTalk private CA or another external CA**, to its webUI when strong authentication is enabled by the KeyTalk System Administrator.

To enable strong authentication, make a backup first, and have the System Administrator enabled initially on just 1 virtual appliance, and once its verified to work properly, enable it on all active KeyTalk virtual application servers:



Select Login Method

Logged in as admin (system administrator) logout

Admin

Manage Access t

You are logged in to the

You have system admir

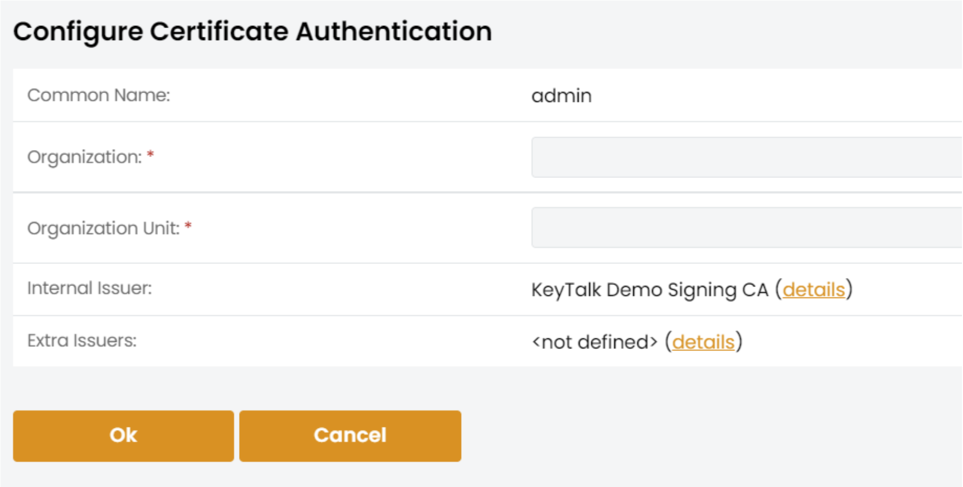
You make use of passw

Choose the way users lo
Notice that self-service i

○ With username and pa
● With a certificate

Configure Certificate Authentication

Common Name:	admin
Organization: *	My Organization Name
Organization Unit: *	My department name



Configure Certificate Authentication

Common Name:	admin
Organization: *	
Organization Unit: *	
Internal Issuer:	KeyTalk Demo Signing CA (details)
Extra Issuers:	<not defined> (details)

define the corresponding mandatory certificate Organization value and Organizational Unit value for the user “admin” (ie ensure the user “admin” exists on your IDP), these values must match the values in the issued certificates for the management roles.

Since client certificates are used for strong authentication, the System Administrator can choose to use whatever RA source is used such as Active Directory, or existing Radius based tokens, and optionally leverage this authentication with an additional KeyTalk hardware recognition authentication factor.

It is advised to at least define the admin account in the local KeyTalk Db to issue the initial Admin certificate under a unique management TEMPLATE. This way setting up strong authentication before switching to an AD is easier to control. When switching to for example AD, the username “admin” needs to be defined in the AD as a separate user, or the value “admin” must come from a mapped AD attribute to the certificate CN (see LDAP/AD certificate attribute mapping in chapter 12.2 of this manual.

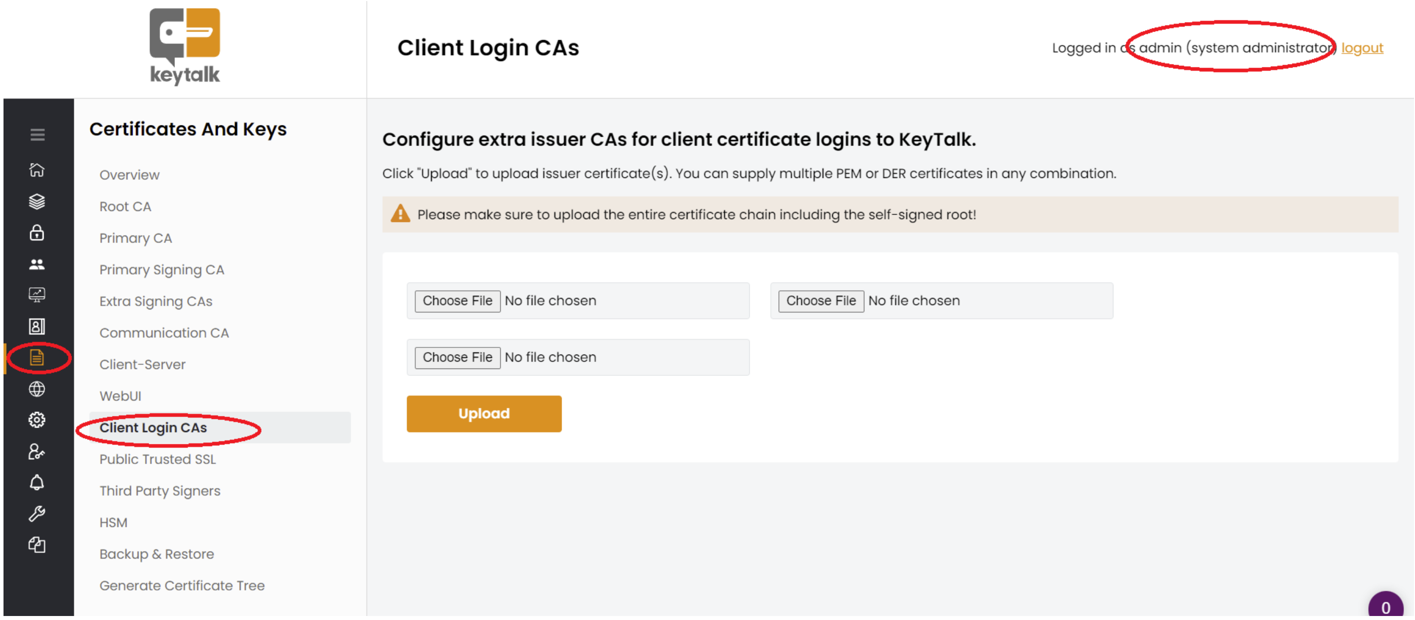
Should a System Administrator ever be locked out of a KeyTalk application server, then login using the KeyTalk CLI admin and run: /usr/local/bin/keytalk/www/reset-admin-passwd
This will set the login to factory default username/password.

20.1 Enable external CA-source trust for strong authentication

The KeyTalk virtual appliance by default will only trust its own internal Certificate Authority as a source for certificates used to authenticate to it.

Expanding the trust to other CAs requires these CA's to be imported onto the KeyTalk virtual appliance as an additional trust.

Ensure you get the entire CA tree of the external CA source, and import it in PEM or DER format in:



Client Login CAs

Logged in as admin (system administrator) logout

Configure extra issuer CAs for client certificate logins to KeyTalk.

Click "Upload" to upload issuer certificate(s). You can supply multiple PEM or DER certificates in any combination.

Please make sure to upload the entire certificate chain including the self-signed root!

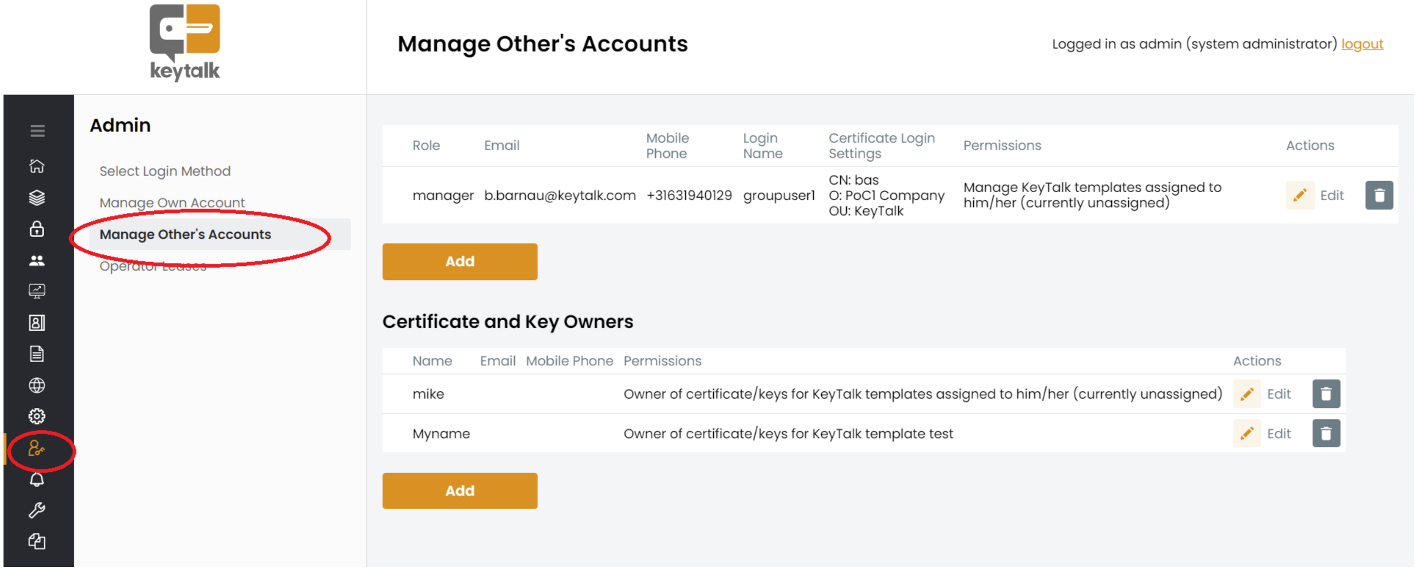
Choose File No file chosen

Choose File No file chosen

Choose File No file chosen

20.2 Active Directory for Management Access

The KeyTalk virtual appliance by default will only make use of its own defined user accounts and designated authorization roles as defined in:



Manage Other's Accounts

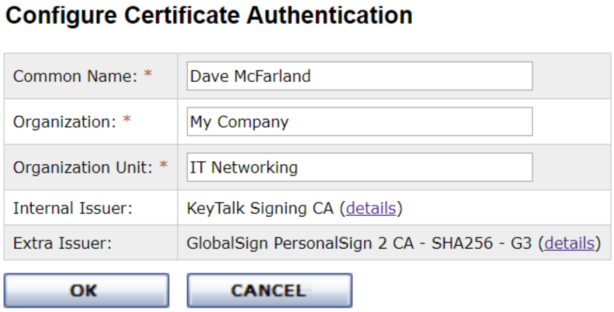
Logged in as admin (system administrator) logout

Role	Email	Mobile Phone	Login Name	Certificate Login Settings	Permissions	Actions
manager	b.barnau@keytalk.com	+31631940129	groupuser1	CN: bas O: PoC1 Company OU: KeyTalk	Manage KeyTalk templates assigned to him/her (currently unassigned)	Edit

Certificate and Key Owners

Name	Email	Mobile Phone	Permissions	Actions
mike			Owner of certificate/keys for KeyTalk templates assigned to him/her (currently unassigned)	Edit
Myname			Owner of certificate/keys for KeyTalk template test	Edit

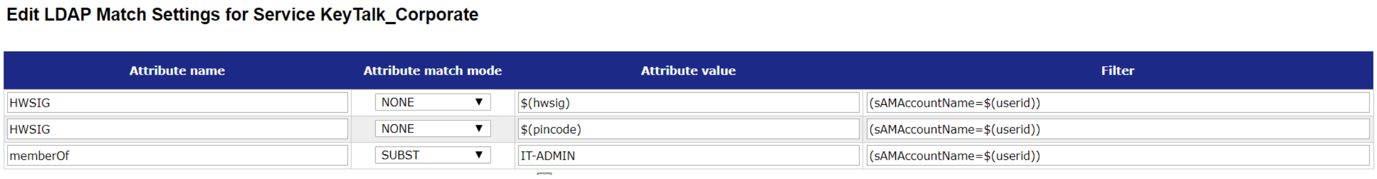
Each defined KeyTalk management user account can be mapped to specific client authentication certificate subject meta data.



Configure Certificate Authentication

Common Name: *	Dave McFarland
Organization: *	My Company
Organization Unit: *	IT Networking
Internal Issuer:	KeyTalk Signing CA (details)
Extra Issuer:	GlobalSign PersonalSign 2 CA - SHA256 - G3 (details)

By creating a KeyTalk TEMPLATE, that gets connected to the company AD/LDAP (under Authentication Modules), and optionally restricting the Bind to at least 1 matching Security Group, for example:



Edit LDAP Match Settings for Service KeyTalk_Corporate

Attribute name	Attribute match mode	Attribute value	Filter
HWSIG	NONE	$(hwsig)	(sAMAccountName=$(userid))
HWSIG	NONE	$(pincode)	(sAMAccountName=$(userid))
memberOf	SUBST	IT-ADMIN	(sAMAccountName=$(userid))

You can issue client certificates to your KeyTalk Management staff , enabling them to login into the KeyTalk Management Portal, using strong certificate based two SSL over TLS authentication, based on their regular AD credentials.

21 Log files and reports

Each KeyTalk application server keeps track of 1 log file per Daemon:

Daemon	Purpose
AuthD	Deals with all incoming and outgoing authentication requests
CAD	Deals with anything Certificate Authority related, both internal CA and external CA
RDD	KeyTalk's daemon traffic cop, it ensures inter daemon traffic gets properly regulated and enforces only valid traffic to get passed. Also keeps track of time
WebUI	Tracks any changes made through KeyTalks administration webUI

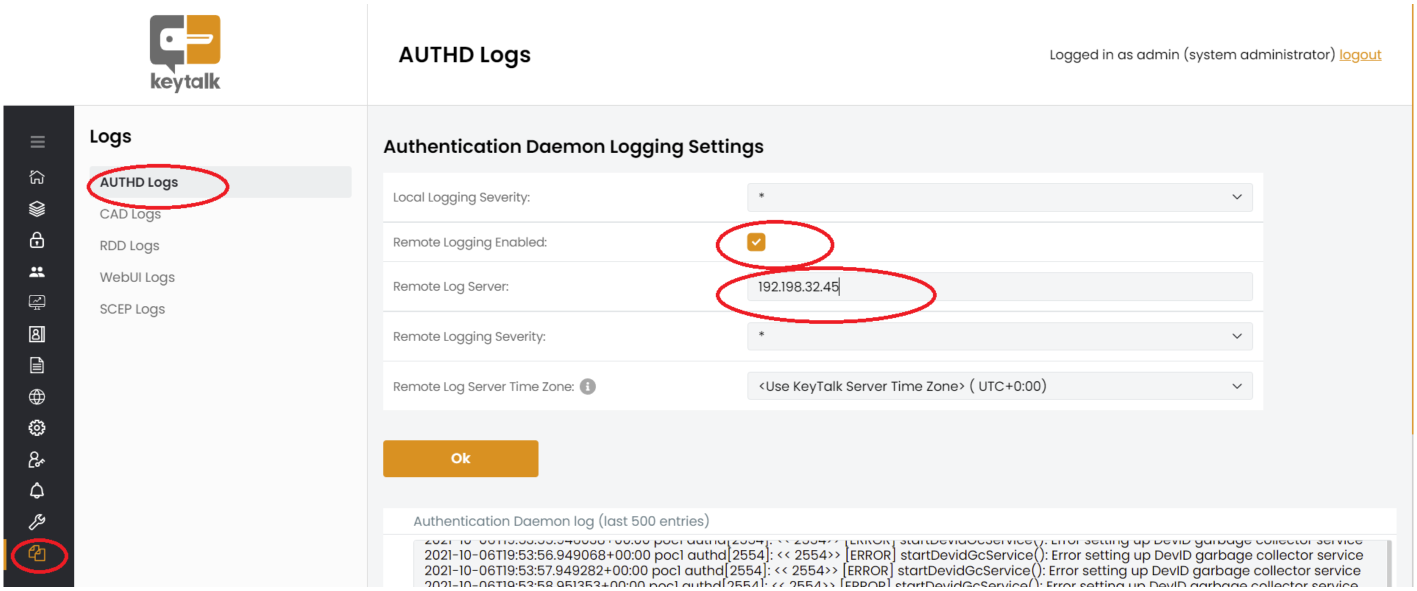
While the KeyTalk application server does keep local logfiles for immediate trouble shooting purposes by either the Network Admin or the Sys Admin, these local logfiles do rotate every 1500 lines, whereby only the last 4 rotated logs are only the youngest log is visible in the WebUI.

Depending on the chosen loglevel the local logfile rotation will happen sooner or later when the 500 line max is reached.

In order to provide proper support, monitoring and auditing, it is highly advisable for production purposes to configure the KeyTalk application server(s) to send their logfiles to a remote syslogserver.

From the remote syslog or SIEM server, logfiles can be concatenated, analyzed and report information can be directly extracted.
Optionally a CLI Admin may choose to set SNMP by installing the SNMP package for Ubuntu using the standard available SNMP package and configuration options.

With time being very important for the issuance of X.509 certificates, KeyTalk enforces the use of UTC across its solution. However, the Network Admin or Sys Admin can set a time zone correction for logfile purposes when the rest of their network logfiles use for example local time only, this results in 2 timestamps in your logfiles.



AUTHD Logs

Logged in as admin (system administrator) logout

Authentication Daemon Logging Settings

Local Logging Severity:	*
Remote Logging Enabled:	☑
Remote Log Server:	192.198.32.45
Remote Logging Severity:	*
Remote Log Server Time Zone:	<Use KeyTalk Server Time Zone> (UTC+0:00)

Authentication Daemon log (last 500 entries)

```
2021-10-06T19:53:56.949068+00:00 poc1 authd[2554]: << 2554>> [ERROR] startDevidGcService(): Error setting up DevID garbage collector service
2021-10-06T19:53:57.949282+00:00 poc1 authd[2554]: << 2554>> [ERROR] startDevidGcService(): Error setting up DevID garbage collector service
2021-10-06T19:53:58.951353+00:00 poc1 authd[2554]: << 2554>> [ERROR] startDevidGcService(): Error setting up DevID garbage collector service
```

Port and format:

port 514, TCP or UDP
standard rsyslog format <PRIORITY><TIMESTAMP> <PROGRAM>[<PID>] MESSAGE
https://rsyslog-5-8-6-doc.neocities.org/rsyslog_conf_templates.html

22 KeyTalk agents, clientless and automated/manual enrolment

22.1 Why a client/API footprint?

KeyTalk primarily uses a client/API footprint in order to ensure security, compatibility, user-experience, and certificate-use-case feature support, such as key-and-certificate-roll over for S/MIME, across all end-points in need of certificates.

Manual enrolment is ofcourse supported using KeyTalk WebUI interface.

Other protocols such as SSH can be supported, just let us know what you need and why you need it, as we often find that the requested protocol does not meet the end-goal of the envisioned customer project.

For a “clientless” enrolment KeyTalk also supports MDM, whereby MDM integrations are added based on customer use-case, as well as SCEP (for Intune).

22.2 Client enrolment, clientless experience

KeyTalk generic agents are available through the regular app stores for Android and iOS and MacOSX
Simply have an end-user download these, send them your KeyTalk configuration file. The app will guide your user to obtain the certificate and key-pair

Similarly, the KeyTalk generic clients for Linux, Mac, Windows, Android and iOS, are available as stand-alone installers, allowing any enterprise to enroll the KeyTalk clients in any way they are used to already.

Additionally, for all its clients, KeyTalk provides the source code through GitHub, allowing customers to self-compile the client including the configuration file(s) for enrolment purposes, or ask KeyTalk to compile such clients with configuration file for you. GitHub access is provided to customer only.

Most admins are mostly concerned about the Windows KeyTalk client enrolment.
Since the KeyTalk client comes as an MSI installer the most commonly use push mechanism is using VBSCRIPT:

```
cscript /nologo MsiSilentInstall.vbs path\\to\\msi [ full-path\\to\\rccd [rccd-proxy-user rccd-proxypassword] ]
Example: msiexec /i KeyTalk.msi /qn RCCDPATH=http://keytalk.com/demo.rccd
```

A sample script for Windows enrolment is provided with the windows client download.

With the support of Kerberos on Windows devices, and renewal scripts running as a scheduled task, an end-user is given a client-less experience.

Using MDM such as Mobile-Iron or Intune creates a true clientless experience, as does the use of SCEP (for Intune)

22.3 Apple end-point device support iPhone/iPad/Mac

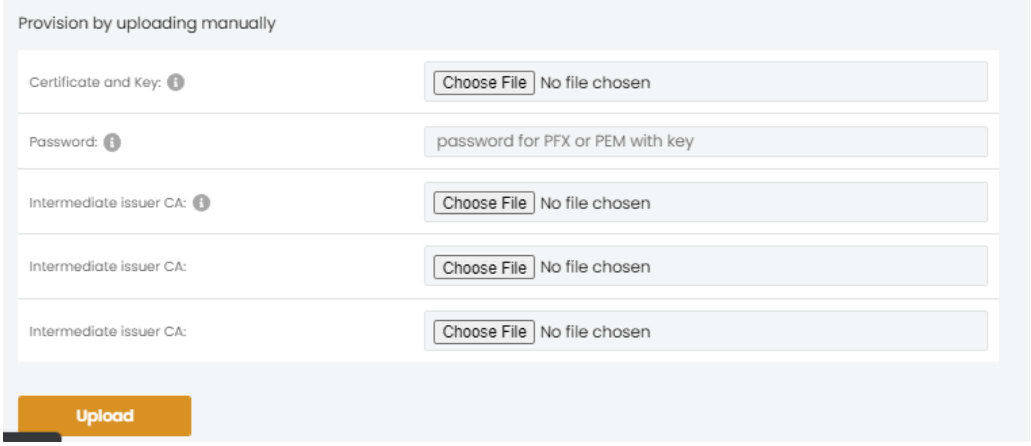
Apple enforces apps to make use of secure connections.

Since the KeyTalk by default makes use of its own internal CA, an Apple device will not trust an internal KeyTalk CA based TLS connection, without having the user either manually add the KeyTalk internal CA trust, or have it pushed by an MDM.

In order to end-user friendly support Apple based end-points, KeyTalk by default expects a globally trusted certificate issued by a trusted Certificate Service Provider (CSP) who's issuing CAs are pre-trusted on Apple devices.

To set a CSP certificate, you need to set it under: CERTIFICATES AND KEYS -> Public Trusted SSL

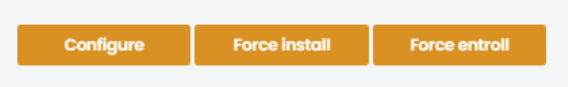
Either upload it manually:

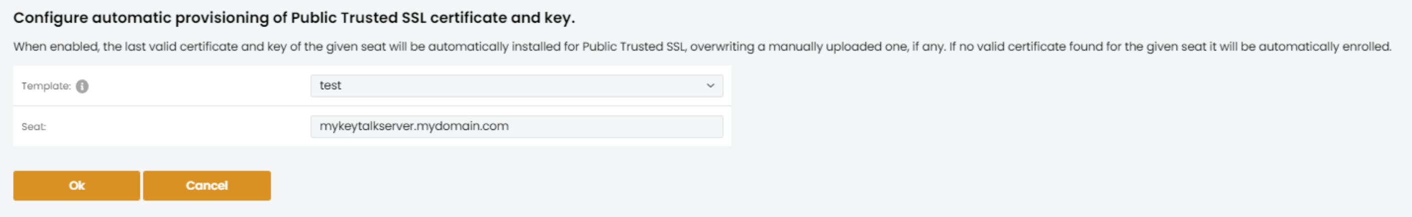


Provision by uploading manually

Certificate and Key:	Choose File No file chosen
Password:	password for PFX or PEM with key
Intermediate issuer CA:	Choose File No file chosen
Intermediate issuer CA:	Choose File No file chosen
Intermediate issuer CA:	Choose File No file chosen

Or configure automated enrolment and replacement by KeyTalk itself, based on a configured TEMPLATE set to issue certificates from a supported CSP.





Configure automatic provisioning of Public Trusted SSL certificate and key.

When enabled, the last valid certificate and key of the given seat will be automatically installed for Public Trusted SSL, overwriting a manually uploaded one, if any. If no valid certificate found for the given seat it will be automatically enrolled.

Template:	test
Seat:	mykeytalkserver.mydomain.com

NOTE: In a HA KeyTalk environment, this certificate must contain a SAN that contains all applicable FQDNs matching the FQDN of the individual KeyTalk servers, or the FQDN used by the Load Balancer.

NOTE: Though ports 8443 and 4443 are primarily meant for support of Apple Device, these can also be use for any of the other KeyTalk clients, ensuring you don't just rely on the KeyTalk internal CA.
Simply generate a KeyTalk config file (RCCD) using port 4443 if you want it used across all your end-points. Apple KeyTalk apps will auto convert port 443 to port 4443 and 8443

NOTE: If for whatever reason an internal domain needs to be supported, for which a trusted certificate cannot be issued, you can download the KeyTalk private CA generated client-server certificate and key from "Certificates and Keys" and upload it for Trusted Mobile SSL purposes. This does however require you to set the private CA (Primary and Communication CA certificate) as trusted on Mac and Android devices due to sandbox restrictions.

22.4 Mobile Device Management (MDM) support

KeyTalk provides functionality to push generated certificates and keys to one or multiple Mobile Device Management solutions.

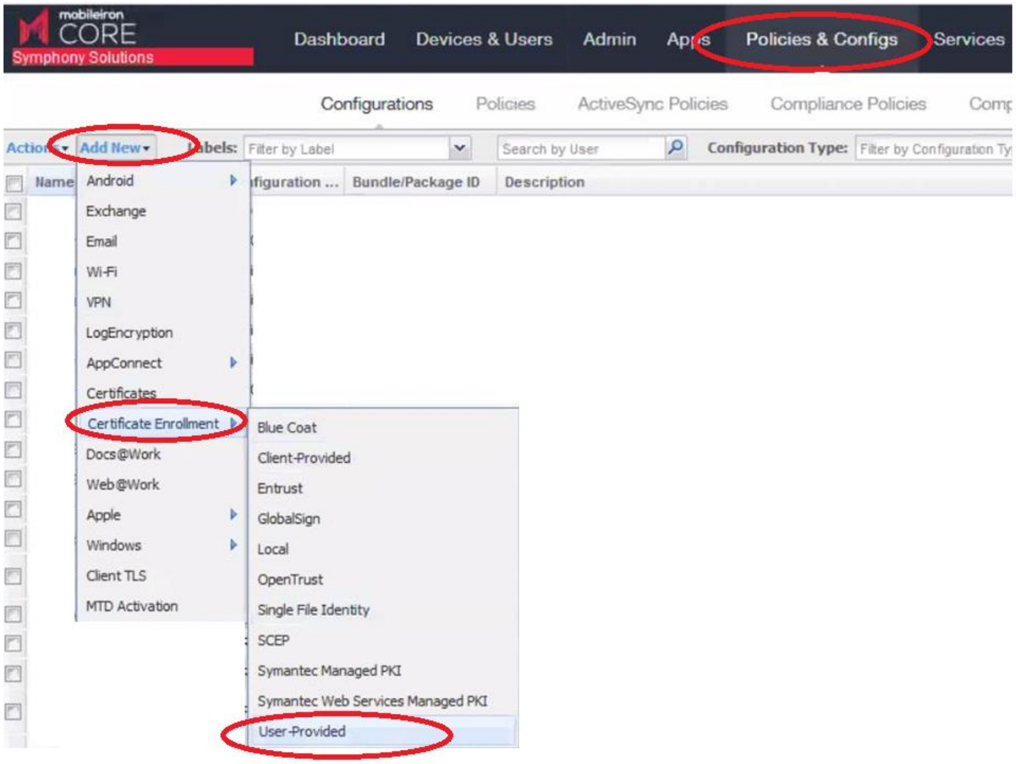
22.4.1 MobileIron Core

When a certificate and key is issued for a particular end-point or user, then the certificate and key can also be sent to MobileIron by means of MobileIron's "User Provided" Certificate Enrollment.

First ensure that MobileIron is configured for **User-Provided certificate enrollment**:

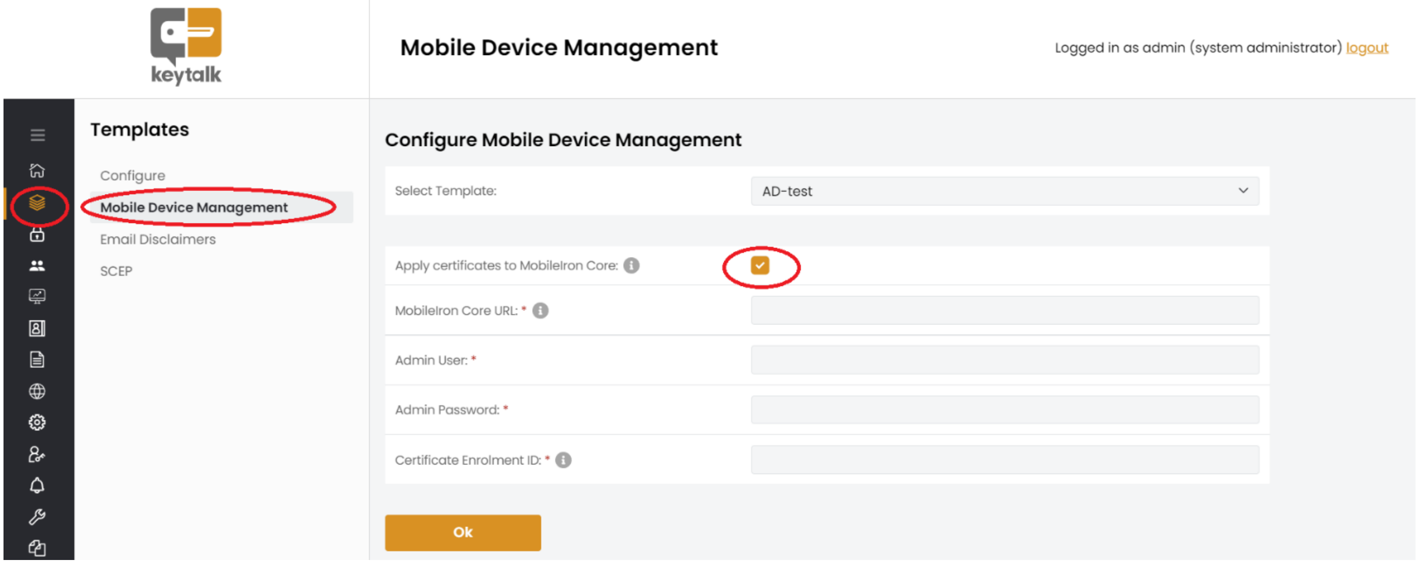
To specify the settings for a user-provided certificate enrolment setting:

1. Go to **Policies & Configs > Configurations** and click **Add New > Certificate Enrollment > User-Provided**.
2. Use the following guidelines to specify the settings:
   Name: Enter brief text that identifies this setting.
   Description: Enter additional text that clarifies the purpose of this setting.
   Display Name: Enter the name that will appear on the user portal where device users upload their certificates. This name also appears in Mobile@Work if Mobile@Work prompts the device user for a certificate's private key password.
   Require Password: This option requires the user to provide a password for the certificate's private key when uploading a certificate associated with this certificate enrollment setting.
   Important: Always require a password!!
   Del PrivKeys After: Select the number of days after a user-provided certificate is uploaded to MobileIron Core after which Core deletes the private key and, if provided, its password, from Core.
   The default is **None**, which means Core does not delete the private key and its password.
3. Click Save
4. Add the appropriate labels to the created User Provided certificate enrolment profile.



KeyTalk side the username used to issue the certificate to, must match the same username used in MobileIron.

Simply configure your configured TEMPLATE in the MobileIron target TEMPLATE:



Select Template: AD-test

Apply certificates to MobileIron Core: ☑

MobileIron Core URL: *

Admin User: *

Admin Password: *

Certificate Enrolment ID: *

Your Certificate EnrollmentID (CEID) is obtained by means of an API command towards your MobileIron MDM, using the following command:

```
curl -k -sS -u user:Password! -XPOST -H "Content-Type: application/json" 'https://[mymobileiron.mydomain.comp/api/v2/configuration/CE/USER_PROVIDED/all/getAllUserProvidedCertificateDetails' -d '{"userId": "[myadminuser]"}'
```

Returned **CEID** value (19) will look similar to:

```
{"results":[{"name":"Keytalk","id":19,"caProtocal":"USER_PROVIDED","rowType":"ADMIN","description":"Keytalk enrollment"}],"resultCount":1}keytalk@somename:~$
```

Pressing SAVE will result in KeyTalk validating:

- the SSL certificate trust of the HTTPS url (default trust are all valid Public Trusted Certificate Providers)
- the entered CEID
- validating the Admin User credentials

Should an error occur, kindly check the WEBUI log. (for more details use Report Problem and check VAR/LOG/KEYTALK)

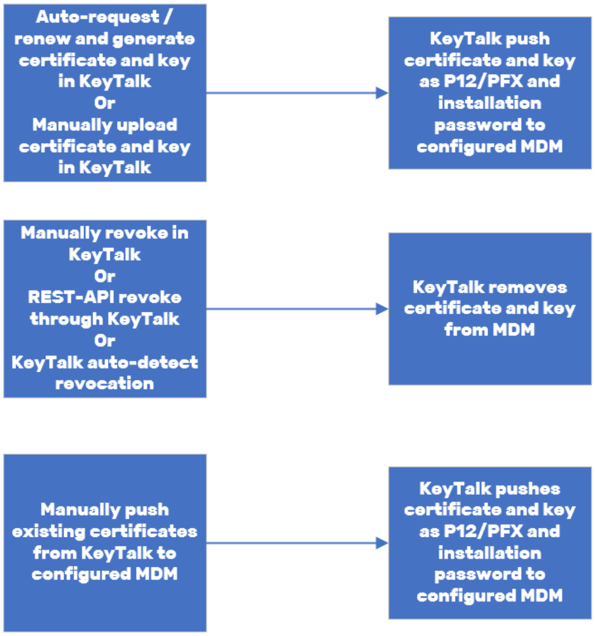
To test a certificate push to MobileIron, go to the SEAT of a user containing a proper S/MIME certificate and key, and select Push to MDM.
If an error occurs, kindly check the WEBUI log for more details (for more details use Report Problem and check VAR/LOG/KEYTALK)

While a full MobileIron Core user will work, MobileIron recommends using minimal authorization rights, which should be:

- ✓ API
- ✓ Managed certificates
- ✓ View certificates
- ✓ Manage configuration
- ✓ View configuration

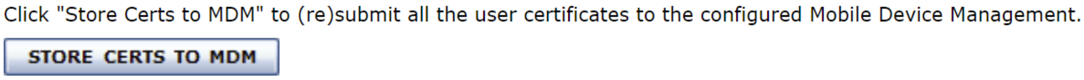
Should this not work, kindly contact your MobileIron vendor or specialist

KeyTalk certificate and key related functions to MobileIron flow:


Auto-request / renew and generate certificate and key in KeyTalk Or Manually upload certificate and key in KeyTalk
KeyTalk push certificate and key as P12/PFX and installation password to configured MDM
Manually revoke in KeyTalk Or REST-API revoke through KeyTalk Or KeyTalk auto-detect revocation
KeyTalk removes certificate and key from MDM
Manually push existing certificates from KeyTalk to configured MDM
KeyTalk pushes certificate and key as P12/PFX and installation password to configured MDM

While KeyTalk will automatically process new manually and automatically obtained certificates and keys, and as a result push them to MobileIron, an Admin can also opt to push already present certificates and keys from KeyTalk to the configured MobileIron Core.

Go to issued certificates under SEATS -> Select the certificate template -> generically push all certificates or select and individual user:


Click "Store Certs to MDM" to (re)submit all the user certificates to the configured Mobile Device Management.

22.4.2 Office 365 Intune

21.4.2.1 BETA Disclaimer

Microsoft's Graph API for Office 365 Intune is still in beta and deemed not yet generally available by Microsoft. As a result, while Microsoft keeps the status in beta, KeyTalk cannot guarantee that the implemented interface will continue to work as intended. Should the interface cease to work due to Microsoft making updates to the beta Graph API and/or its current implemented workflows, then KeyTalk will make updates accordingly in its next firmware release. In the event that this should happen, kindly do let our support department know.

22.4.2.2 Introduction

Microsoft's Office 365 Intune has 2 modes to deploy certificates:

- SCEP/NDES based (used for certificate based authentication unique to the device)
- Imported PKCS based (used for S/MIME)

KeyTalk has been certified by Microsoft as one of few companies to support **both** deployment methods:
https://docs.microsoft.com/en-us/mem/intune/protect/certificate-authority-add-scep-overview
https://docs.microsoft.com/en-us/mem/intune/protect/certificates-imported-pfx-configure#support-for-third-party-partners

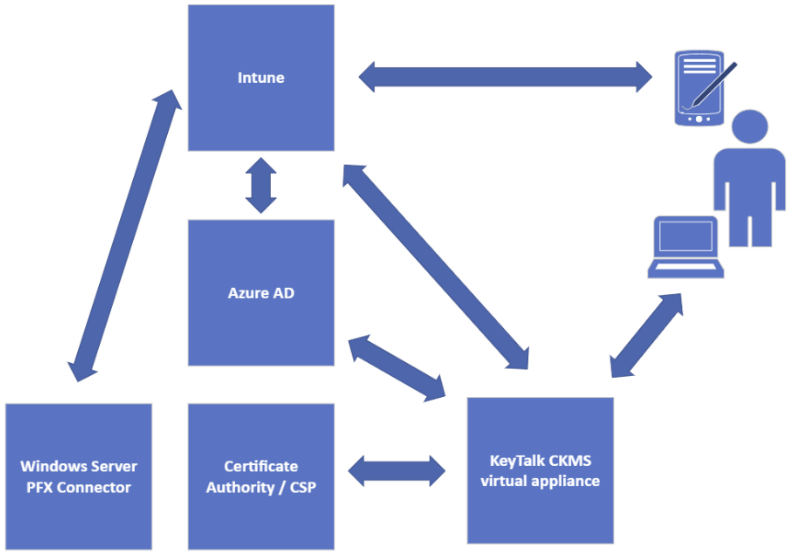
22.4.2.3 High level flow and requirements PKCS import PFX (ie S/MIME certificates)

Unlike some other MDM solutions, Microsoft's Office 365 Intune relies on the UPN of a user being used to deploy the certificate. So ensure your users authenticate with their UPN, or ensure that in the KeyTalk CKMS management interface for the Intune connector is configured to fetch a users UPN from the AD attribute.

A PFX contains a certificate and public-private keypair. Each PFX is protected with a password, which gets randomly generated by your KeyTalk CKMS.

To ensure a PFX file cannot be decrypted by someone with access to the Intune Management Console environment, the PFX installation password gets encrypted with a Intune enrolled device unique key.
This key in turn is encrypted using an a-symmetric key principle, whereby the enrolled device encrypts its unique key, and sends it over the Intune Management Console to the KeyTalk CKMS.
KeyTalk CKMS encrypts the PFX password with this received key, and sends it to the Intune environment, which sends it to the enrolled device, allowing the device to locally install the PFX using its decryption key.

Currently Microsoft relies solely on a Windows server to ensure the keypair, used to encrypt/decrypt the Intune enrolled device PFX installation password encryption key, is securely generated and stored.
This Windows server needs to be configured with several components as outlined in the next steps. Optionally an HSM can be configured with this Windows server, but this configuration will not be covered in this guide.


Intune
Azure AD
Windows Server PFX Connector
Certificate Authority / CSP
KeyTalk CKMS virtual appliance

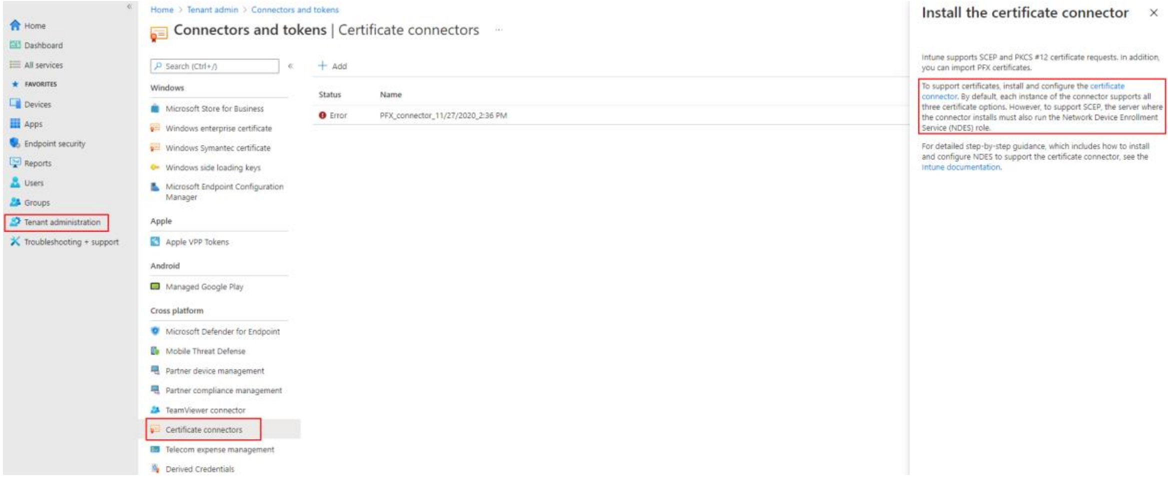
22.4.2.4 KeyTalk Office 365 Intune S/MIME deployment connection configuration steps

22.4.2.4.1 Set up the PFX Certificate Connector

Requirements:

- Windows Server with .NET 4.7.2 Framework or higher installed, and support for TLS 1.2 enabled
- The Microsoft Certificate Connector requires access to the same ports as detailed for managed devices, as found in Microsoft's device endpoint content (https://docs.microsoft.com/en-us/mem/intune/fundamentals/intune-endpoints#access-for-managed-devices)
- To allow the Certificate Connector to auto update: ensure the firewalls are open to allow the connector as running on your Windows server to contact autoupdate.msappproxy.net on port 443.

Step 1: Download the latest Certificate Connector using these steps:
Go to your Endpoint Manager using your Intune organizational account and select -> Tenant Administration -> Connectors and Tokens -> Certificate Connectors -> Add (optional) -> Download the certificate connector software from the provided URL.


Home
Dashboard
All services
FAVORITES
Devices
Apps
Endpoint security
Reports
Users
Groups
Tenant administration
Troubleshooting + support
Home > Tenant admin > Connectors and tokens
Search (Ctrl+/)
Windows
Microsoft Store for Business
Windows enterprise certificate
Windows Symantec certificate
Windows side loading keys
Microsoft Endpoint Configuration Manager
Apple
Apple VPP Tokens
Android
Managed Google Play
Cross platform
Microsoft Defender for Endpoint
Mobile Threat Defense
Partner device management
Partner compliance management
TeamViewer connector
Certificate connectors
Telecom expense management
Derived Credentials
+ Add

Connectors and tokens | Certificate connectors

Status	Name
Error	PFX_connector_11/27/2020_2:36 PM

Install the certificate connector

Intune supports SCEP and PKCS #12 certificate requests. In addition, you can import PFX certificates.

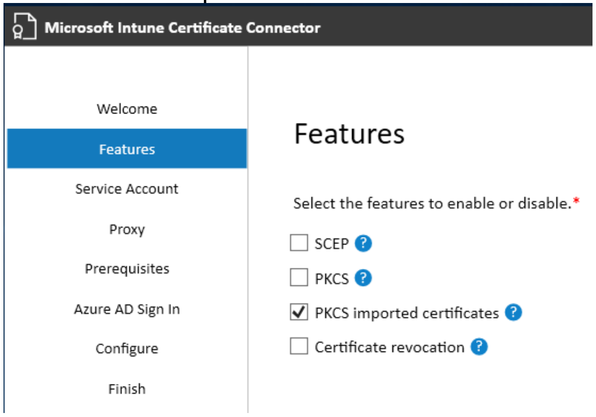
To support certificates, install and configure the certificate connector. By default, each instance of the connector supports all three certificate options. However, to support SCEP, the server where the connector installs must also run the Network Device Enrollment Service (NDES) role.

For detailed step-by-step guidance, which includes how to install and configure NDES to support the certificate connector, see the Intune documentation.

Step 2: Ensure the proper ports are open (see Requirements)

Step 3: Install the Certificate Connector on your Windows Server as administrator, after installation select 'Configure'
Alternatively you can run the 'Certificate Connector for Microsoft Intune' app as administrator should you already had it installed.

Step 4: Select "PKCS imported certificates" in section Features



Features

Select the features to enable or disable.*

☐ SCEP
☐ PKCS
☑ PKCS imported certificates
☐ Certificate revocation

Step 5: Choose SYSTEM account or enter service user credentials under 'Domain account' in section Service Account

Step 6: Optional: Configure Proxy settings

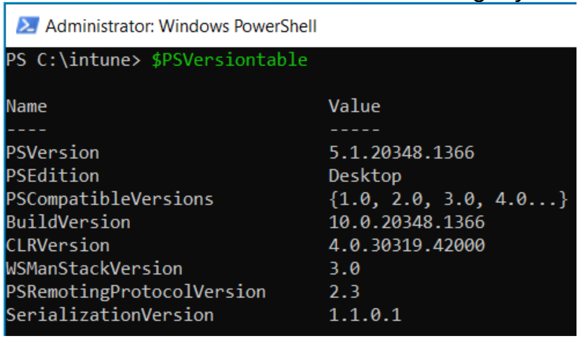
Step 7: Log in to Azure AD

Step 8: The Configuration step should happen automatically after which you can finish.
If the Configuration step fails, please verify that you ran the installer or 'Certificate Connector for Microsoft Intune' as administrator

Step 9: Download the IntunePfxImportHelper.zip as found here:
https://downloads.keytalk.com/downloads/tools/IntunePfxImportHelper.zip
Validation SHA2: 7FA9D28F9ED81C69174B34232D63ADF9896465066748FA2B91FA8E81441637AD
The latest version can also be found here: https://github.com/microsoft/Intune-Resource-Access/tree/develop/src/PFXImportPowershell

Step 10 Extract all files from the zip file on the target Windows server to a local folder

Step 11 Open PowerShell 5.1 **as administrator**, and navigate to the directory where the zip contents are extracted. PowerShell 7 **as administrator** uses slightly different installation commands as of step 13



```
PS C:\intune> $PSVersiontable

Name                           Value
----                           -----
PSVersion                      5.1.20348.1366
PSEdition                      Desktop
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
BuildVersion                   10.0.20348.1366
CLRVersion                     4.0.30319.42000
WSManStackVersion              3.0
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1
```

Step 12 Ensure the IntunePFXImportHelper files aren’t blocked by Windows, by performing the following command:

```
Get-ChildItem *.* -Recurse | Unblock-File
```

Afterwards reboot the server for the unblock to take effect.

Step 13 In the listed commands you can edit "Microsoft Software Key Storage Provider" to a different Key Storage Provider, "PFXEncryptionKey" to a different Key Pair Name and "C:\exportPFXEncKeyTest.pub" to a different file path)

Ensure you can execute the command unrestricted in Powershell. Perform the following command:

```
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope Process
```

Do not close Powershell to ensure the unrestricted execution policy remains in effect.

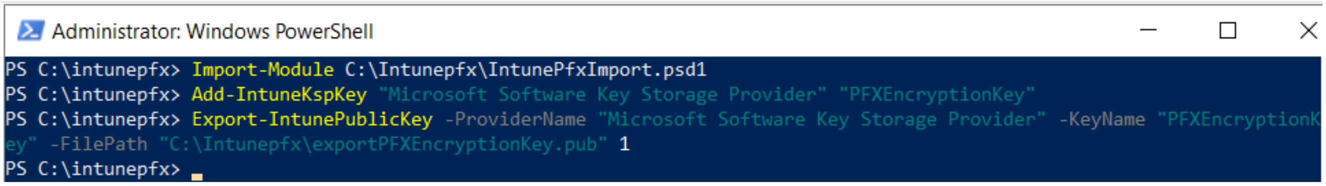
Run the following commands either for PS5 or PS7

PS5: `Import-Module <ABSOLUTE_DIR_PATH>\IntunePfxImport.psd1`

PS7: `Import-Module <ABSOLUTE_DIR_PATH>\IntunePfxImport.psd1 -UseWindowsPowerShell`

```
Add-IntuneKspKey "Microsoft Software Key Storage Provider" "PFXEncryptionKey"
```

```
Export-IntunePublicKey -ProviderName "Microsoft Software Key Storage Provider" -
KeyName "PFXEncryptionKey" -FilePath "C:\exportPFXEncryptionKey.pub" 1
```



```
PS C:\intunepfx> Import-Module C:\Intunepfx\IntunePfxImport.psd1
PS C:\intunepfx> Add-IntuneKspKey "Microsoft Software Key Storage Provider" "PFXEncryptionKey"
PS C:\intunepfx> Export-IntunePublicKey -ProviderName "Microsoft Software Key Storage Provider" -KeyName "PFXEncryptionK
ey" -FilePath "C:\Intunepfx\exportPFXEncryptionKey.pub" 1
PS C:\intunepfx> _
```

Step 14 Note down / store the Key Storage Provider name, Key Name & Public Key when configuring KeyTalk for use with Intune

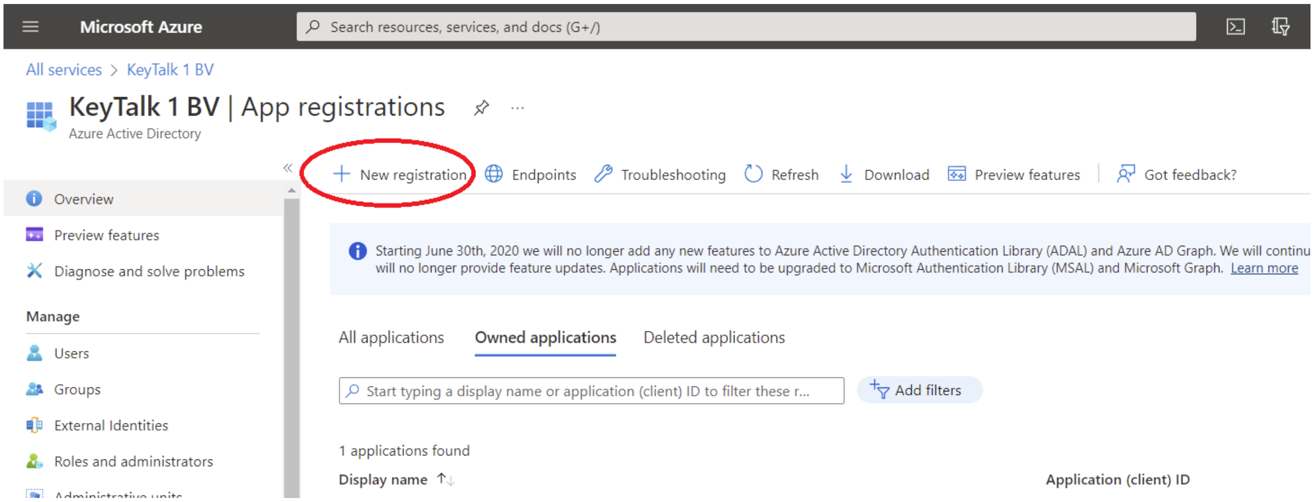
22.4.2.4.2 Set up Azure Enterprise Application rights

The Enterprise Application/tenant requires additional permissions to allow certs to be pushed to Intune by KeyTalk.

Step 1: Login to https://portal.azure.com/

Step 2: Select [Azure Active Directory](#)

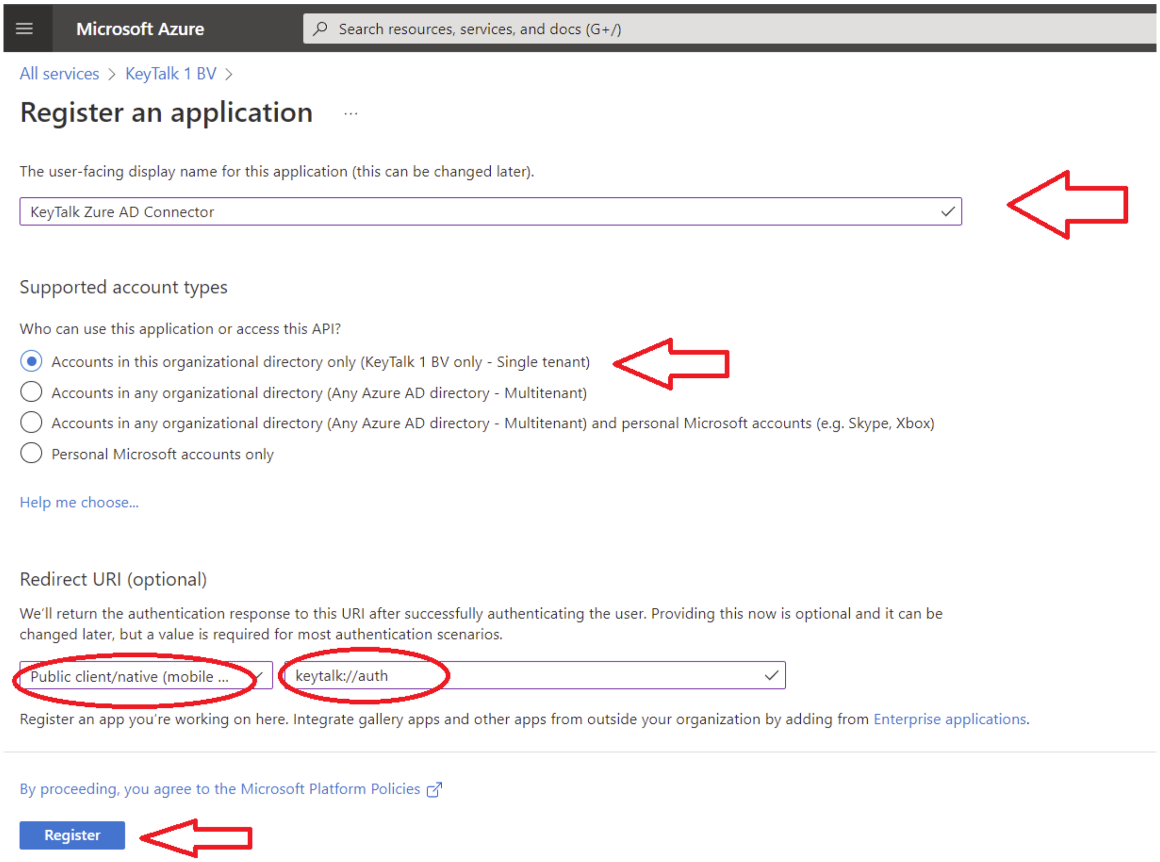
Step 3: Select [app registration](#) and select New registration:



Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continu will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more

1 applications found

Step 4: Configure the app registration


Microsoft Azure
Search resources, services, and docs (G+/)
All services > KeyTalk 1 BV >
Register an application
Help me choose...
Register

The user-facing display name for this application (this can be changed later).

KeyTalk Zure AD Connector

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (KeyTalk 1 BV only - Single tenant)
○ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
○ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
○ Personal Microsoft accounts only

Redirect URI (optional)

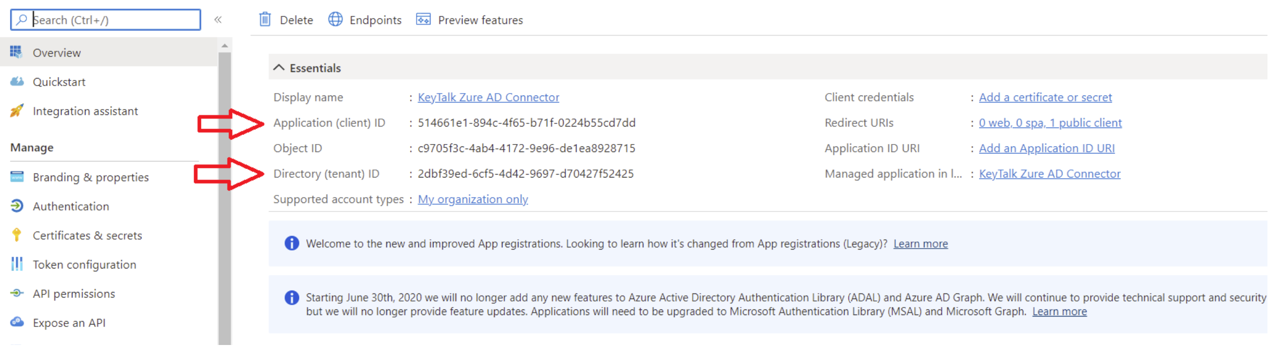
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Public client/native (mobile ... | keytalk://auth

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from Enterprise applications.

By proceeding, you agree to the Microsoft Platform Policies

Step 5: Write down the client and tenant ID



Display name : KeyTalk Zure AD Connector

Application (client) ID : 514661e1-894c-4f65-b71f-0224b55cd7dd

Object ID : c9705f3c-4ab4-4172-9e96-de1ea8928715

Directory (tenant) ID : 2dbf39ed-6cf5-4d42-9697-d70427f52425

Supported account types : My organization only

Client credentials : Add a certificate or secret

Redirect URIs : 0 web, 0 spa, 1 public client

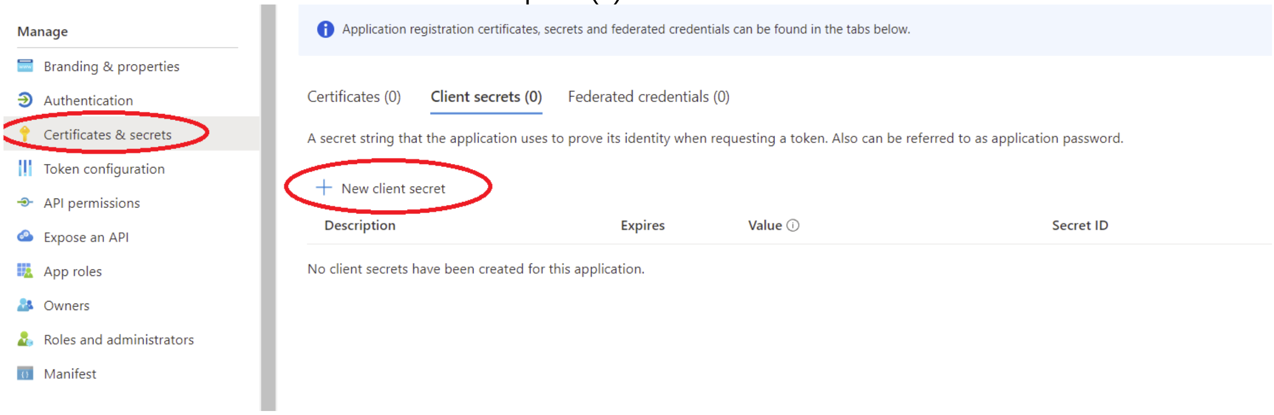
Application ID URI : Add an Application ID URI

Managed application in l... : KeyTalk Zure AD Connector

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? Learn more

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more

Step 6: Configure a client secret and note the set client secret and expiration date (!!) and set a reminder to set the new client secret before it expires (!!)

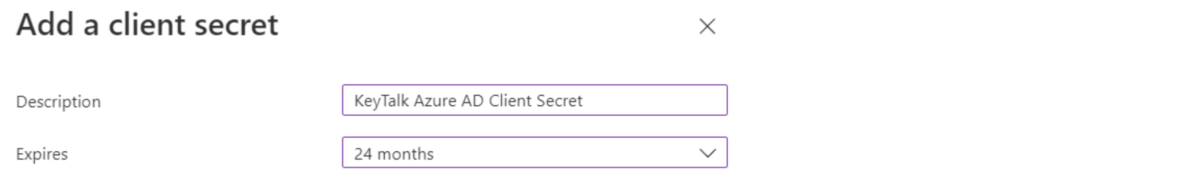


Application registration certificates, secrets and federated credentials can be found in the tabs below.

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

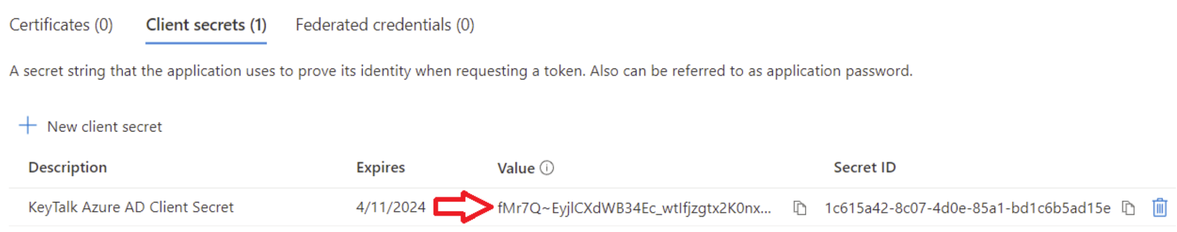
Description	Expires	Value	Secret ID

No client secrets have been created for this application.



Description: KeyTalk Azure AD Client Secret

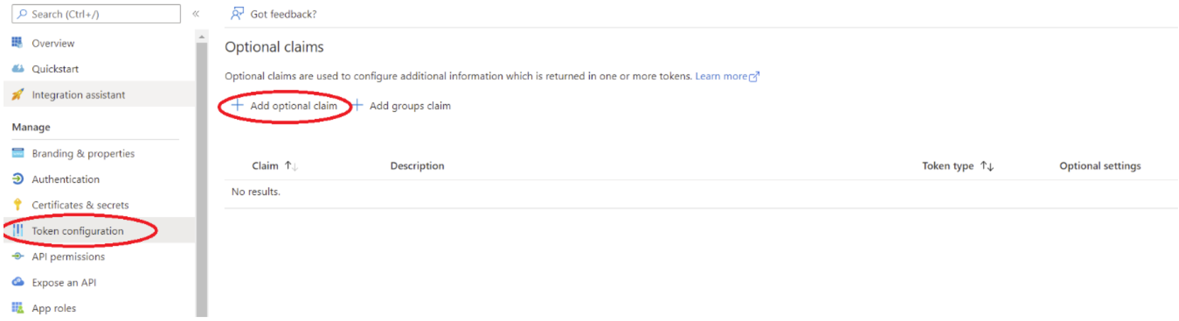
Expires: 24 months



A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

Description	Expires	Value	Secret ID
KeyTalk Azure AD Client Secret	4/11/2024	fMr7Q~EyjlCXdWB34Ec_wtIfjzgtx2K0nx...	1c615a42-8c07-4d0e-85a1-bd1c6b5ad15e

Step 7: Configure an ACCESS token claim

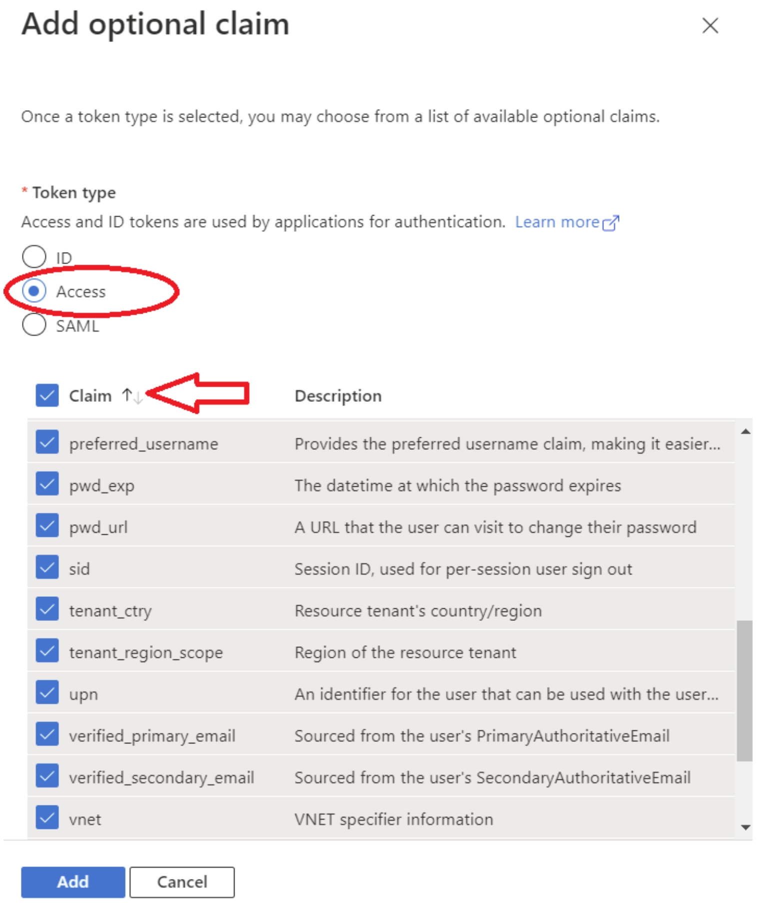


Optional claims

Optional claims are used to configure additional information which is returned in one or more tokens. Learn more

Claim	Description	Token type	Optional settings

No results.



Add optional claim

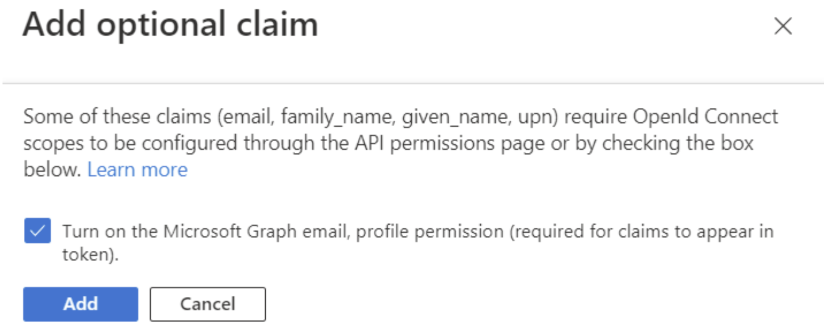
Once a token type is selected, you may choose from a list of available optional claims.

* Token type

Access and ID tokens are used by applications for authentication. Learn more

○ ID
● Access
○ SAML

☑ Claim ↑↓	Description
☑ preferred_username	Provides the preferred username claim, making it easier...
☑ pwd_exp	The datetime at which the password expires
☑ pwd_url	A URL that the user can visit to change their password
☑ sid	Session ID, used for per-session user sign out
☑ tenant_ctry	Resource tenant's country/region
☑ tenant_region_scope	Region of the resource tenant
☑ upn	An identifier for the user that can be used with the user...
☑ verified_primary_email	Sourced from the user's PrimaryAuthoritativeEmail
☑ verified_secondary_email	Sourced from the user's SecondaryAuthoritativeEmail
☑ vnet	VNET specifier information

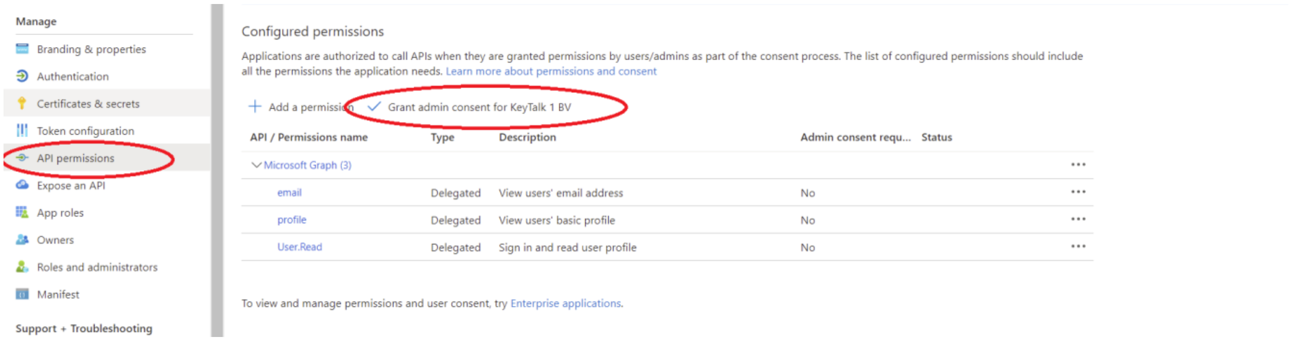


Add optional claim

Some of these claims (email, family_name, given_name, upn) require OpenId Connect scopes to be configured through the API permissions page or by checking the box below. Learn more

☑ Turn on the Microsoft Graph email, profile permission (required for claims to appear in token).

Step 7: Configure API permissions, Grant Admin consent:

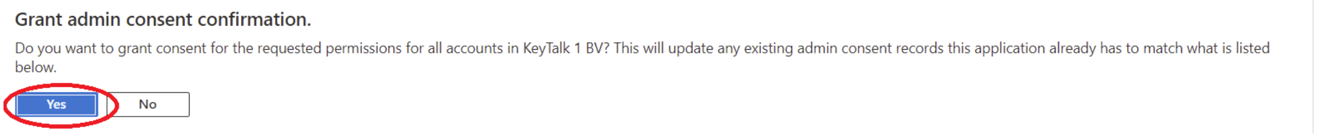


Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

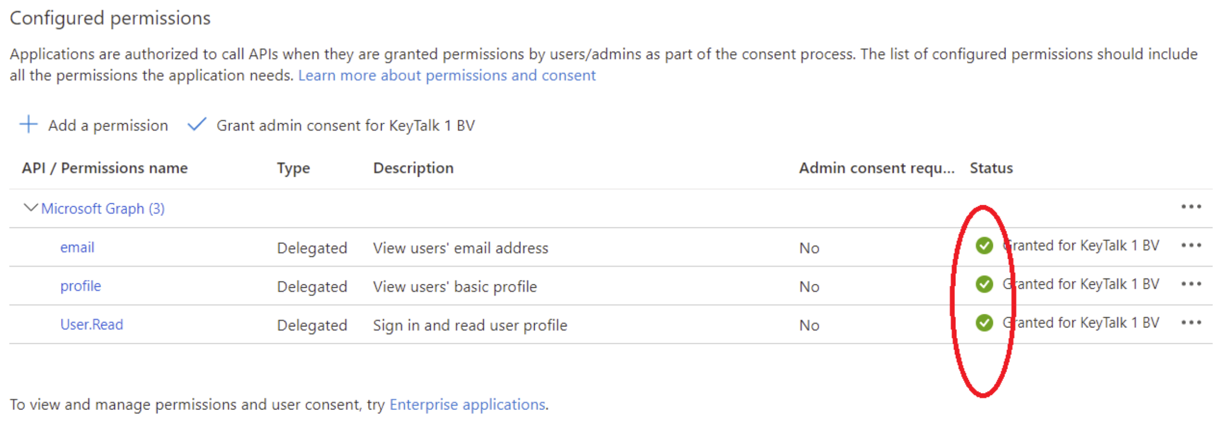
API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (3)				
email	Delegated	View users' email address	No	
profile	Delegated	View users' basic profile	No	
User.Read	Delegated	Sign in and read user profile	No	

To view and manage permissions and user consent, try Enterprise applications.



Grant admin consent confirmation.

Do you want to grant consent for the requested permissions for all accounts in KeyTalk 1 BV? This will update any existing admin consent records this application already has to match what is listed below.



Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

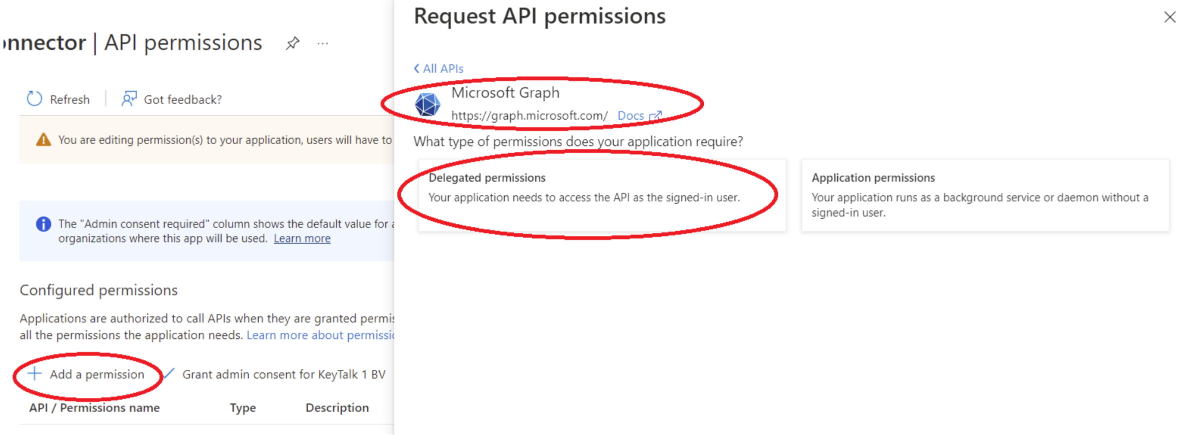
API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (3)				
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV

To view and manage permissions and user consent, try Enterprise applications.

Step 8: Add Microsoft Graph API permissions:

Select (API) Permissions and add at least:

- Microsoft Graph - DeviceManagementConfiguration.ReadWrite.All – Delegated
- Microsoft Graph - DeviceManagementConfiguration.ReadWrite.All – Application
- Intune - pfx_cert_provider - Application



You are editing permission(s) to your application, users will have to

The "Admin consent required" column shows the default value for a organizations where this app will be used. Learn more

Configured permissions

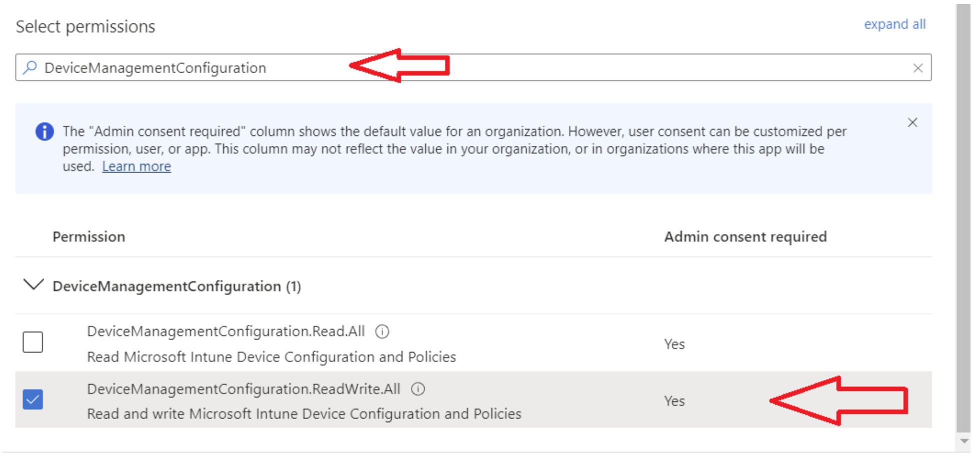
Applications are authorized to call APIs when they are granted permi all the permissions the application needs. Learn more about permissi

Request API permissions

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

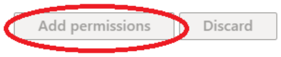


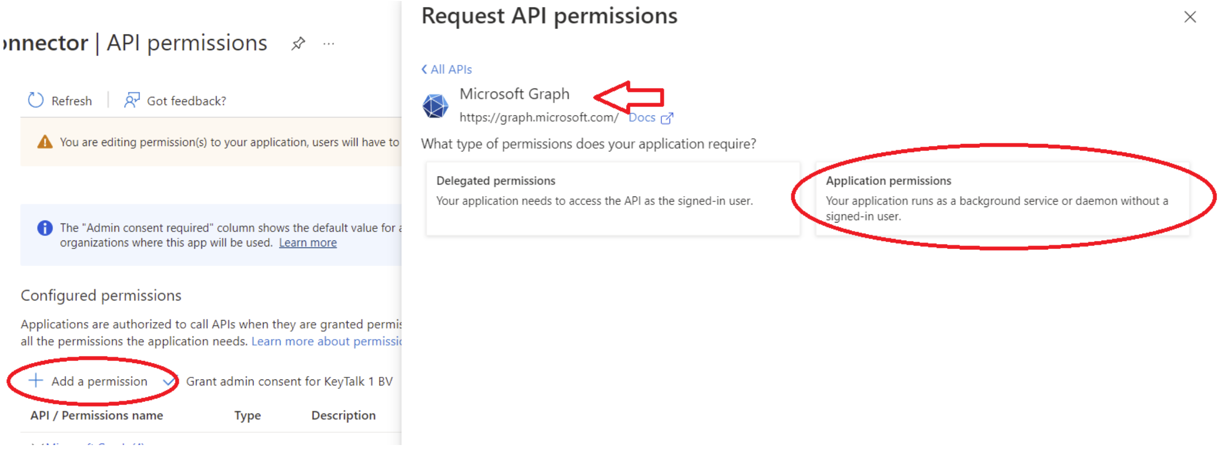
Select permissions

DeviceManagementConfiguration

The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. Learn more

Permission	Admin consent required
DeviceManagementConfiguration (1)	
□ DeviceManagementConfiguration.Read.All Read Microsoft Intune Device Configuration and Policies	Yes
☑ DeviceManagementConfiguration.ReadWrite.All Read and write Microsoft Intune Device Configuration and Policies	Yes





onnector | API permissions

Refresh | Got feedback?

You are editing permission(s) to your application, users will have to

The "Admin consent required" column shows the default value for a organizations where this app will be used. Learn more

Configured permissions

Applications are authorized to call APIs when they are granted permis all the permissions the application needs. Learn more about permissio

+ Add a permission ✓ Grant admin consent for KeyTalk 1 BV

API / Permissions name	Type	Description

Request API permissions

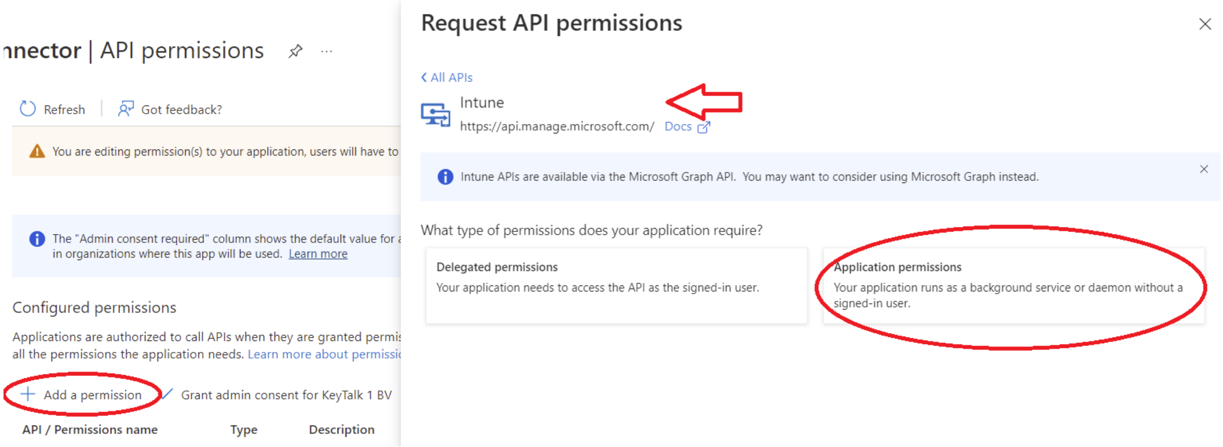
< All APIs

Microsoft Graph
https://graph.microsoft.com/ Docs

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.



nnector | API permissions

Refresh | Got feedback?

You are editing permission(s) to your application, users will have to

The "Admin consent required" column shows the default value for a in organizations where this app will be used. Learn more

Configured permissions

Applications are authorized to call APIs when they are granted permis all the permissions the application needs. Learn more about permissio

+ Add a permission ✓ Grant admin consent for KeyTalk 1 BV

API / Permissions name	Type	Description

Request API permissions

< All APIs

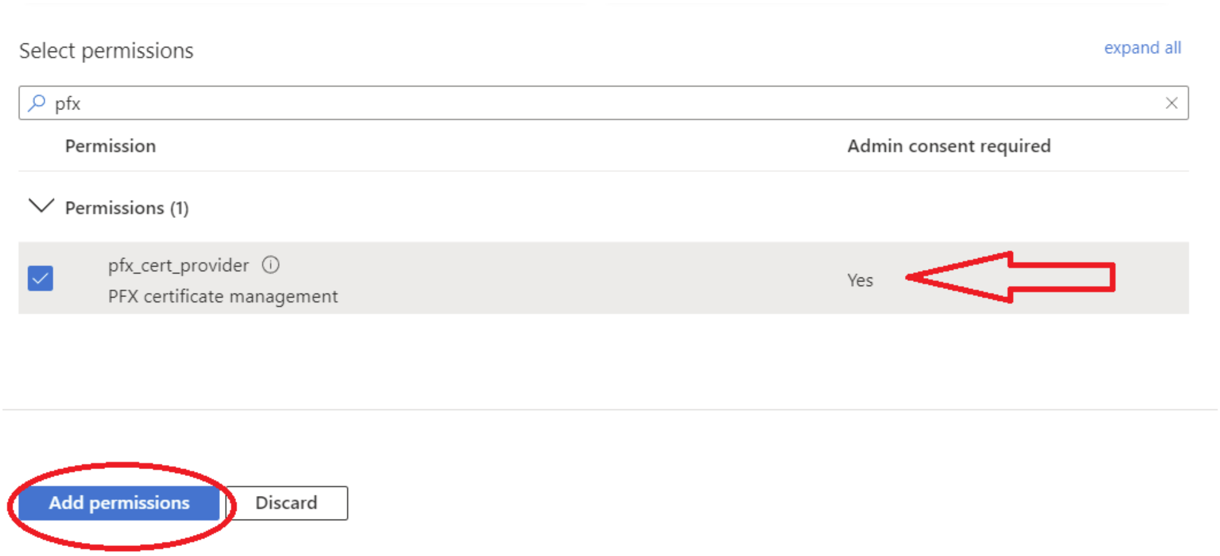
Intune
https://api.manage.microsoft.com/ Docs

Intune APIs are available via the Microsoft Graph API. You may want to consider using Microsoft Graph instead.

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.



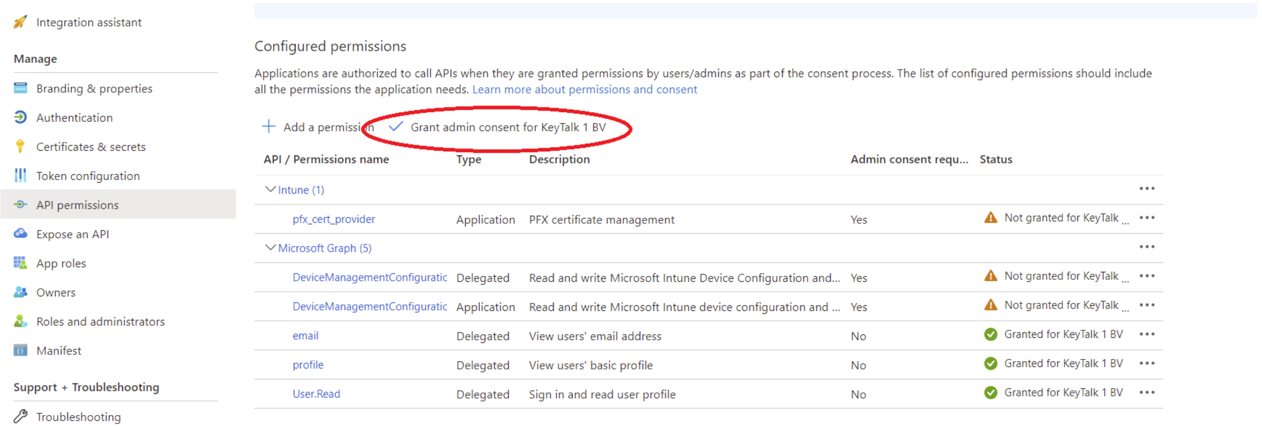
Select permissions

expand all

pfx

Permission	Admin consent required
Permissions (1)	
☑ pfx_cert_provider ⓘ PFX certificate management	Yes

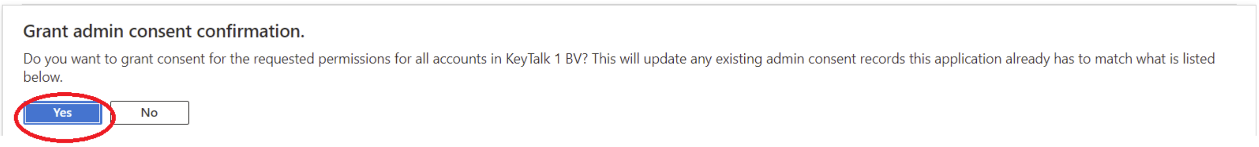
Step 9: Grant admin consent



Configured permissions

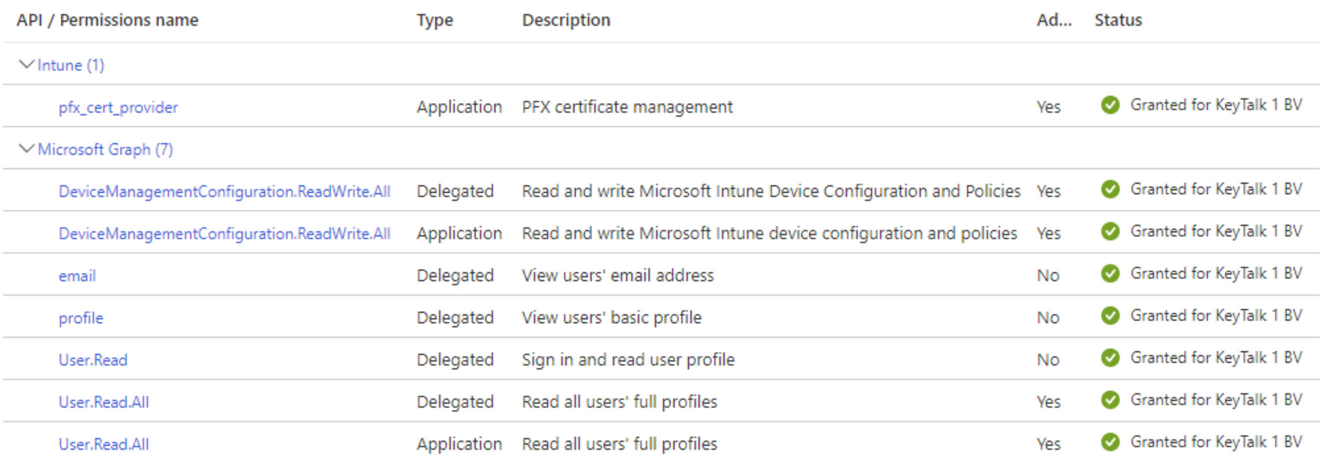
Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent requ...	Status
Intune (1)				
pfx_cert_provider	Application	PFX certificate management	Yes	⚠ Not granted for KeyTalk ...
Microsoft Graph (5)				
DeviceManagementConfiguratic	Delegated	Read and write Microsoft Intune Device Configuration and...	Yes	⚠ Not granted for KeyTalk ...
DeviceManagementConfiguratic	Application	Read and write Microsoft Intune device configuration and ...	Yes	⚠ Not granted for KeyTalk ...
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV



Grant admin consent confirmation.

Do you want to grant consent for the requested permissions for all accounts in KeyTalk 1 BV? This will update any existing admin consent records this application already has to match what is listed below.



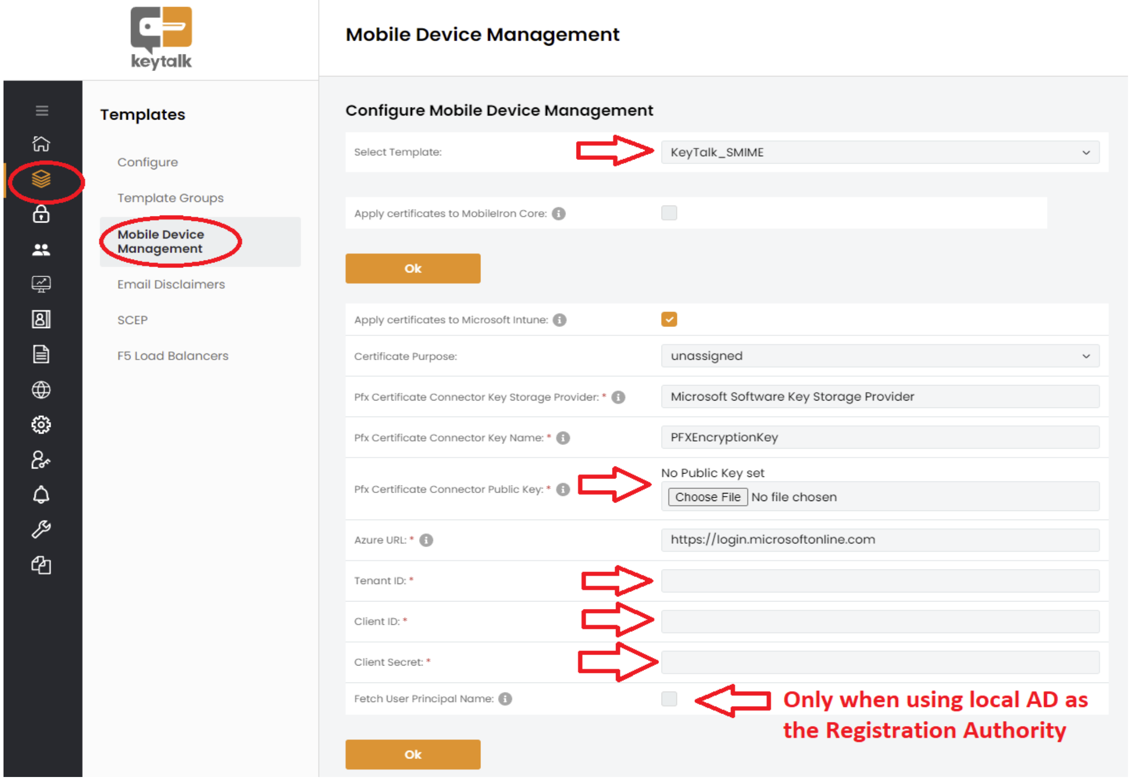
API / Permissions name	Type	Description	Ad...	Status
Intune (1)				
pfx_cert_provider	Application	PFX certificate management	Yes	✓ Granted for KeyTalk 1 BV
Microsoft Graph (7)				
DeviceManagementConfiguration.ReadWrite.All	Delegated	Read and write Microsoft Intune Device Configuration and Policies	Yes	✓ Granted for KeyTalk 1 BV
DeviceManagementConfiguration.ReadWrite.All	Application	Read and write Microsoft Intune device configuration and policies	Yes	✓ Granted for KeyTalk 1 BV
email	Delegated	View users' email address	No	✓ Granted for KeyTalk 1 BV
profile	Delegated	View users' basic profile	No	✓ Granted for KeyTalk 1 BV
User.Read	Delegated	Sign in and read user profile	No	✓ Granted for KeyTalk 1 BV
User.Read.All	Delegated	Read all users' full profiles	Yes	✓ Granted for KeyTalk 1 BV
User.Read.All	Application	Read all users' full profiles	Yes	✓ Granted for KeyTalk 1 BV

The above shows permissions for KeyTalk in regards to:

- *AzureAD Authentication user/account validation*
- *AzureAD account attribute reading (usually for UPN lookup and/or certificate attribute mapping)*
- *Writing PFX certificates to Intune*

22.4.2.4.3 Finalize KeyTalk configuration for Intune and push S/MIME PFX to Intune

Step 10: Finalize configuration in KeyTalk for Intune.



Mobile Device Management

Configure Mobile Device Management

Select Template: KeyTalk_SMIME

Apply certificates to MobileIron Core: □

Apply certificates to Microsoft Intune: ☑

Certificate Purpose: unassigned

Pfx Certificate Connector Key Storage Provider: * Microsoft Software Key Storage Provider

Pfx Certificate Connector Key Name: * PFXEncryptionKey

Pfx Certificate Connector Public Key: * No Public Key set — Choose File No file chosen

Azure URL: * https://login.microsoftonline.com

Tenant ID: *

Client ID: *

Client Secret: *

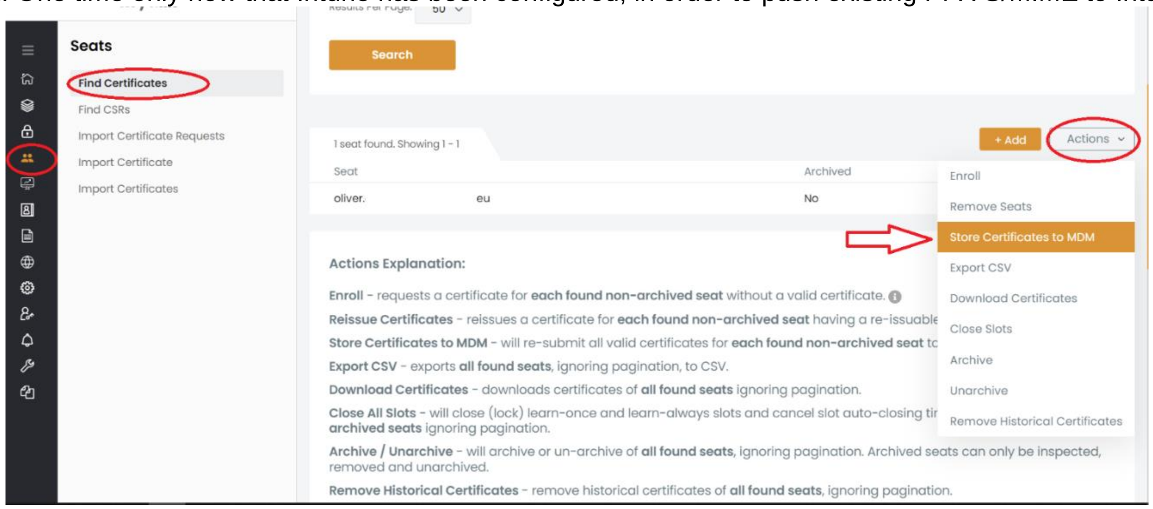
Fetch User Principal Name: □

Only when using local AD as the Registration Authority

NOTE: Intune REQUIRES the use of UPN values as the username.
So when using local AD ensure you enable Fetching User Principle Name, otherwise the assumption is that users are being authenticated using their UPN

Now that Intune with pushed PFX has been configured, every S/MIME that is created for a user will also be pushed to Intune upon successful creation.

Step 11: One time only now that Intune has been configured, in order to push existing PFX S/MIME to Intune:



1 seat found. Showing 1 - 1

Seat		Archived
oliver.	eu	No

Actions Explanation:

Enroll - requests a certificate for **each found non-archived seat** without a valid certificate.

Reissue Certificates - reissues a certificate for **each found non-archived seat** having a re-issuable

Store Certificates to MDM - will re-submit all valid certificates for **each found non-archived seat** tc

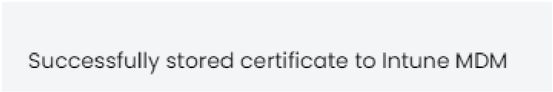
Export CSV - exports **all found seats**, ignoring pagination, to CSV.

Download Certificates - downloads certificates of **all found seats** ignoring pagination.

Close All Slots - will close (lock) learn-once and learn-always slots and cancel slot auto-closing tir **archived seats** ignoring pagination.

Archive / Unarchive - will archive or un-archive of **all found seats**, ignoring pagination. Archived seats can only be inspected, removed and unarchived.

Remove Historical Certificates - remove historical certificates of **all found seats**, ignoring pagination.

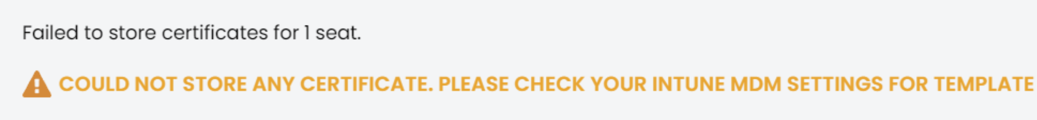


Successfully stored certificate to Intune MDM

<u>WebUI log sample of a successful upload:</u>

```
2023-02-17T15:08:43.191353+00:00 keytalk webui devid[2268]: << 2268>> [DEBUG] uploadCertificateToIntune(): Successfully
uploaded certificate to Microsoft Intune. Certificate purpose 'unassigned', certificate fingerprint:
ef740034ead3495718e940e58ddd847c3dec119d, user 'm.vandersman@keytalk.com', service 'KeyTalk_SMIME', Pfx encryption
OpenSSL v1 (3DES-CBC, legacy).
```

NOTE: When Intune ALREADY received the PFX, it will return an error message



Failed to store certificates for 1 seat.

⚠ COULD NOT STORE ANY CERTIFICATE. PLEASE CHECK YOUR INTUNE MDM SETTINGS FOR TEMPLATE

WebUI log sample of an unsuccessful upload:

```
2023-02-17T15:14:06.329260+00:00 keytalk webui devid[2268]: << 2268>> [ERROR] requestPostWithJson(): POST call got
HTTP response code '400' and curl msg: . URL: 'https://graph.microsoft.com/beta/deviceManagement/userPfxCertificates'
2023-02-17T15:14:06.332065+00:00 keytalk webui devid[2268]: << 2268>> [ERROR] uploadCertificateToIntune(): Failed to
upload Certificate to Microsoft Intune. Certificate purpose 'unassigned', certificate fingerprint:
ef740034ead3495718e940e58ddd847c3dec119d, user 'm.vandersman@keytalk.com', service 'KeyTalk_SMIME', Pfx encryption
OpenSSL v1 (3DES-CBC, legacy).. Error received from Graph API
```

When Intune has been configured correctly certificates and keys should now be submitted to Intune when a user authenticates with his/her (Azure) AD credentials, using the KeyTalk client (on for example Windows). Alternatively, a certificate and key can be submitted to Intune via the DEVID USERS WebUI for one or more users with active certificates, using the manual button “enroll to MDM”

22.4.2.4.4 Validating if Intune received the PFX certificate

KeyTalk server logs will prove that a PFX was successfully pushed to Intune.
Regretfully Microsoft Intune does not provide any UI means to validate that the PFX indeed resides in Intune, ready to be pushed by Intune to an enrolled device with the proper profile.

To check which certificates (and private keys) are available in Intune, the following project:
https://github.com/microsoft/Intune-Resource-Access/tree/develop/src/PFXImportPowershell

Get PFX Certificate Example

1) Get-PfxCertificates (Specific records)
```
Get-IntuneUserPfxCertificate -UserThumbprintList <UserThumbprintObjs>
```

2) Get-PfxCertificates (Specific users)
```
Get-IntuneUserPfxCertificate -UserList "<UserUPN>"
```

3) Get-PfxCertificates (All records)
```
Get-IntuneUserPfxCertificate
```

22.4.2.5 KeyTalk Office 365 Intune SCEP deployment connection configuration steps

KeyTalk's Intune SCEP integration relies on: https://docs.microsoft.com/en-us/mem/intune/protect/certificate-authority-add-scep-overview

22.4.2.5.1 *Configure your KeyTalk certificate TEMPLATE*

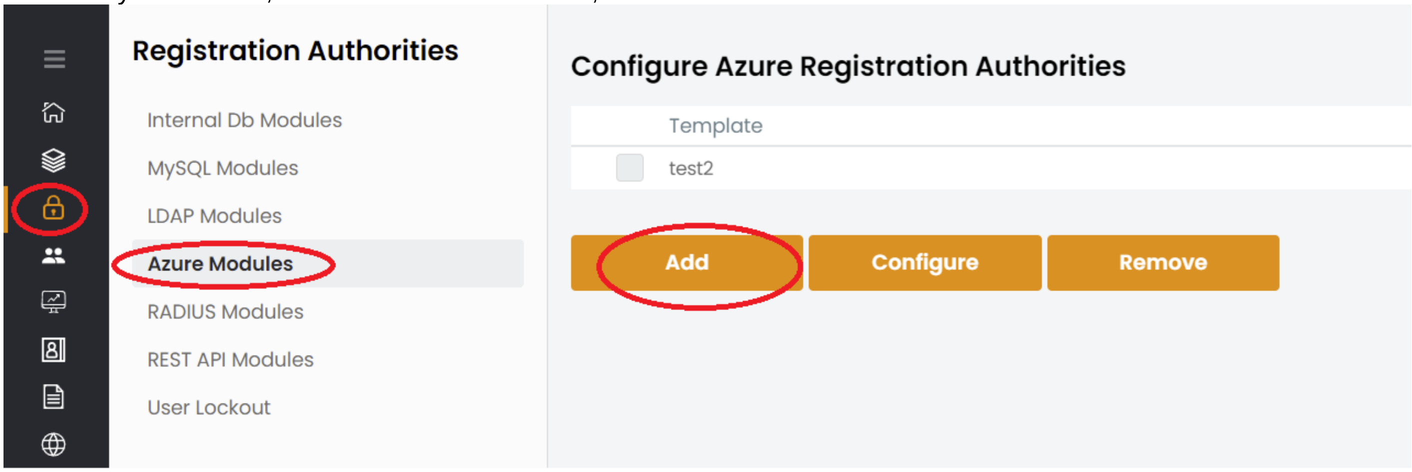
Similar to other certificate issuance template, a certificate template needs to be configured, which will be dedicated for use with Intune SCEP.
When SCEP is configured, the template is not usable for end-point KeyTalk client/agent based or REST-API fetching, it will solely allow SCEP based calls coming directly from an end-point.

Ensure your certificate template is configured to issue Extended Key Usage client-authentication certificates when using the KeyTalk private CA, or ensure that the connected external CA source allows for the issuance of client-authentication certificates.

Note: SCEP certificates **require** a Subject Alternative Name value, usually DNS or UPN, or RFC822 Name/email.
So ensure that the connected CA source supports it.

22.4.2.5.2 *Connect your SCEP certificate template to an Azure Module RA connector*

Within KeyTalk CKMS, select AUTHENTICATION, and create an Azure Module connector



22.4.2.5.3 *Whitelist your Firewall rules for inbound SCEP ports*

Allow your Firewall for inbound TCP traffic from requesting devices to your KeyTalk CKMS instances / LoadBalancer for ports:

- 80
- 443

22.4.2.5.4 *Obtain your SCEP for Intune Registration Authority certificates*

A Registration Authority (RA) must be configured to allow KeyTalk to handle SCEP requests on behalf of a CA. The RA consists of three parts:

- Encryption certificate (Recipient)
- Signing certificate (Signer)
- Issuing CA certificate(s) (trust-chain)

The Recipient & Signer certificates must be issued directly from the Issuing CA and must include their respective private keys.

The Recipient certificate keyUsage must be set to only include either keyEncipherment or dataEncipherment.

The Signer certificate keyUsage must be set to only include digitalSignature.

The Issuing CA certificate must either be only that certificate, or include the certificate chain up-to-and-including the Root CA.

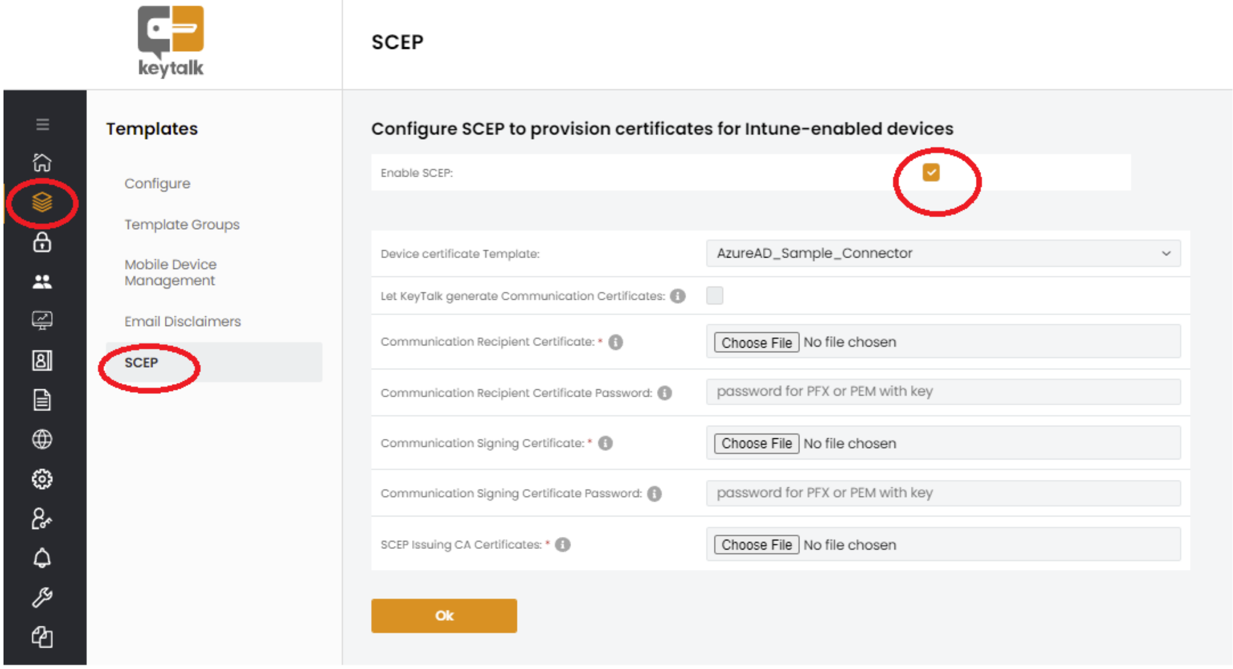
The Recipient & Signer certificate and keys can be uploaded either via PFX or PEM, password encryption optional.

The Issuing CA certificate(s chain) should be uploaded in PEM format.

Contact KeyTalk Support for more information on acquiring the RA certificates, or ask your CA provider on how to obtain these RA certificates for their CA platform

22.4.2.5.5 Configure KeyTalk as a SCEP server

KeyTalk CKMS currently only supports 1 Intune SCEP connector.
To activate it, go to:



Configure SCEP to provision certificates for Intune-enabled devices

Enable SCEP: ☑

Device certificate Template: AzureAD_Sample_Connector

Let KeyTalk generate Communication Certificates: □

Communication Recipient Certificate: * Choose File No file chosen

Communication Recipient Certificate Password: password for PFX or PEM with key

Communication Signing Certificate: * Choose File No file chosen

Communication Signing Certificate Password: password for PFX or PEM with key

SCEP Issuing CA Certificates: * Choose File No file chosen

Upload the appropriate certificates (and key) as provided by your CA-provider, or generated on your KeyTalk CKMS based on its internal CA.

22.4.2.5.6 Test the KeyTalk SCEP server with intune

The SCEP server should now be running. To test this fill out and enter the following URL (preferably on a client device):

http://[Your IP address]/intunescep/pkiclient.exe?operation=GetCACaps

Example: https://demo.keytalkdemo.com/intunescep/pkiclient.exe?operation=GetCACaps

If the server is configured correctly you should see a list of 3 items:

```
AES
POSTPKIOperation
SHA-256
```

If you cannot reach the page and/or you get a timeout, please ensure you have access to the KeyTalk server. If you see an error page like ‘HTTP Status 500 – Internal Server Error’, please ensure you have taken every numbered step in this section.

22.4.2.5.7 *Configure Office 365 for SCEP using Intune*

Before configuring Intune, the Issuing CA certificates chain or Issuing CA certificate must be prepared to be uploaded to the Intune Endpoint Manager admin center. For these steps you will need the Issuing CA file which you uploaded in section ‘RA Configuration’, in PEM format.

<u>Scenario A: the RA Configuration contains an Issuing CA certificates chain</u>

1. Open the PEM file with any text editor, you should see multiple sections of text encapsulated with the lines “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----".
2. Extract the bottom certificate by copying the entire last section including “-----BEGIN CERTIFICATE-----" & “-----END CERTIFICATE-----" into a new file*.
3. Save this file with the .cer extension. Example: “rootca.cer”.

**Note: Be sure to end the file with a single empty line after* -----END CERTIFICATE-----

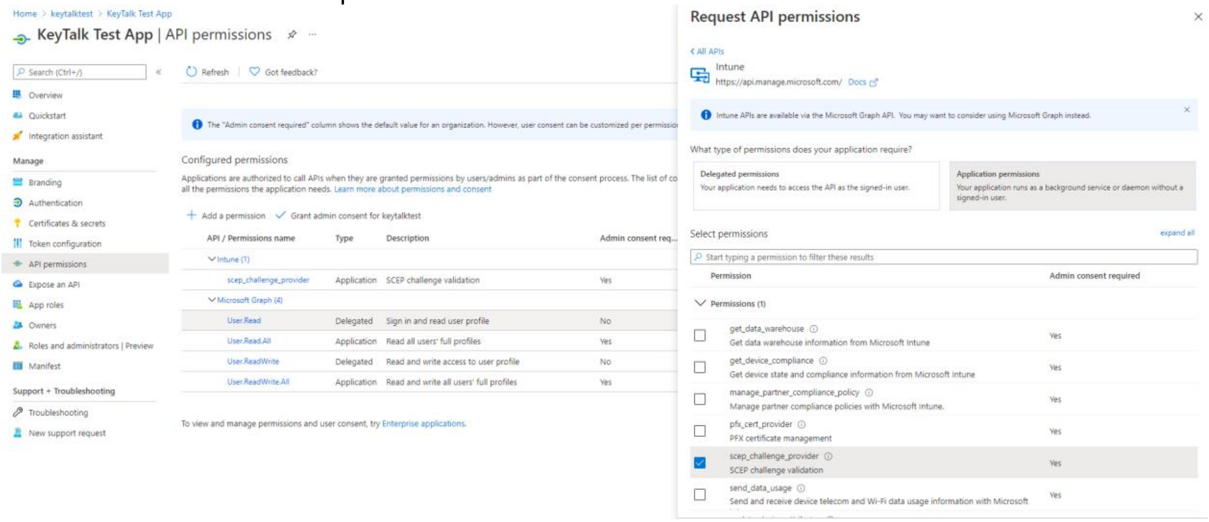
<u>Scenario B: the RA Configuration contains just the Issuing CA certificate</u>

1. Ensure you have a .pem or .cer file, for example: “issuingca.pem”.
2. If the file extension is .pem, simply rename it to .cer.

Configure Intune: App permissions

The Enterprise Application requires additional permissions.

Go to the Azure Portal -> Azure AD -> App registrations and select the Enterprise Application that is used to configure user authentication with KeyTalk and go to API Permissions.
Add permission for Intune -> Application permissions -> scep_challenge_provider.
Grant admin consent for the permission.



The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permissio

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of co all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent req...
Intune (1)			
scep_challenge_provider	Application	SCEP challenge validation	Yes
Microsoft Graph (4)			
User.Read	Delegated	Sign in and read user profile	No
User.Read.All	Application	Read all users' full profiles	Yes
User.ReadWrite	Delegated	Read and write access to user profile	No
User.ReadWrite.All	Application	Read and write all users' full profiles	Yes

To view and manage permissions and user consent, try Enterprise applications.

Request API permissions

< All APIs

Intune
https://api.manage.microsoft.com/ Docs

Intune APIs are available via the Microsoft Graph API. You may want to consider using Microsoft Graph instead.

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions

	Permission	Admin consent required
	Permissions (1)	
□	get_data_warehouse Get data warehouse information from Microsoft Intune	Yes
□	get_device_compliance Get device state and compliance information from Microsoft Intune	Yes
□	manage_partner_compliance_policy Manage partner compliance policies with Microsoft Intune.	Yes
□	pfx_cert_provider PFX certificate management	Yes
☑	scep_challenge_provider SCEP challenge validation	Yes
□	send_data_usage Send and receive device telecom and Wi-Fi data usage information with Microsoft	Yes

Configure Intune: Configuration profiles

Configuration Profiles must be configured to allow devices to request and install SCEP certificates. Generally, this involves 3 types of configuration profiles: Trusted Certificate, SCEP certificate and a usage profile, for example Wi-Fi or VPN Authentication. These profiles should be created in that order.

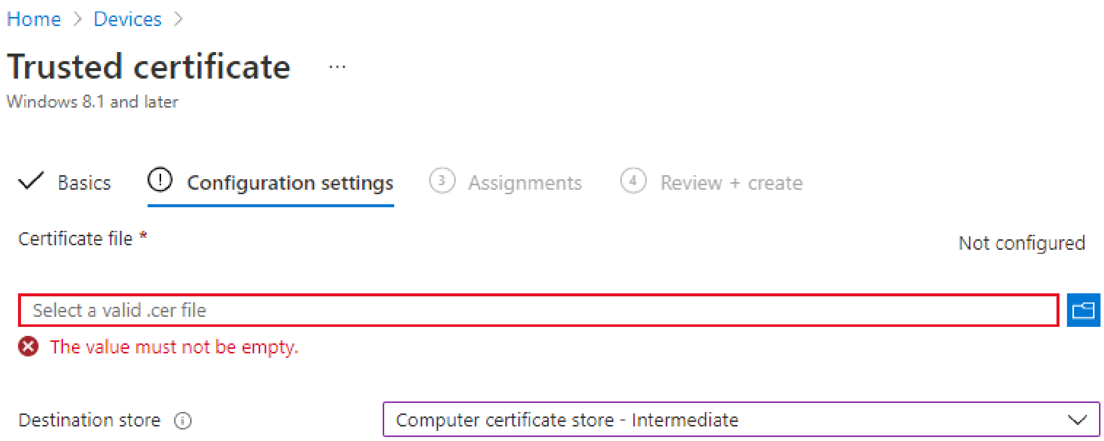
Go to the Endpoint Manager admin center -> Devices -> Configuration profiles.

Configure Intune: Trusted certificate profile

First create a Trusted certificate profile. This profile will install the trusted certificate for each platform* you support.
**Note: For Windows choose the platform “Windows 8.1 and later”.*

- Select Create profile, choosing the platform and Profile type ‘Trusted certificate’.
- ‘Step 2. Configuration setting’ depends on your RA Configuration Issuing CA certificate(s).
- Upload the .cer file you have prepared in the ‘Configure Intune: Prepare the Trusted certificate’ step.

- *Windows only*: For an RA configuration with Issuing CA chain, choose Destination Store ‘Computer certificate store – Root’. For Issuing CA only, choose ‘Computer certificate store – Intermediate’
- Step 3: Assign to/exclude specific Device groups and/or choose ‘Add all devices’.



Certificate file * Not configured

Select a valid .cer file

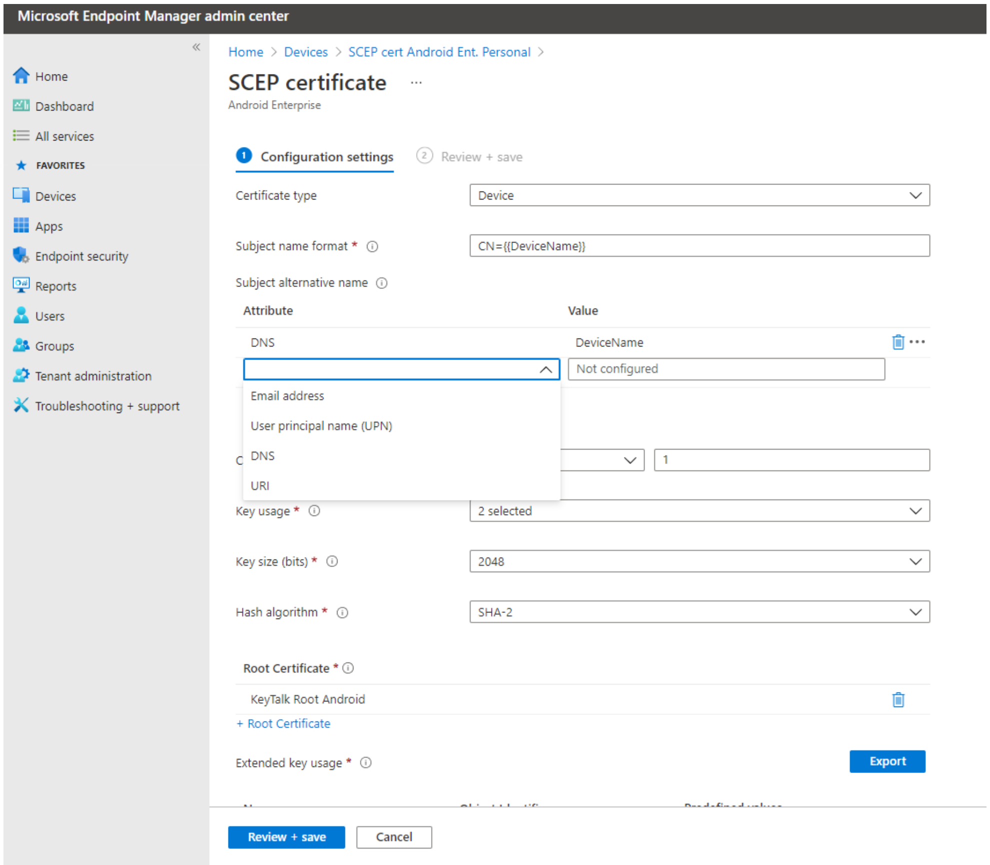
The value must not be empty.

Destination store: Computer certificate store - Intermediate

Configure Intune: SCEP profile

Second create a SCEP profile.

- Select Create profile, choosing the desired platform and Profile type ‘SCEP certificate’.
- Configure the certificate settings so they correctly match your CA profile settings where possible.
- Select the Trusted certificate profile you configured as the ‘Root certificate’*
  **Note: Do this regardless of whether it truly is a Root Certificate.*
- Add your available SCEP Server URLs in the following pattern
  [http/https]://[IP or address]:[port]/intunescep
  Examples: http://demo.keytalkdemo.com:19580/intunescep OR https://demo.keytalkdemo.com:19443/intunescep
- Assign to/exclude specific Device groups and/or choose ‘Add all devices’.
- Ensure a SAN value is defined to be used for SCEP requestt



Certificate type: Device

Subject name format *: CN={{DeviceName}}

Subject alternative name

Attribute	Value
DNS	DeviceName
	Not configured

Key usage *: 2 selected

Key size (bits) *: 2048

Hash algorithm *: SHA-2

Root Certificate *

KeyTalk Root Android

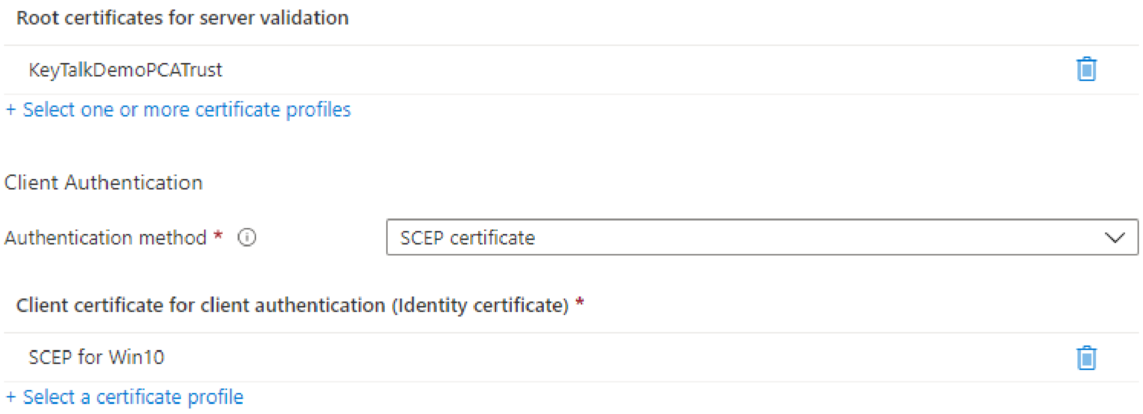
+ Root Certificate

Extended key usage *

Configure Intune: Remaining profiles

Create a profile for which the certificate should be used.
Windows Wi-Fi example, when setting the Configuration settings:

- Choose ‘SCEP certificate’ as the Authentication method.
- Select the correct SCEP profile under “Client certificate for client authentication (Identity certificate)”



Root certificates for server validation

KeyTalkDemoPCATrust

+ Select one or more certificate profiles

Client Authentication

Authentication method * : SCEP certificate

Client certificate for client authentication (Identity certificate) *

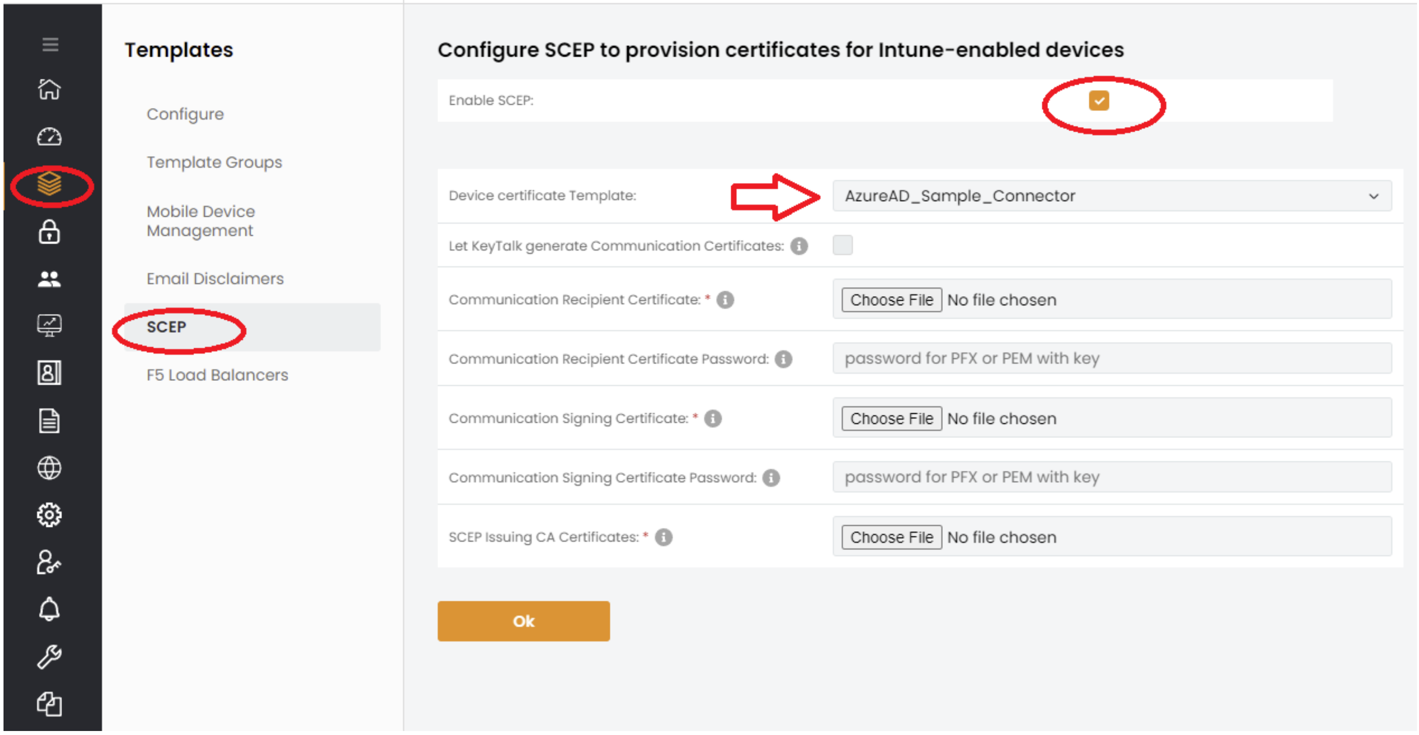
SCEP for Win10

+ Select a certificate profile

Maintenance

Please ensure to keep the RA certificates on the KeyTalk CKSM side, up-to-date should your CA-source ever change its issuing CA.

Manual mode:

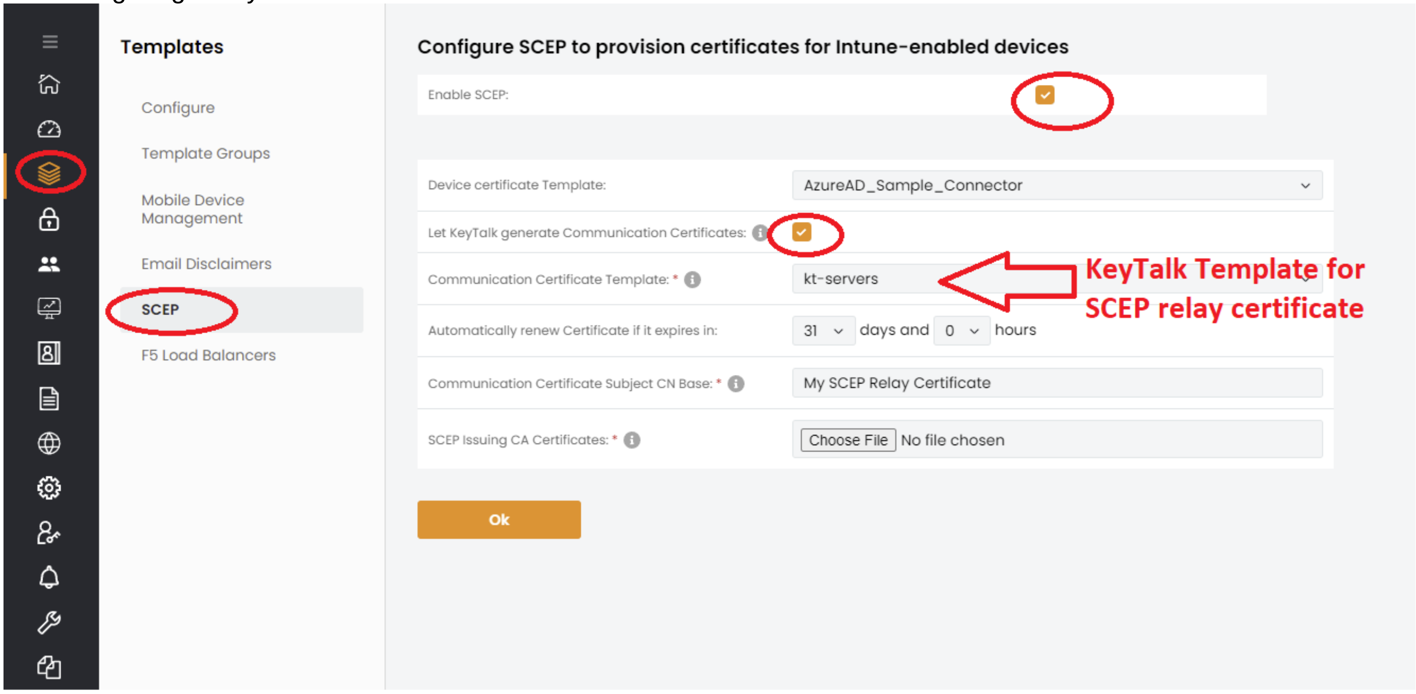


Configure SCEP to provision certificates for Intune-enabled devices

Enable SCEP:	☑
Device certificate Template:	AzureAD_Sample_Connector
Let KeyTalk generate Communication Certificates:	□
Communication Recipient Certificate: *	Choose File No file chosen
Communication Recipient Certificate Password:	password for PFX or PEM with key
Communication Signing Certificate: *	Choose File No file chosen
Communication Signing Certificate Password:	password for PFX or PEM with key
SCEP Issuing CA Certificates: *	Choose File No file chosen

Ok

The maintenance can be automated by assigning a TEMPLATE to issue the RA certificate for SCEP purposes under and configuring it KeyTalk CKMS side:



Configure SCEP to provision certificates for Intune-enabled devices

Enable SCEP: ☑

Device certificate Template: AzureAD_Sample_Connector

Let KeyTalk generate Communication Certificates: ☑

Communication Certificate Template: * kt-servers

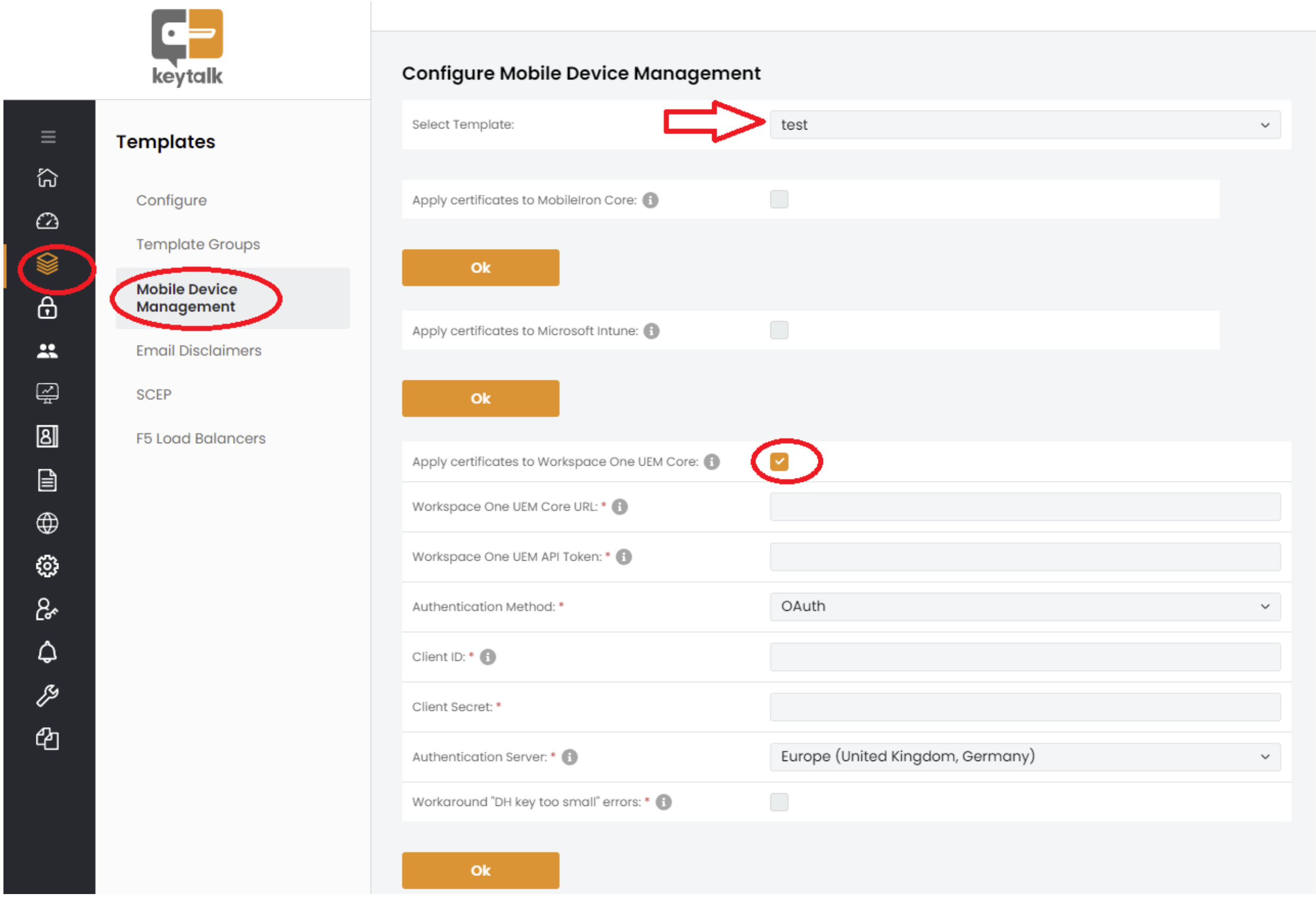
Automatically renew Certificate if it expires in: 31 days and 0 hours

Communication Certificate Subject CN Base: * My SCEP Relay Certificate

SCEP Issuing CA Certificates: * Choose File No file chosen

22.4.3 VMware Workspace One UEM / AirWatch

In order to have KeyTalk CKMS push S/MIME certificates to VMware Workspace One UEM, simply configure a KeyTalk TEMPLATE to send its issued certificates and keys to it.



Configure Mobile Device Management

Select Template:	test
Apply certificates to MobileIron Core:	☐

Apply certificates to Microsoft Intune:	☐

Apply certificates to Workspace One UEM Core:	☑
Workspace One UEM Core URL: *	
Workspace One UEM API Token: *	
Authentication Method: *	OAuth
Client ID: *	
Client Secret: *	
Authentication Server: *	Europe (United Kingdom, Germany)
Workaround "DH key too small" errors: *	☐

Should OAuth as the principal authentication method not be used, then change the authentication to certificate based.

In order to push certificates to VMware Workspace One UEM, the used username, either used to authenticate against KeyTalk CKMS or as imported into KeyTalk CKMS, must match with the usernames used in the MDM.

22.5 Manual certificate (and key) enrolment

Depending on the connected CA-source in a configured KeyTalk certificate template TEMPLATE, generated CSRs are sent automatically to the CA-source, or will await manual processing against an (airgapped) CA.

22.5.1 Process flow and logic

A certificate template TEMPLATE in KeyTalk defines the applied CA-source(s), and depending on the selected CA-source its default certificate key length, subject, CA-source, KU, and EKU.

Each certificate template TEMPLATE, is connected to a Registration Authority authentication module.
KeyTalk CKMS will validate and end-point's certificate request based on the certificate template, against the configured Registration Authority authentication module.
When a username or CN of an end-point does NOT exist on the RA authentication module, KeyTalk CKMS will refuse the issuance of the certificate.

When a username or CN of an end-point does exist, and authenticates positively (the auth credentials provided by the end-point, or the TEMPLATE admin account as configured on the RA auth module for manual processing), the KeyTalk CKMS will request the RA auth module for optional other values, such as default overwrite values for CN, subject, SAN, KU and EKU.
So should a certificate require a value in the CN of the certificate that differs from the used username, and default certificate template details, this unique information MUST come from the configured RA authentication module. When an AD is connected as RA, these values will come from the configured certificate field mapping against relevant AD attributes.

When the RA authentication module is configured to be the KeyTalk CKMS internal DB, all these unique values in a certificate for an end-point, will thus come from the KeyTalk internalDB, and be applied to the generated CSR.

The CSR is either sent automatically to the configured CA-source, or can be manually created and downloaded for further processing.

When the sending of the CSR to the CA-source is automated, the returned certificate is stored with its private key (when applicable) in the KeyTalk CKMS for further manual or automated distribution.
When no CA-source was configured, the KeyTalk CKMS will await the upload of the certificate as DER, PEM, CRT, CER or PB7.

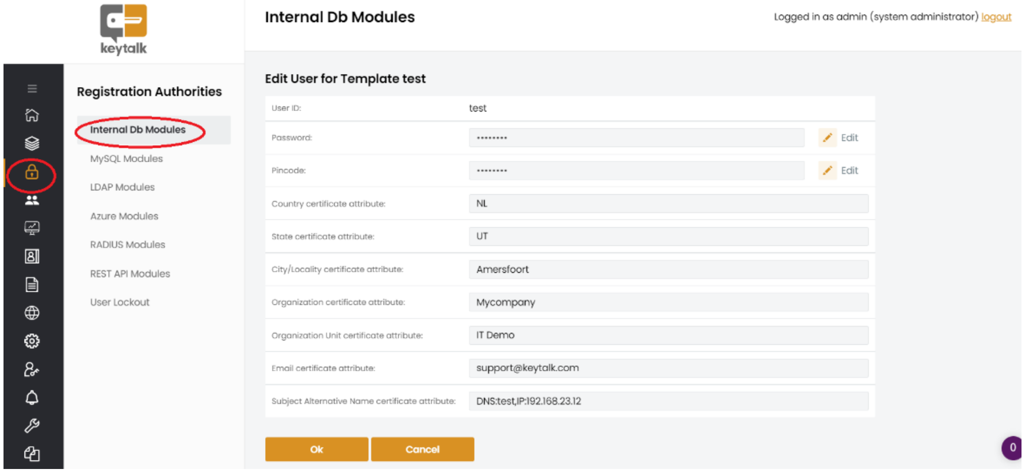
Once the certificate is in the KeyTalk CKMS, KeyTalk CKMS and/or the KeyTalk client on the actual end-point will read the expiry date and revocation information, and use it to guard the status of the certificate for auto-renewal or manual renewal with a set of automated warnings to the in KeyTalk CKMS registered certificate owner and/or the certificate template TEMPLATE managers.

22.5.2 KeyTalk CKMS InternalDB RA auth module and custom certificate meta data

All remote RA authentication modules and how unique end-point certificate meta data can be fetched is described in chapter 13.

Using the KeyTalk CKMS InternalDB as an RA, allows for full manual control within just the KeyTalk CKMS.

When an account with an authorized role related to the used certificate TEMPLATE manually creates a new end-point, this person can also define the unique contents such as CN and SAN.



Internal Db Modules

Edit User for Template test

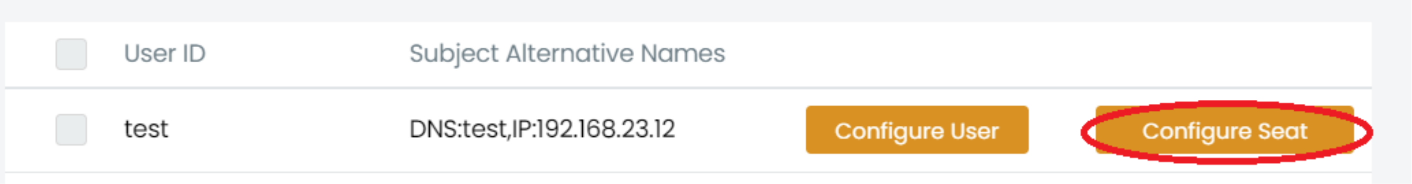
User ID:	test
Password:	••••••••
Pincode:	••••••••
Country certificate attribute:	NL
State certificate attribute:	UT
City/Locality certificate attribute:	Amersfoort
Organization certificate attribute:	Mycompany
Organization Unit certificate attribute:	IT Demo
Email certificate attribute:	support@keytalk.com
Subject Alternative Name certificate attribute:	DNS:test,IP:192.168.23.12

When no custom CN is defined, KeyTalk will use the User ID as the CN.

When no password is defined, KeyTalk CKMS will add a random password for security purposes.

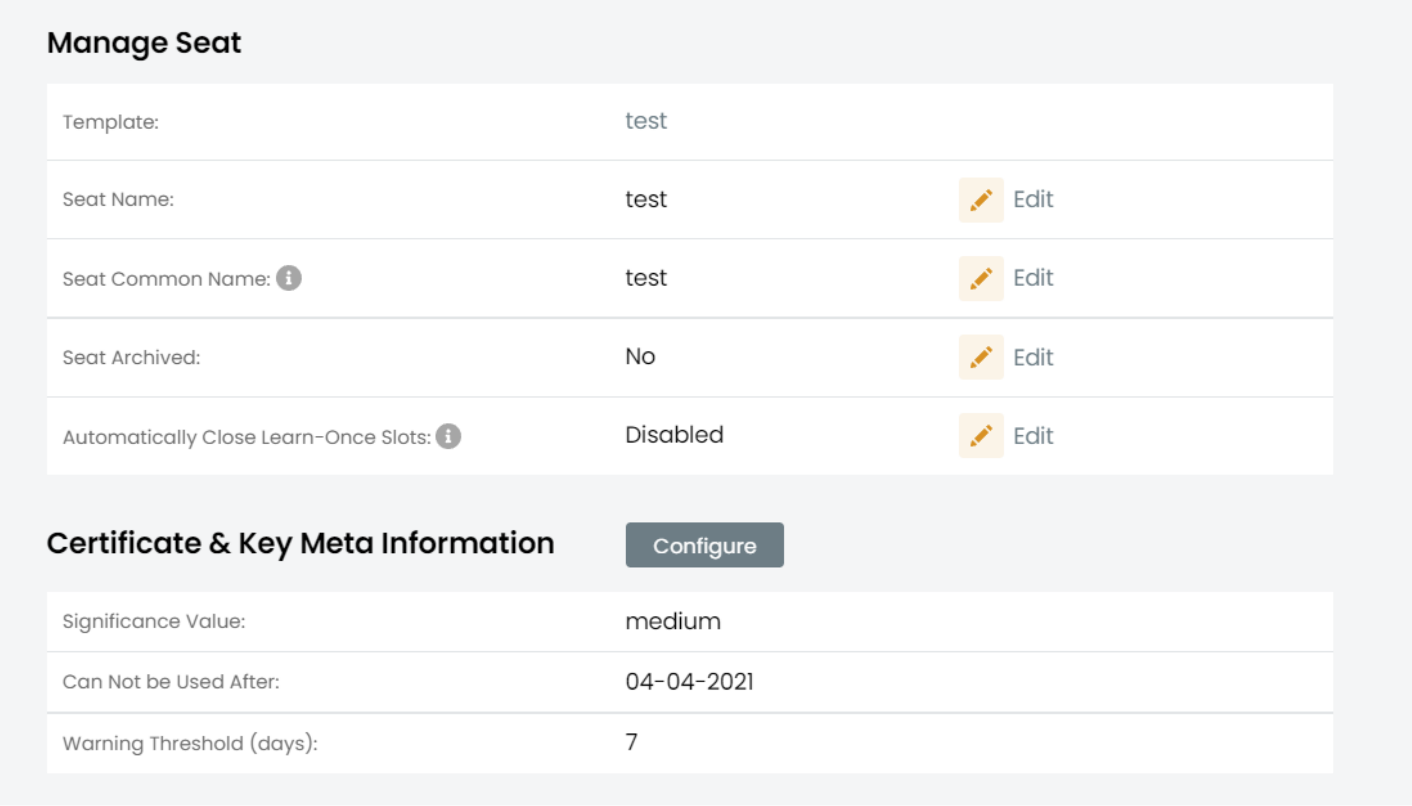
22.5.3 Enrolled devices and users

Once defined in the InternalDB, the end-point seat is generated as an entity in SEATS under its relevant certificate TEMPLATE. From the InternalDB the authorized person can directly click to the enrolled entity for further processing when manual processing is configured.



□	User ID	Subject Alternative Names
□	test	DNS:test,IP:192.168.23.12

Selecting the Actions button allows for accessing the enrolled certificate, meta-data, target devices the certificate has been enrolled to, and more:



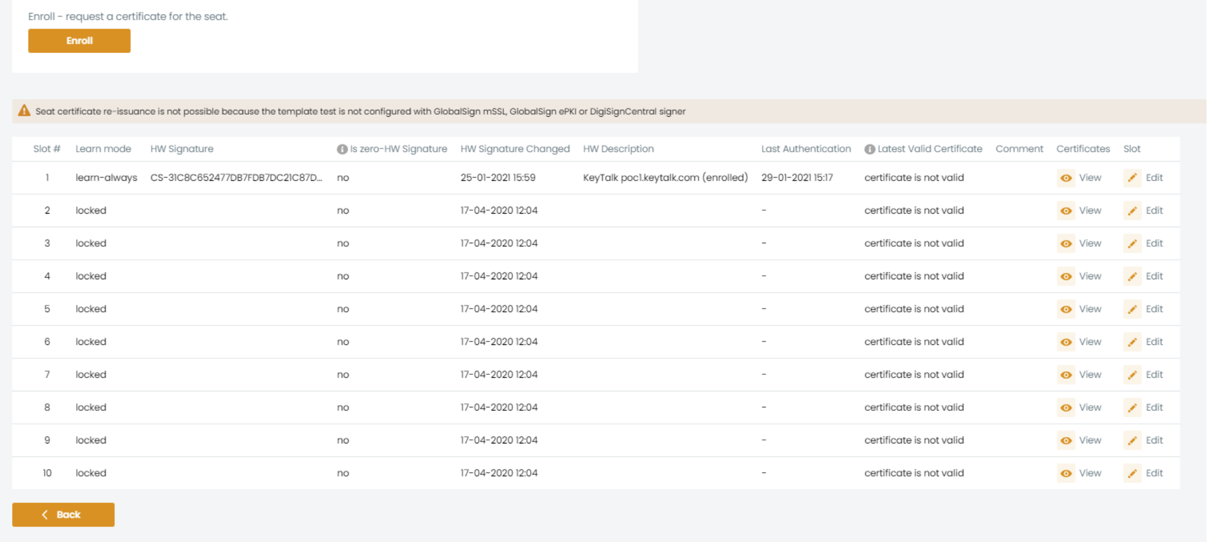
Manage Seat

Template:	test
Seat Name:	test
Seat Common Name:	test
Seat Archived:	No
Automatically Close Learn-Once Slots:	Disabled

Certificate & Key Meta Information

Significance Value:	medium
Can Not be Used After:	04-04-2021
Warning Threshold (days):	7

The upper half of the enrolled device identity and user end-point screen, allows for the direct manipulation of the meta data, and if the logged in authorized person does not have permissions to download a private key, they can request their manager permission to download the private key using a 4-eye principle:



Enroll - request a certificate for the seat.

⚠ Seat certificate re-issuance is not possible because the template test is not configured with GlobalSign mSSL, GlobalSign ePKI or DigiSignCentral signer

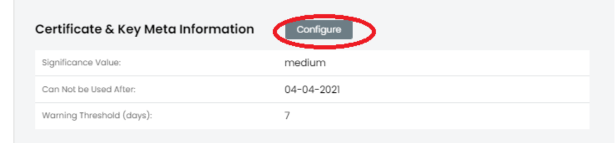
Slot #	Learn mode	HW Signature	Is zero-HW Signature	HW Signature Changed	HW Description	Last Authentication	Latest Valid Certificate	Comment	Certificates	Slot
1	learn-always	CS-31C8C652477DB7FDB7DC21C87D...	no	25-01-2021 15:59	KeyTalk poc1.keytalk.com (enrolled)	29-01-2021 15:17	certificate is not valid		View	Edit
2	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
3	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
4	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
5	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
6	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
7	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
8	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
9	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit
10	locked		no	17-04-2020 12:04		-	certificate is not valid		View	Edit

The lower half of the enrolled device identity and user end-point screen, allows for managing which end-point devices a certificate (and key) have been enrolled to. Each slot numbered 1-10 represent a device with its own name and hardware/software derived SHA2 pre-shared secret, as calculated by the KeyTalk (client) software over the in the TEMPLATE defined characteristics.

ENROLLing a certificate, is only possible when a CA-source has been configured, and when reuse issued certificate and key has been selected in the certificate template TEMPLATE.

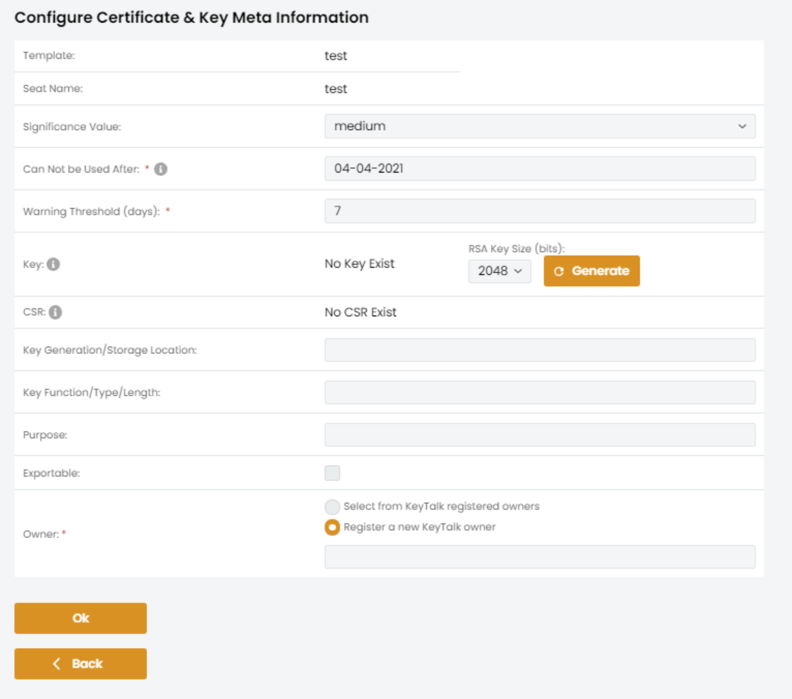
22.5.4 Managing certificate reporting meta data, key and CSR, email/SMS warnings

Where the CN, SAN and subject meta-data can be uniquely configured in the RA auth module, certificate and key related management and reporting meta data can be manually requested/updated/changed by selecting the meta data change button as described in 21.5.3



Certificate & Key Meta Information

Significance Value:	medium
Can Not be Used After:	04-04-2021
Warning Threshold (days):	7



Configure Certificate & Key Meta Information

Template:	test
Seat Name:	test
Significance Value:	medium
Can Not be Used After: *	04-04-2021
Warning Threshold (days): *	7
Key:	No Key Exist RSA Key Size (bits): 2048 Generate
CSR:	No CSR Exist
Key Generation/Storage Location:	
Key Function/Type/Length:	
Purpose:	
Exportable:	□
Owner: *	○ Select from KeyTalk registered owners ● Register a new KeyTalk owner

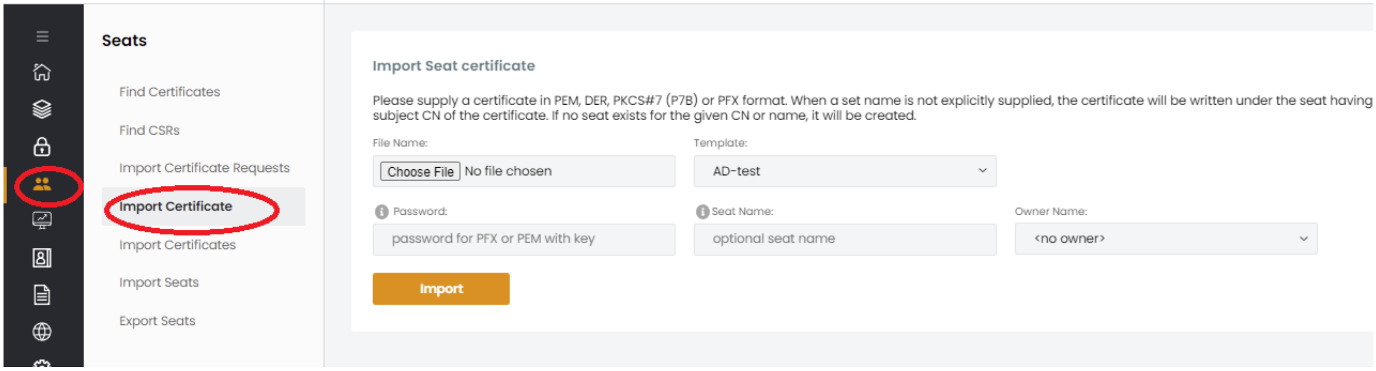
This meta data can be used for certificate & key reporting and tracking even when no actual certificate exists. In this case the KeyTalk CKMS will treat the entry as a label only.

Only when a key was generated, will it be possible to generate the CSR for manual download and manual signing processing.

KeyTalk CKMS will send an email to the assigned certificate and key Owner, when the custom Warning Threshold is reached. Optionally (certificate template configuration) the certificate template TEMPLATE administrator is warning in parallel with the assigned Owner.
Additional warnings are sent at 30, 7, 6, 5, 4, 3, 2, 1, 0 days before expiry.

22.5.5 CSR manual processing, and signed certificate manual upload

After having manually generated a CSR, the CSR can be downloaded from the same screen as it was generated.
Once the CSR has been processed, the certificate only needs to be uploaded as a DER, PEM, PB7, CRT, CER file.
KeyTalk CKMS will match the found cert CN to the record that is pending an upload based on a generated CSR:



Import Seat certificate

Please supply a certificate in PEM, DER, PKCS#7 (P7B) or PFX format. When a set name is not explicitly supplied, the certificate will be written under the seat having subject CN of the certificate. If no seat exists for the given CN or name, it will be created.

File Name: Choose File No file chosen

Template: AD-test

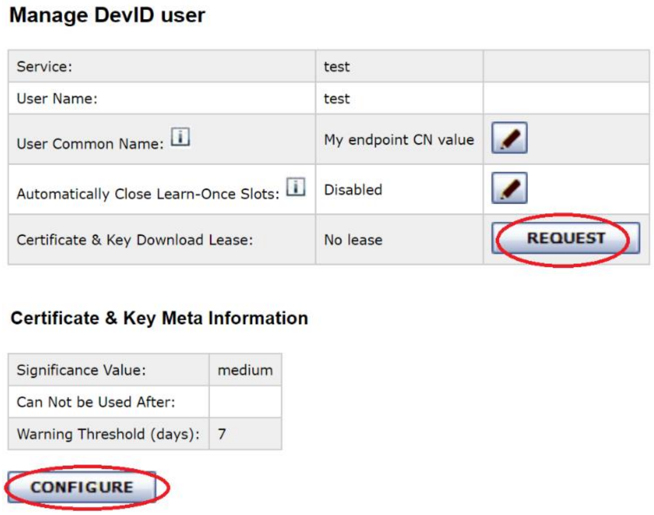
Password: password for PFX or PEM with key

Seat Name: optional seat name

Owner Name: <no owner>

22.5.6 Downloading current and historic/archived certificates and keys

An enrolled certificate and private key, can only be downloaded when proper authorization exists.
When no authorization exist, the person who wants to download the certificate and private key need to request a download approval lease.



Manage DevID user

Service:	test	
User Name:	test	
User Common Name:	My endpoint CN value	
Automatically Close Learn-Once Slots:	Disabled	
Certificate & Key Download Lease:	No lease	REQUEST

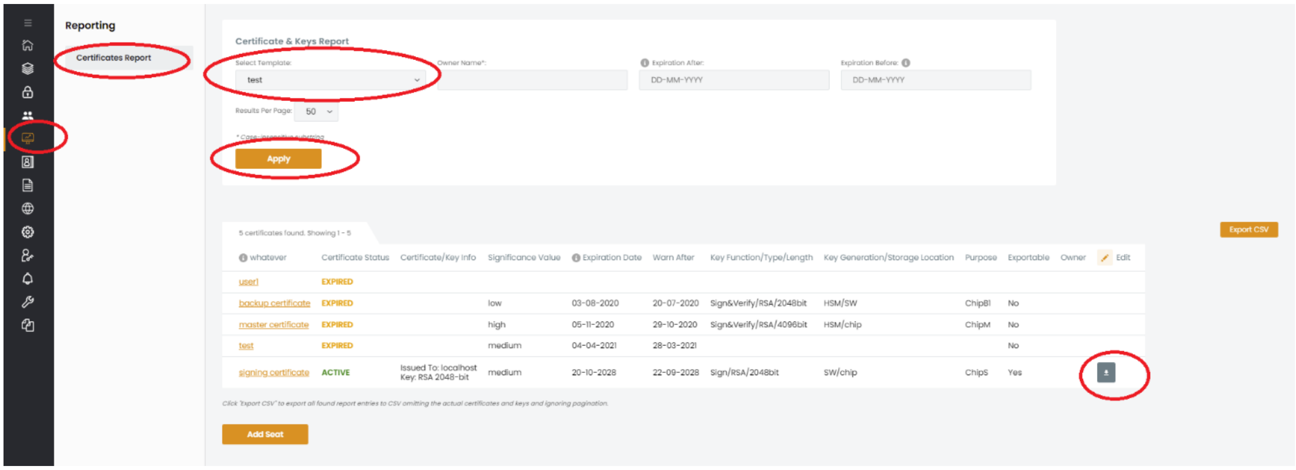
Certificate & Key Meta Information

Significance Value:	medium
Can Not be Used After:	
Warning Threshold (days):	7

After the certificate TEMPLATE administrator has approved the download lease can the certificate and key be downloaded during a given window of opportunity.

Certificate TEMPLATE administrators and System Admins can currently always download certificates and private keys, though this will likely change in the near future.

Select the target device slot that a certificate was enrolled to, and press Actions to download the current certificate and private key, or download all historic certificates and keys.



Certificate & Keys Report

Select Template: test

Owner Name*:

Expiration After: DD-MM-YYYY

Expiration Before: DD-MM-YYYY

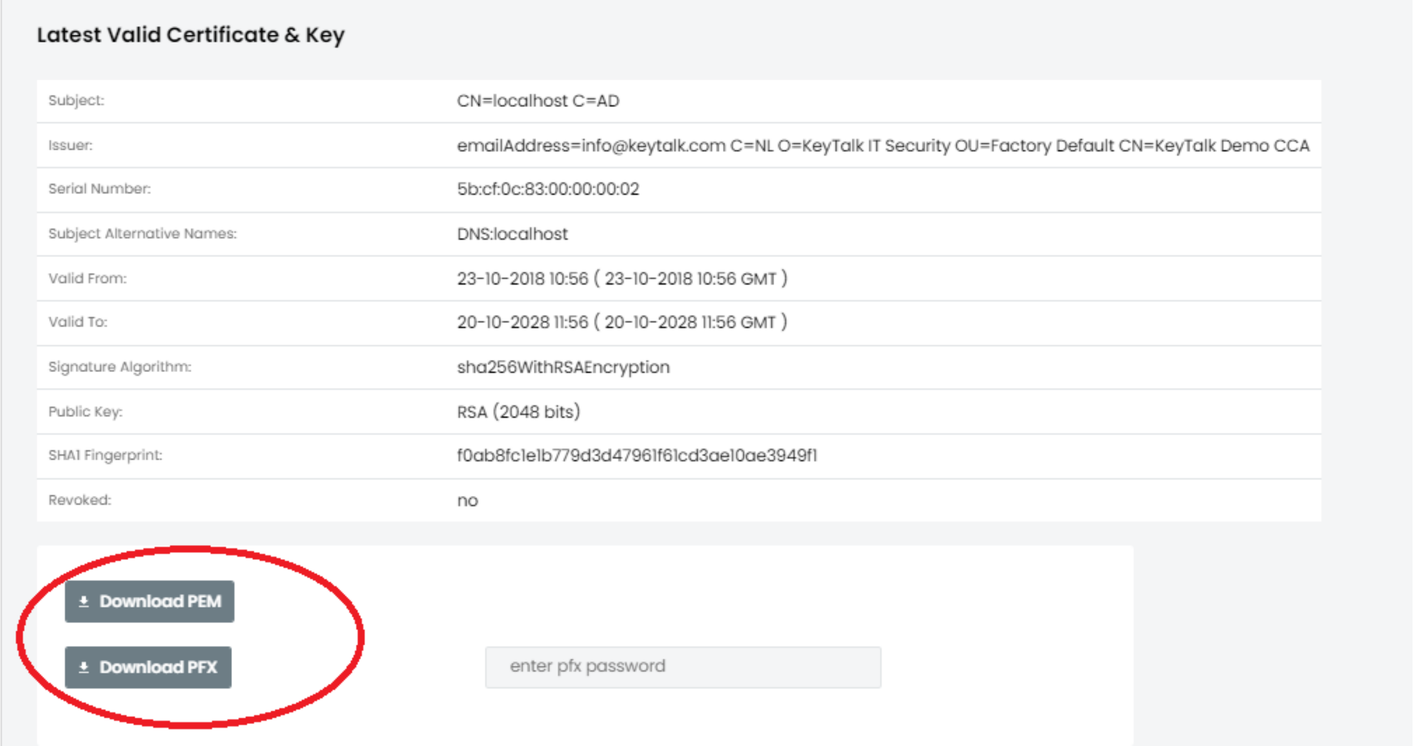
Results Per Page: 50

* Case-insensitive substring

5 certificates found. Showing 1 - 5

whatever	Certificate Status	Certificate/Key Info	Significance Value	Expiration Date	Warn After	Key Function/Type/Length	Key Generation/Storage Location	Purpose	Exportable	Owner	Edit
user1	EXPIRED										
backup certificate	EXPIRED		low	03-08-2020	20-07-2020	Sign&Verify/RSA/2048bit	HSM/SW	ChipB1	No		
master certificate	EXPIRED		high	05-11-2020	29-10-2020	Sign&Verify/RSA/4096bit	HSM/chip	ChipM	No		
test	EXPIRED		medium	04-04-2021	28-03-2021				No		
signing certificate	ACTIVE	Issued To: localhost Key: RSA 2048-bit	medium	20-10-2028	22-09-2028	Sign/RSA/2048bit	SW/chip	ChipS	Yes		

Click 'Export CSV' to export all found report entries to CSV omitting the actual certificates and keys and ignoring pagination.



Latest Valid Certificate & Key

Subject:	CN=localhost C=AD
Issuer:	emailAddress=info@keytalk.com C=NL O=KeyTalk IT Security OU=Factory Default CN=KeyTalk Demo CCA
Serial Number:	5b:cf:0c:83:00:00:00:02
Subject Alternative Names:	DNS:localhost
Valid From:	23-10-2018 10:56 (23-10-2018 10:56 GMT)
Valid To:	20-10-2028 11:56 (20-10-2028 11:56 GMT)
Signature Algorithm:	sha256WithRSAEncryption
Public Key:	RSA (2048 bits)
SHA1 Fingerprint:	f0ab8fc1e1b779d3d47961f61cd3ae10ae3949f1
Revoked:	no

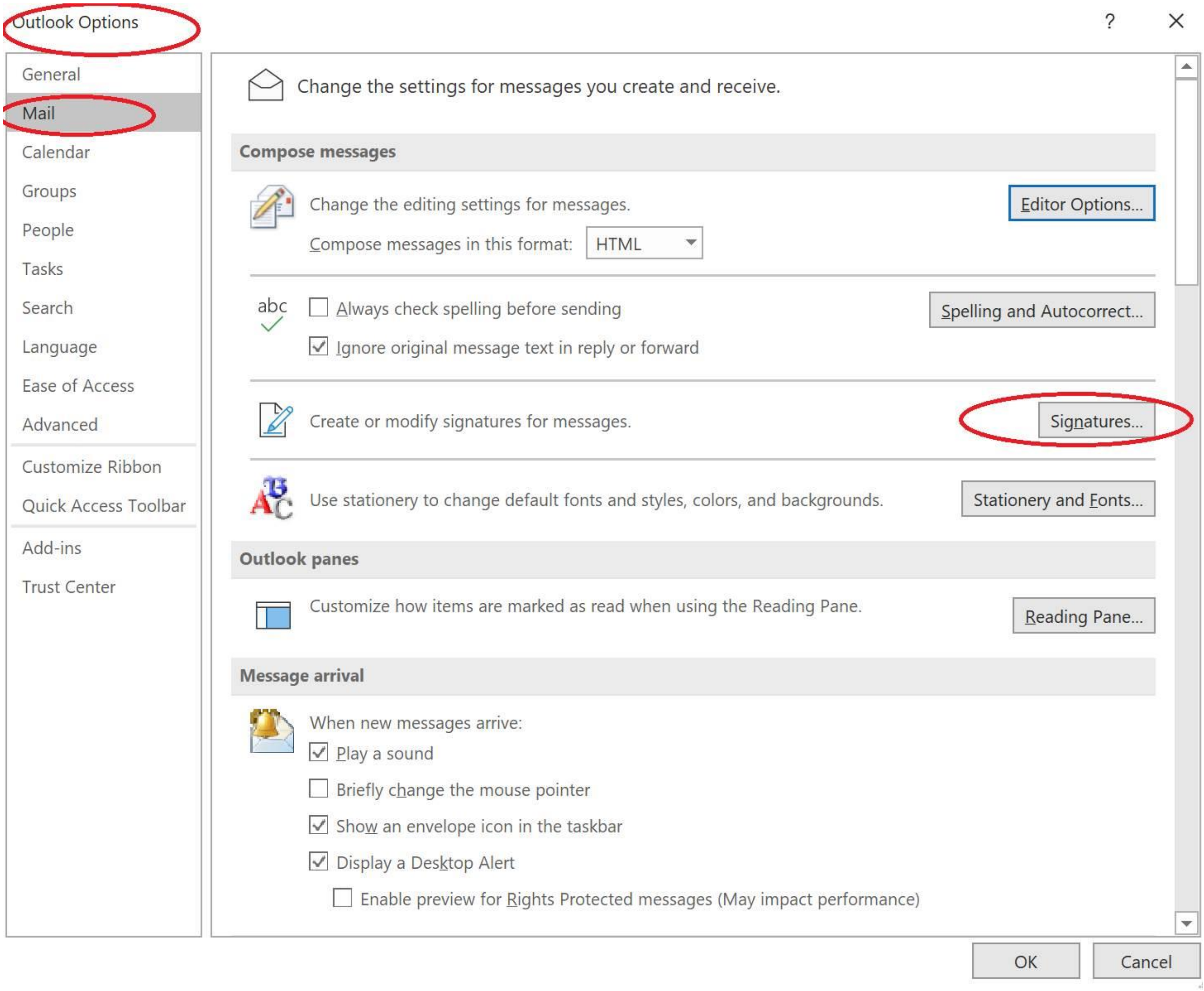
23 (Windows) Outlook automated email disclaimer configuration

When issuing S/MIME certificates through KeyTalk CKMS, for email digital signing and/or encryption purposes, the KeyTalk CKMS can automatically configure Outlook (for Windows) to apply the S/MIME certificate.

KeyTalk CKMS can also auto configure any applicable textual email signature/disclaimer after an S/MIME certificate is successfully obtained.

Follow these steps to configure KeyTalk CKMS to apply any particular email textual signature/disclaimer.

Step 1: Create the default email disclaimer in Outlook



Outlook Options

Change the settings for messages you create and receive.

Compose messages

Change the editing settings for messages. [Editor Options...]

Compose messages in this format: HTML

abc □ Always check spelling before sending [Spelling and Autocorrect...]

☑ Ignore original message text in reply or forward

Create or modify signatures for messages. [Signatures...]

Use stationery to change default fonts and styles, colors, and backgrounds. [Stationery and Fonts...]

Outlook panes

Customize how items are marked as read when using the Reading Pane. [Reading Pane...]

Message arrival

When new messages arrive:

☑ Play a sound

□ Briefly change the mouse pointer

☑ Show an envelope icon in the taskbar

☑ Display a Desktop Alert

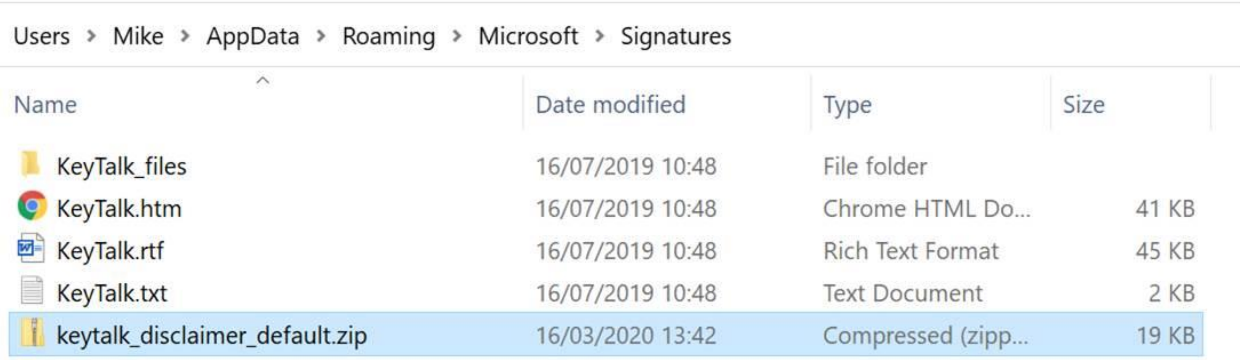
□ Enable preview for Rights Protected messages (May impact performance)

NOTE: Ensure the disclaimer text starts with the word: DISCLAIMER

The word disclaimer is used as an anchorpoint to determine where the Disclaimer text starts

Step 2: Add the default signature of Outlook into a zip file.

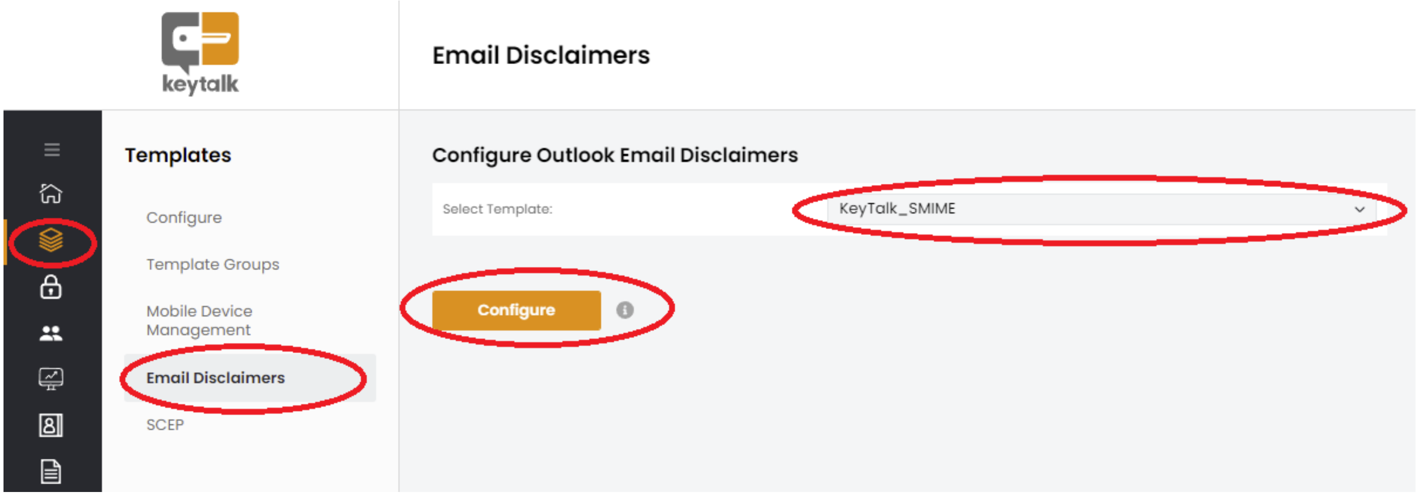
Include the subdirectory , the htm rtf and txt file



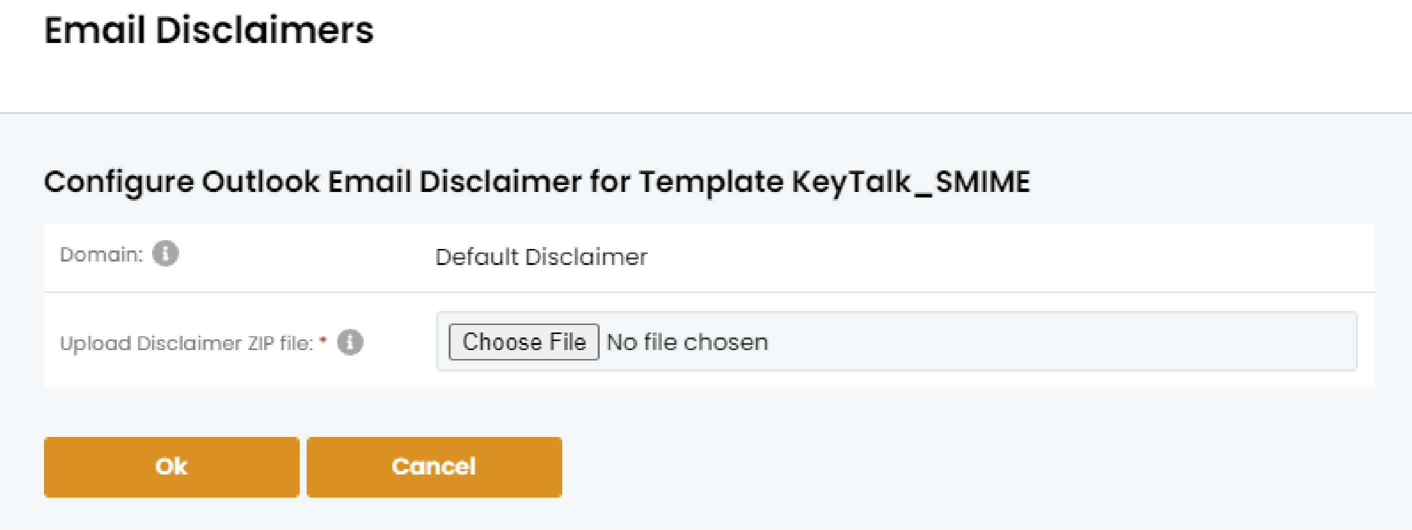
Users › Mike › AppData › Roaming › Microsoft › Signatures

Name	Date modified	Type	Size
KeyTalk_files	16/07/2019 10:48	File folder	
KeyTalk.htm	16/07/2019 10:48	Chrome HTML Do...	41 KB
KeyTalk.rtf	16/07/2019 10:48	Rich Text Format	45 KB
KeyTalk.txt	16/07/2019 10:48	Text Document	2 KB
keytalk_disclaimer_default.zip	16/03/2020 13:42	Compressed (zipp...	19 KB

Step 3: Upload the default signature zip file into KeyTalk



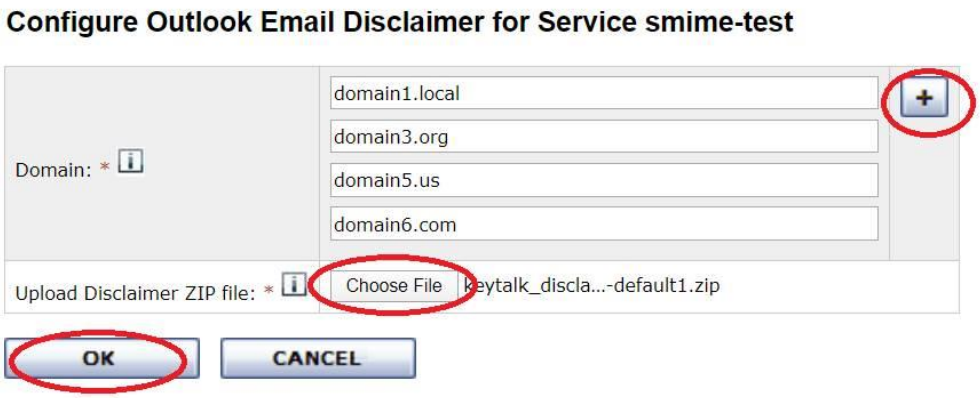
Select Template: KeyTalk_SMIME


Email Disclaimers
Configure Outlook Email Disclaimer for Template KeyTalk_SMIME
Choose File
Ok
Cancel

Domain: Default Disclaimer

Upload Disclaimer ZIP file: * No file chosen

Step 4: Optionally add non-default disclaimers, as applied to other defined domains



Configure Outlook Email Disclaimer for Service smime-test

Domain: * domain1.local
domain3.org
domain5.us
domain6.com

Upload Disclaimer ZIP file: * keytalk_discla...-default1.zip

24 Configuring OWA/EO for S/MIME

Once S/MIME certificates have been deployed, the Exchange or Office 365 webmail environment and other devices often need to be configured for use with S/MIME.

KeyTalk has summarized most commonly needed configurations here:
https://downloads.keytalk.com/downloads/documents/KeyTalk_Anything_You_Ever_Wanted_To_Know_About_SMIME_Email_Encryption_DigitalSigning_Configurations._But_Were_Afraid_To_Ask.pdf

25 Changing docker IP

KeyTalk runs a docker image with its own internal IP, which could cause a conflict in your network.

To change it, login using SSH or CLI:

```
$ sudo vim /lib/systemd/system/docker.service
```

Append `--bip "IP/netmask"` at the end of "`ExecStart`" command

for example,

```
ExecStart=/usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --bip
"172.18.0.1/16"
```

```
$ sudo systemctl daemon-reload
```

```
$ sudo systemctl restart docker
```

26 KeyTalk contact details and 3rd line support

If in need of support due to something not working right, kindly always include a Problem Report.
This PR can be generated on both clients and the KeyTalk virtual appliance.
On the virtual appliance: SYSTEM -> Report Problem -> Generate

KeyTalk IT Security is registered with the Dutch chamber of commerce under: 59072555
with registered VAT number: NL853305766B01

Visit us at:
Maanlander 47
3824 MN Amersfoort
The Netherlands

Our invoice address:
Keulsekade 218
3534AC Utrecht
The Netherlands

Phone: +31 88 KEYTALK or +31 88 5398255
Email: sales@keytalk.com
Opening hours: Mo-Fr 08:00 – 18:00 (10/5)

Customer and partner technical 3rd line support
Phone: +31 88 KEYTALK or +31 88 5398255
Email: support@keytalk.com
Opening hours: Mo-Su 00:00 – 24:00 (24/7)

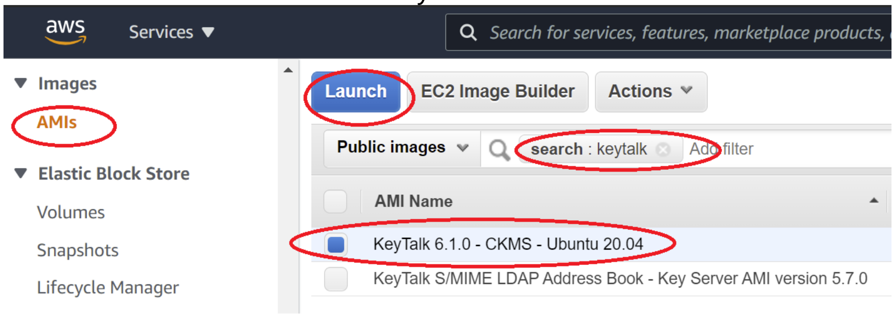
Website:	https://www.keytalk.com
Firmware/software:	https://www.keytalk.com/download

ANNEX A: Deploying KeyTalk virtual appliance in AWS

This guide assumes that you have already created an account for AWS and configured payment for it. These steps are meant for a quick setup, thus skips some advanced configuration options.

Step 1: Login to AWS https://aws.amazon.com/console/

Step 2: Go to EC2 -> AMI -> Search for KeyTalk

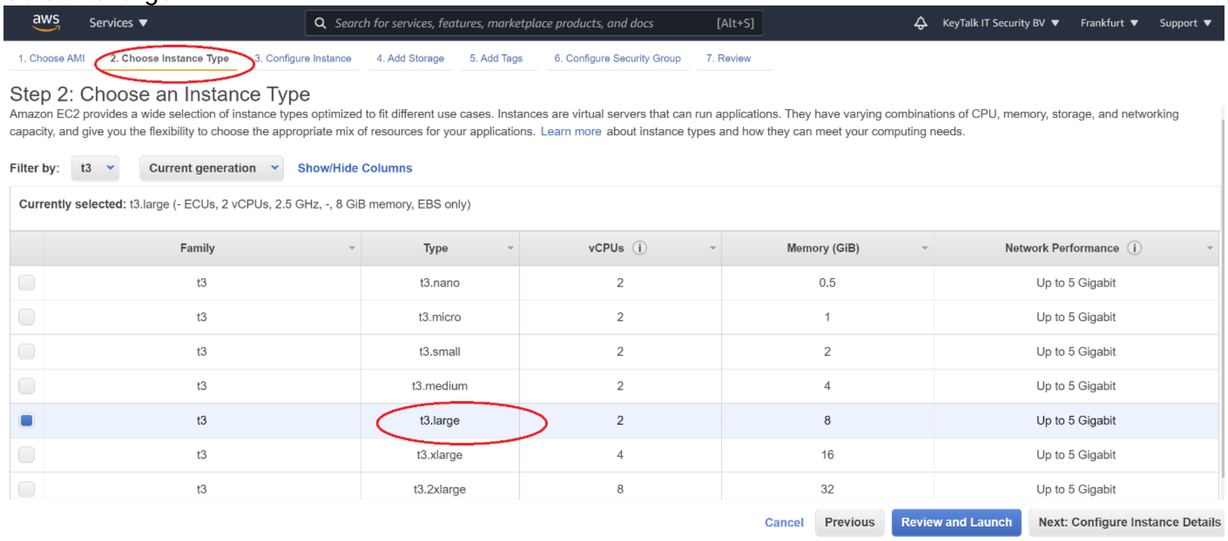


	AMI Name
☑	KeyTalk 6.1.0 - CKMS - Ubuntu 20.04
□	KeyTalk S/MIME LDAP Address Book - Key Server AMI version 5.7.0

***Note**: depending on various ongoing firmware updates, the KeyTalk version shown might differ*

Step 3: Launch the KeyTalk CKMS AMI

Step 4: Select t3.large



Step 2: Choose an Instance Type

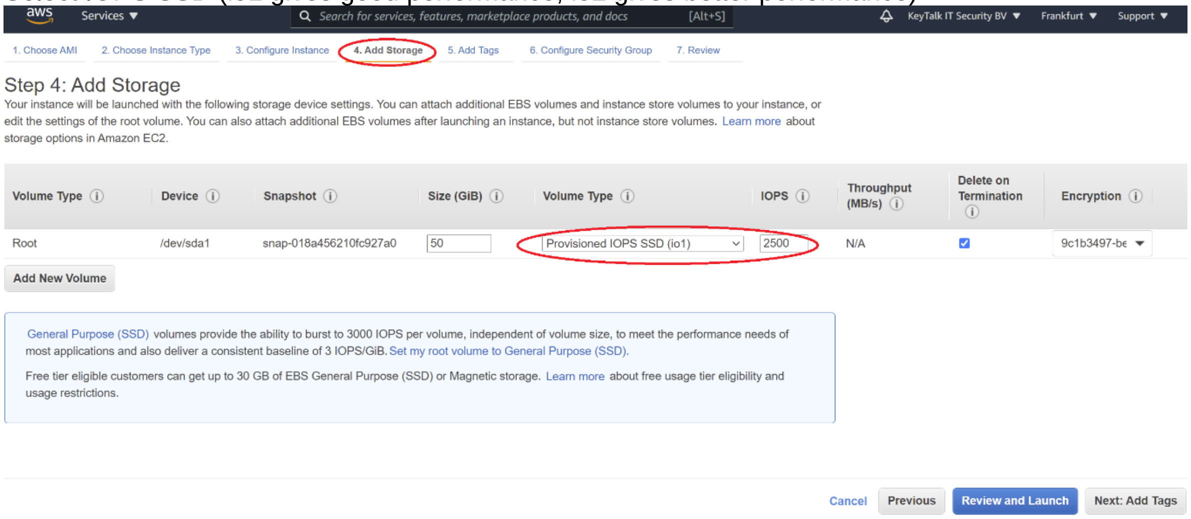
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: t3 | Current generation | Show/Hide Columns

Currently selected: t3.large (- ECUs, 2 vCPUs, 2.5 GHz, -, 8 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Network Performance
□	t3	t3.nano	2	0.5	Up to 5 Gigabit
□	t3	t3.micro	2	1	Up to 5 Gigabit
□	t3	t3.small	2	2	Up to 5 Gigabit
□	t3	t3.medium	2	4	Up to 5 Gigabit
☑	t3	t3.large	2	8	Up to 5 Gigabit
□	t3	t3.xlarge	4	16	Up to 5 Gigabit
□	t3	t3.2xlarge	8	32	Up to 5 Gigabit

Step 5: Select IOPS SSD (io1 gives good performance, io2 gives better performance)



Step 4: Add Storage

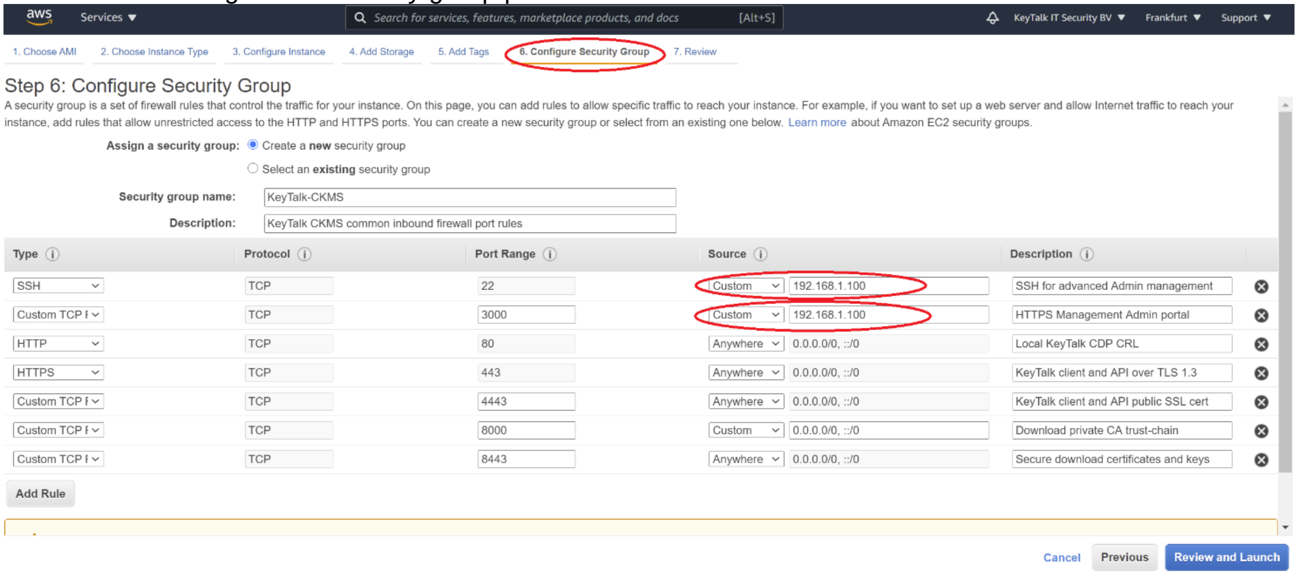
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-018a456210fc927a0	50	Provisioned IOPS SSD (io1)	2500	N/A	☑	9c1b3497-be

General Purpose (SSD) volumes provide the ability to burst to 3000 IOPS per volume, independent of volume size, to meet the performance needs of most applications and also deliver a consistent baseline of 3 IOPS/GiB. Set my root volume to General Purpose (SSD).

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

Step 6: Select and/or configure the security group ports



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

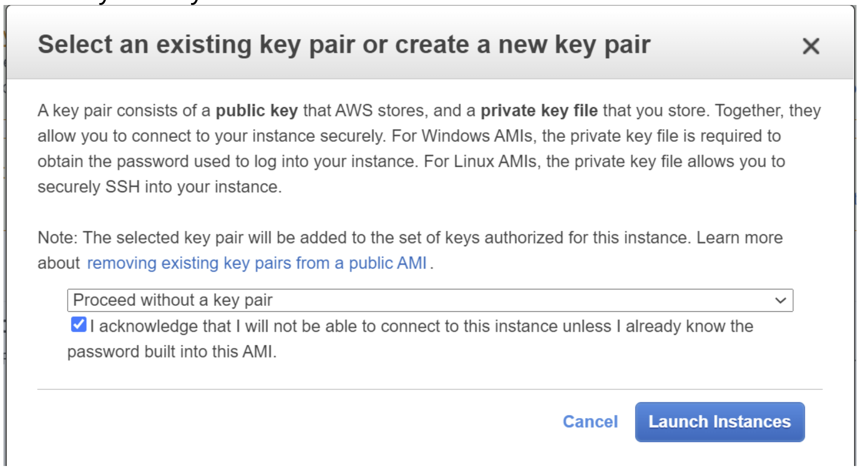
Security group name: KeyTalk-CKMS

Description: KeyTalk CKMS common inbound firewall port rules

Type	Protocol	Port Range	Source		Description
SSH	TCP	22	Custom	192.168.1.100	SSH for advanced Admin management
Custom TCP F	TCP	3000	Custom	192.168.1.100	HTTPS Management Admin portal
HTTP	TCP	80	Anywhere	0.0.0.0/0, ::/0	Local KeyTalk CDP CRL
HTTPS	TCP	443	Anywhere	0.0.0.0/0, ::/0	KeyTalk client and API over TLS 1.3
Custom TCP F	TCP	4443	Anywhere	0.0.0.0/0, ::/0	KeyTalk client and API public SSL cert
Custom TCP F	TCP	8000	Custom	0.0.0.0/0, ::/0	Download private CA trust-chain
Custom TCP F	TCP	8443	Anywhere	0.0.0.0/0, ::/0	Secure download certificates and keys

Note: *Ensure the source IP for ports 22 and 3000 are initially configured for management IPs only.*
KeyTalk self-TEMPLATE portal for end-users also runs on port 3000 but only with 2 way SSL over TLS 1.3 enabled ..

Step 7: Launch your KeyTalk instance



Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

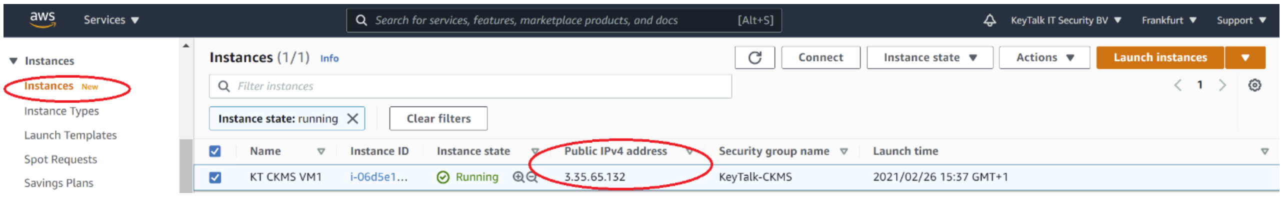
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Proceed without a key pair

☑ I acknowledge that I will not be able to connect to this instance unless I already know the password built into this AMI.

Note: *Possibly the instance will not launch if your account first needs a verification by AWS based on your chosen region. This typically only happens when your account is brand new.*

Step 8: Go to EC2 -> Instances -> select your KeyTalk CKMS instance -> Ensure its running.
Copy the public IP address
Launch a browser which uses your configured port 3000 management IP (see step 6) and visit:
https://<myKeyTalkCKMS_ip>:3000
Login using username: admin password: change!



Instances (1/1) Info

Filter instances

Instance state: running ✕

☑	Name	Instance ID	Instance state	Public IPv4 address	Security group name	Launch time
☑	KT CKMS VM1	i-06d5e1...	Running	3.35.65.132	KeyTalk-CKMS	2021/02/26 15:37 GMT+1

ANNEX B: Importing KeyTalk virtual appliance in Azure

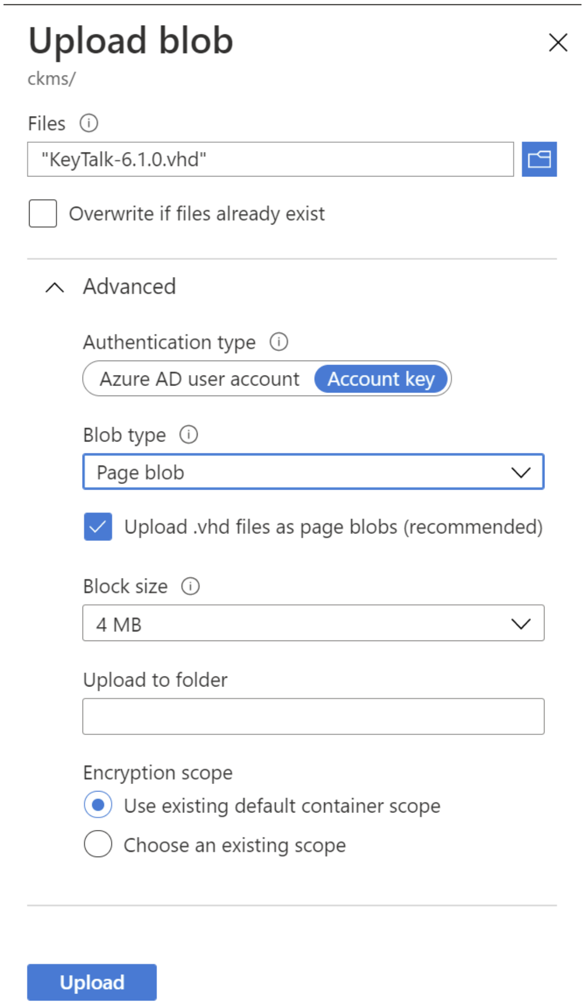
KeyTalk is in the process of being accepted as an Azure Marketplace Partner. While we await our KeyTalk CKMS and LDAP and SSL Scanner to be accepted to the Marketplace, you can use the following steps to deploy your KeyTalk instance to Azure.

These steps assume you are knowledgeable on Azure and already have an Azure account.

Step 1: Download the KeyTalk virtual appliance for Hyper-V / Azure from the KeyTalk website SUPPORT section

Step 2:
Extract the VHD file from the ZIP file, found under: Virtual Hard Disks

Step 3:
In Azure, go to Storage Account -> Resource Groups -> Containers -> Your Blob container
And upload the 50GB VHD file as a Page blob



Upload blob

ckms/

Files: "KeyTalk-6.1.0.vhd"

☐ Overwrite if files already exist

Advanced

Authentication type: Azure AD user account | [✓ Account key]

Blob type: Page blob

☑ Upload .vhd files as page blobs (recommended)

Block size: 4 MB

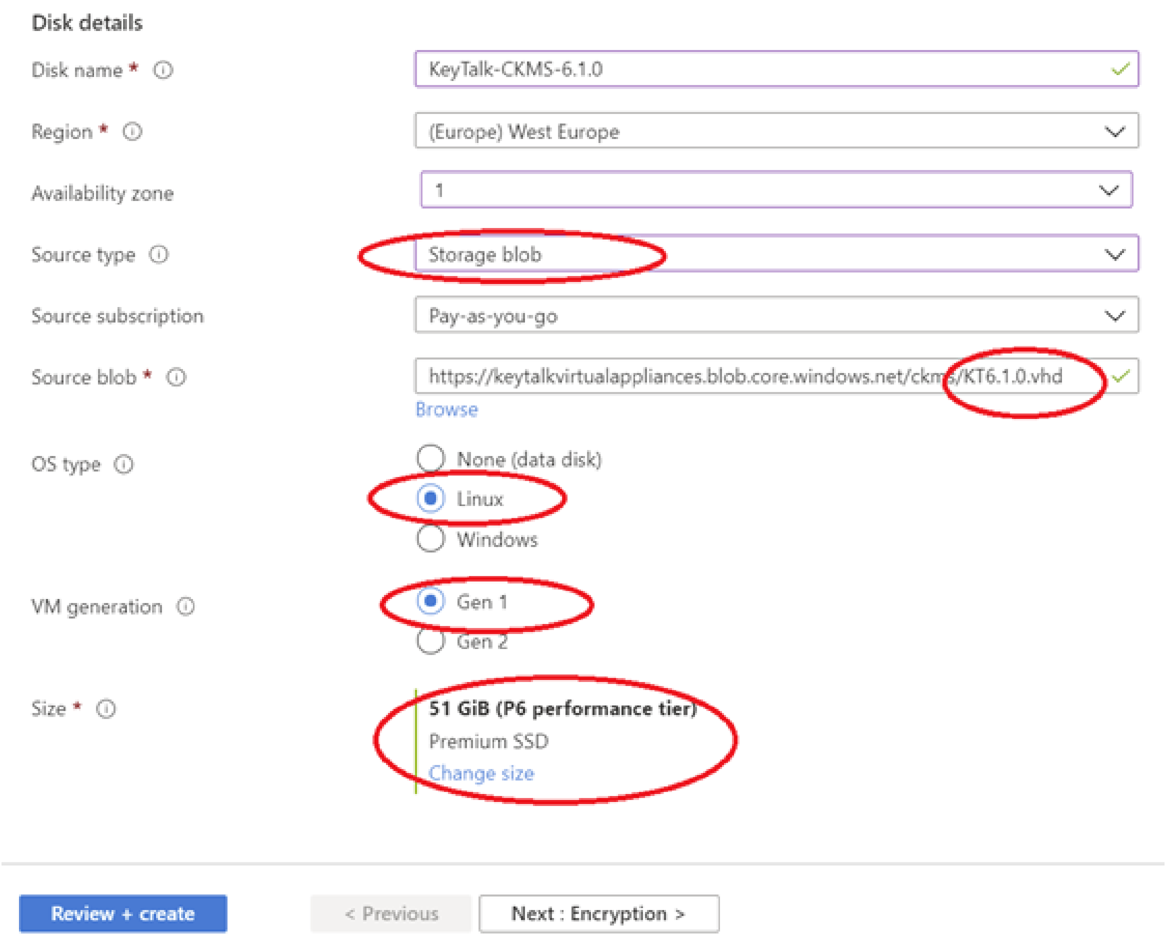
Upload to folder:

Encryption scope

● Use existing default container scope

○ Choose an existing scope

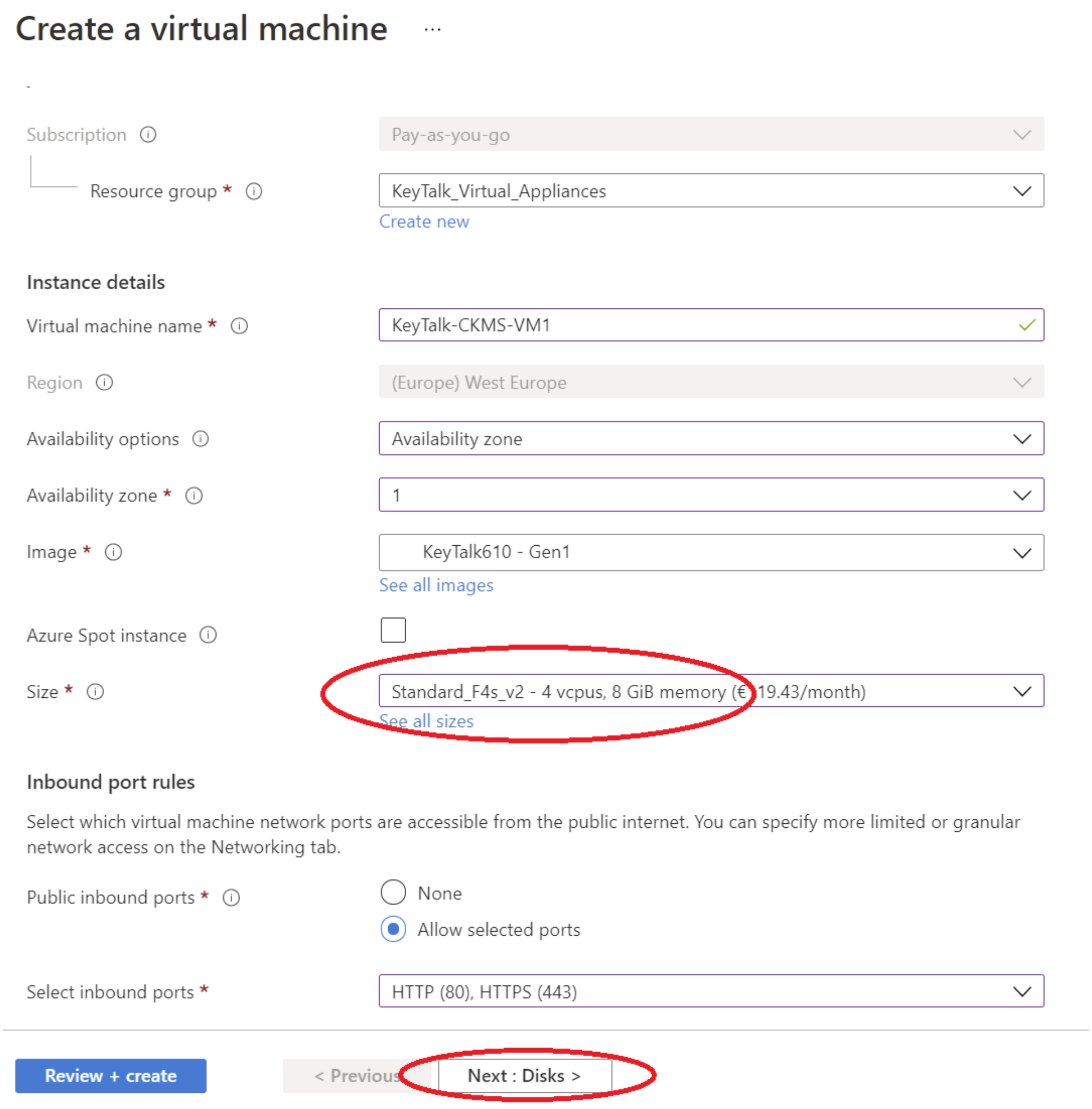
Step 4:
In Azure, go to Disks -> Add/Create a Managed Disk



Disk details

Disk name * ⓘ	KeyTalk-CKMS-6.1.0 ✓
Region * ⓘ	(Europe) West Europe
Availability zone	1
Source type ⓘ	Storage blob
Source subscription	Pay-as-you-go
Source blob * ⓘ	https://keytalkvirtualappliances.blob.core.windows.net/ckms/KT6.1.0.vhd ✓ Browse
OS type ⓘ	○ None (data disk) ● Linux ○ Windows
VM generation ⓘ	● Gen 1 ○ Gen 2
Size * ⓘ	**51 GiB (P6 performance tier)** Premium SSD Change size

Step 5:
In Azure -> Disks -> Select the created disk and select Create VM



Create a virtual machine ···

Subscription ⓘ	Pay-as-you-go
Resource group * ⓘ	KeyTalk_Virtual_Appliances Create new

Instance details

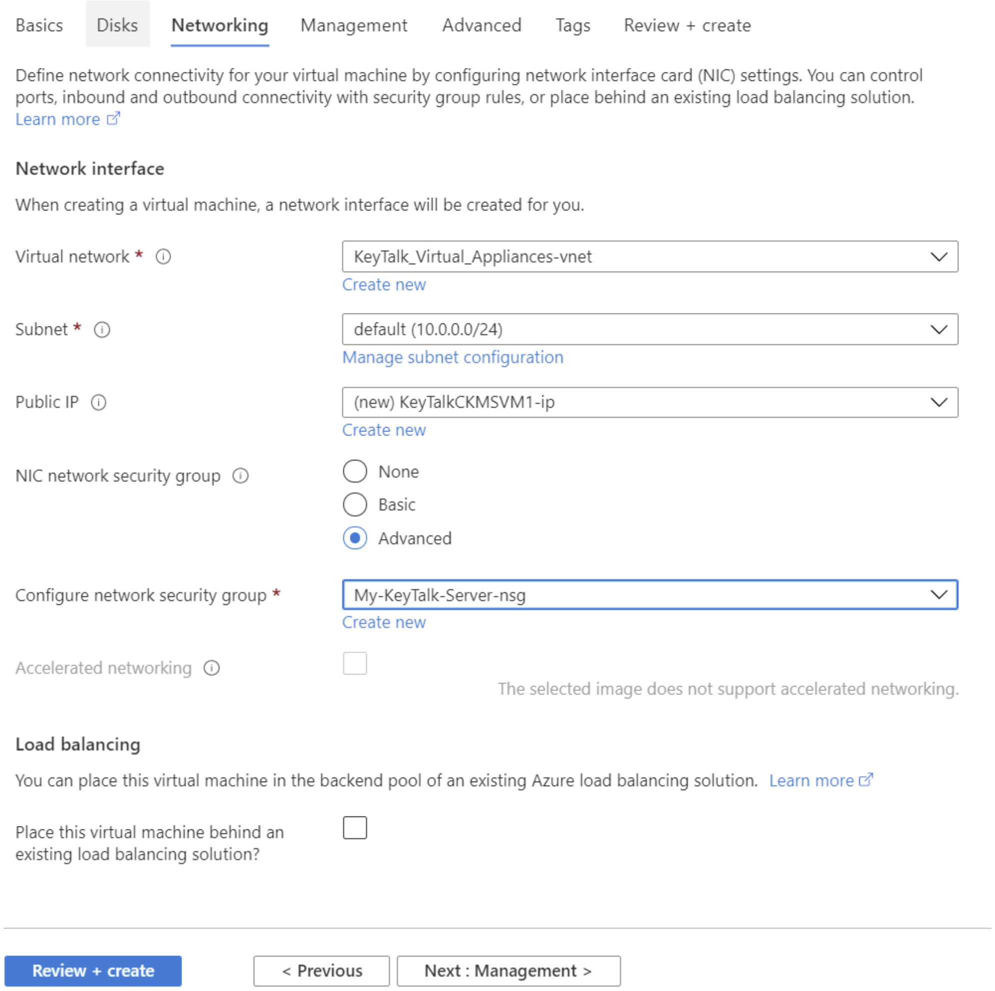
Virtual machine name * ⓘ	KeyTalk-CKMS-VM1 ✓
Region ⓘ	(Europe) West Europe
Availability options ⓘ	Availability zone
Availability zone * ⓘ	1
Image * ⓘ	KeyTalk610 - Gen1 See all images
Azure Spot instance ⓘ	□
Size * ⓘ	Standard_F4s_v2 - 4 vcpus, 8 GiB memory (€ 19.43/month) See all sizes

Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ	○ None ● Allow selected ports
Select inbound ports *	HTTP (80), HTTPS (443)

Step 6:
From the Disks networking tab, configure appropriate settings.
Load Balancing (HTTP passthrough) can be enabled when running KeyTalk CKMS in High Availability mode, also network security groups:



Define network connectivity for your virtual machine by configuring network interface card (NIC) settings. You can control ports, inbound and outbound connectivity with security group rules, or place behind an existing load balancing solution.
Learn more

Network interface

When creating a virtual machine, a network interface will be created for you.

Virtual network * : KeyTalk_Virtual_Appliances-vnet
Create new

Subnet * : default (10.0.0.0/24)
Manage subnet configuration

Public IP : (new) KeyTalkCKMSVM1-ip
Create new

NIC network security group
○ None
○ Basic
● Advanced

Configure network security group * : My-KeyTalk-Server-nsg
Create new

Accelerated networking □
The selected image does not support accelerated networking.

Load balancing

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. Learn more

Place this virtual machine behind an existing load balancing solution? □

For your network security group enable inbound:

Source	Type	Port	Usage
Any or LoadBalancer	TCP	80	Locally hosted CDP CRL
Any or LoadBalancer	TCP	443	KeyTalk client REST-API interface default
Any or LoadBalancer	TCP	4443	KeyTalk client REST-API interface explicit public trust
Any or LoadBalancer	TCP	8000	Download KeyTalk private CA Trust Chain
Management IPs	TCP	22	SSH
Management IPs	TCP	3000	KeyTalk admin management portal webUI

For your network security group enable outbound:

Target	Type	Port	Usage
Any	TCP	80	Ubuntu security updates and other
Any	TCP	443	Various security updates
emea.api.hvca.globalsign.com	TCP	4443	GlobalSign Atlas High Volume CA platform
<configured syslog server>	TCP/UDP	514	Write syslog files

Step 7:
Review and Create your KeyTalk virtual machine

Step 8: OPTIONALLY install the Azure Agent onto the KeyTalk virtual appliance from the CLI/SSH using:
sudo apt-get update && sudo apt-get install walinuxagent

```
sudo systemctl enable walinuxagent
sudo apt-get install python-minimal
```

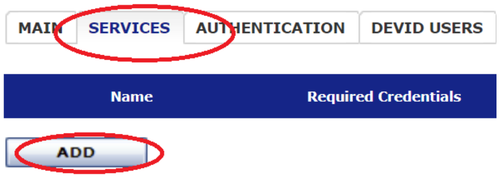
ANNEX C: Sample S/MIME use-case configuration

In this sample configuration, we'll go step by step over enabling the ability to mass deploy S/MIME certificates for your company related email addresses using the KeyTalk client/app.

This use-case sample assumes that:

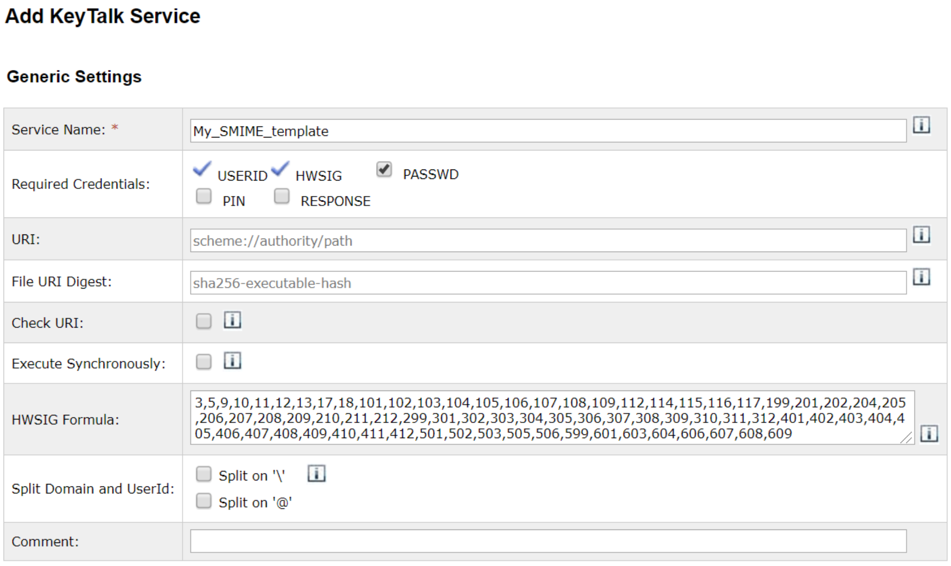
- ✓ an Active Directory will be used, as this is the most common use-case.
- ✓ a KeyTalk server is already generically configured. See chapter 8: Quickguide.
- ✓ An LDAP Address Book is already generically configured (such as the KeyTalk S/MIME LDAP Address Book). Do note that an LDAP Address Book is not required for this use-case, but is often used.

Step 1: Configure a KeyTalk certificate template under TEMPLATES



Name	Required Credentials

Step 2: Configure the TEMPLATE general settings, and ensure you select password. Optionally make appropriate changes to the HWSIG Formula hardware/software characteristics.

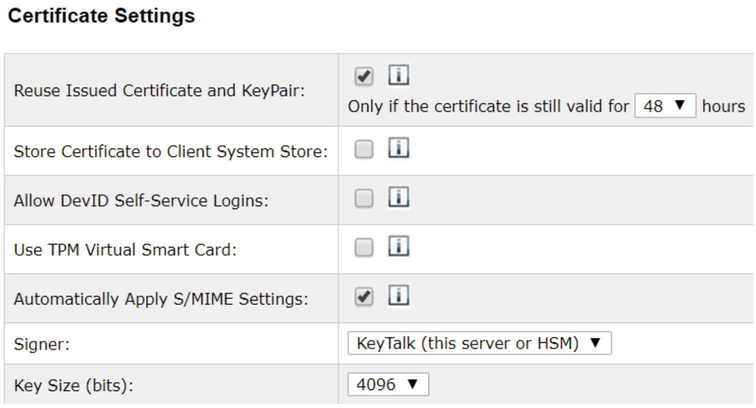


Add KeyTalk Service

Generic Settings

Service Name: *	My_SMIME_template
Required Credentials:	✓ USERID ✓ HWSIG ☑ PASSWD ☐ PIN ☐ RESPONSE
URI:	scheme://authority/path
File URI Digest:	sha256-executable-hash
Check URI:	☐
Execute Synchronously:	☐
HWSIG Formula:	3,5,9,10,11,12,13,17,18,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,599,601,603,604,606,607,608,609
Split Domain and UserId:	☐ Split on '\' ☐ Split on '@'
Comment:	

Step 3: Configure certificate settings

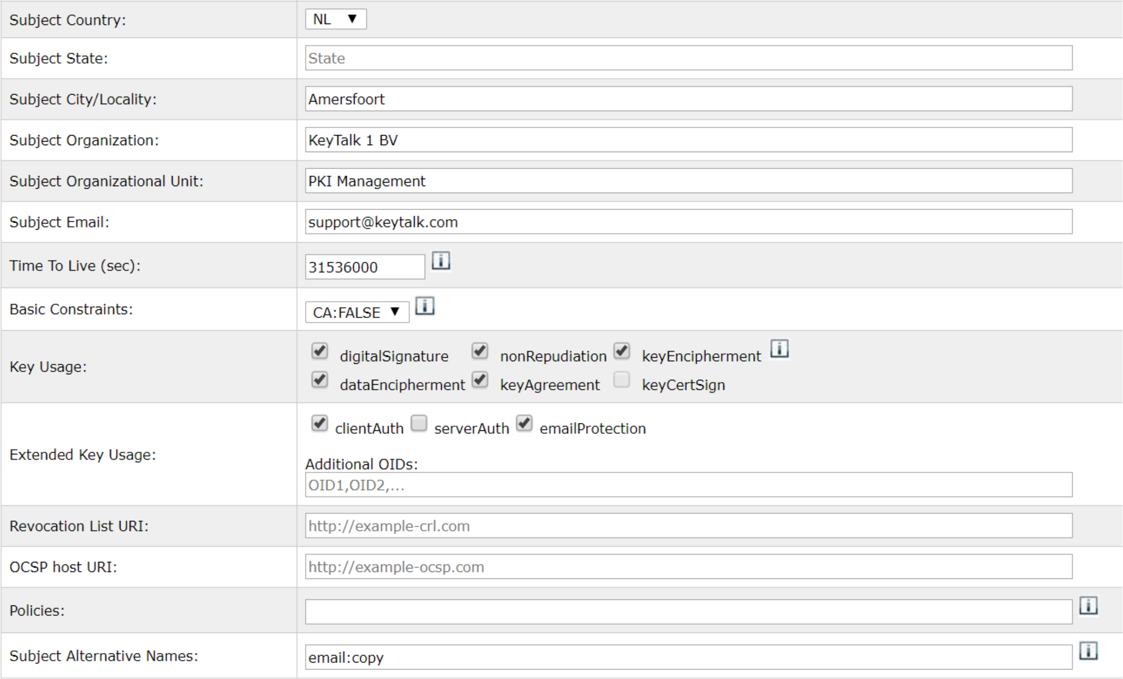


Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ Only if the certificate is still valid for 48 ▼ hours
Store Certificate to Client System Store:	☐
Allow DevID Self-Service Logins:	☐
Use TPM Virtual Smart Card:	☐
Automatically Apply S/MIME Settings:	☑
Signer:	KeyTalk (this server or HSM) ▼
Key Size (bits):	4096 ▼

- ✓ Ensure Reuse issued certificate and KeyPair is checked and set to 48 hours
- ✓ Ensure Auto Apply S/MIME Settings is checked
- ✓ Set your CA source and key-size

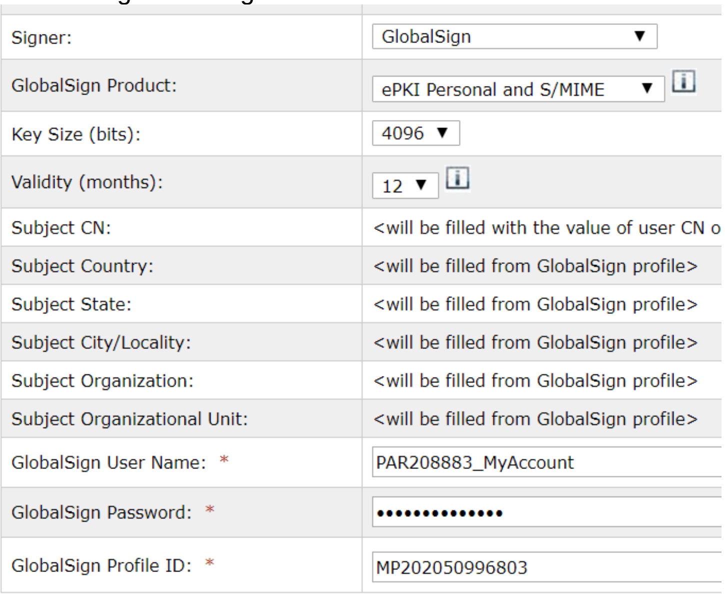
Step 4a: When using KeyTalk Private CA:



Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Time To Live (sec):	31536000 ⓘ
Basic Constraints:	CA:FALSE ▼ ⓘ
Key Usage:	☑ digitalSignature ☑ nonRepudiation ☑ keyEncipherment ⓘ ☑ dataEncipherment ☑ keyAgreement ☐ keyCertSign
Extended Key Usage:	☑ clientAuth ☐ serverAuth ☑ emailProtection Additional OIDs: OID1,OID2,...
Revocation List URI:	http://example-crl.com
OCSP host URI:	http://example-ocsp.com
Policies:	ⓘ
Subject Alternative Names:	email:copy ⓘ

- ✓ Optionally set the subject information to match your own company
- ✓ Set Time to Live to (sec): 31536000 (1 year), or use any other value you want.
- ✓ Set Basic Constaints to CA:FALSE
- ✓ Set Extended Key Usage to: clientAuth, emailProtection
- ✓ Set Subject Alternative Names to: email:copy (this ensure that your AD email address is copied into the SAN)

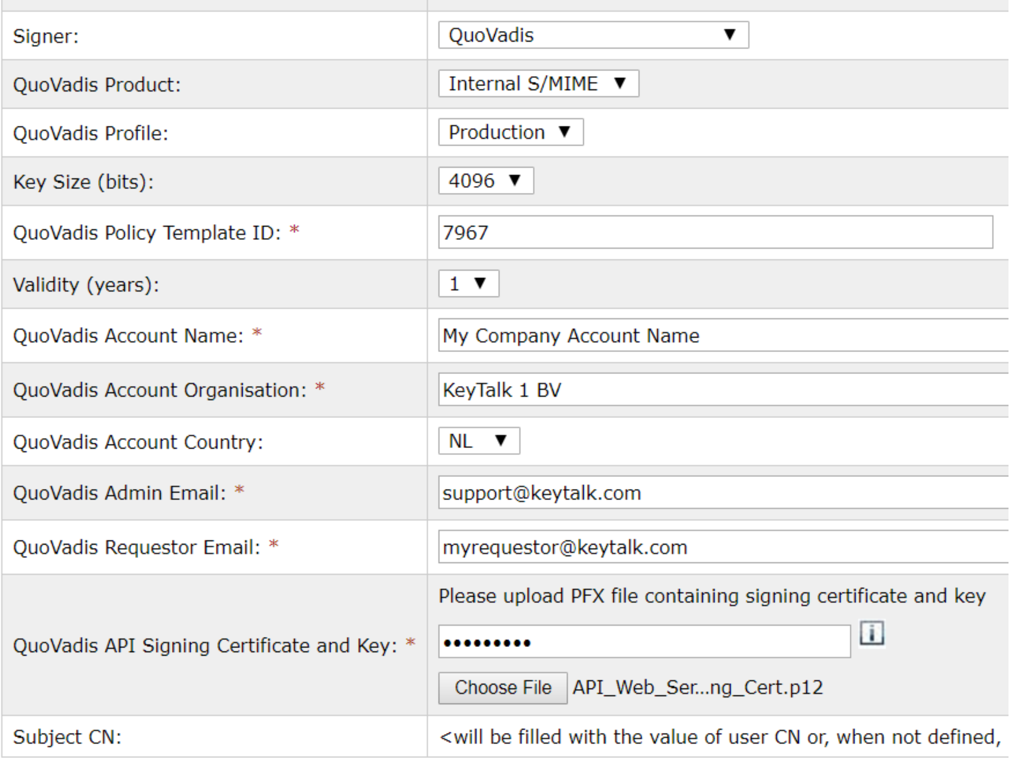
Step 4b: When using GlobalSign/TRUSTZONE:



Signer:	GlobalSign ▼
GlobalSign Product:	ePKI Personal and S/MIME ▼ ⓘ
Key Size (bits):	4096 ▼
Validity (months):	12 ▼ ⓘ
Subject CN:	<will be filled with the value of user CN o
Subject Country:	<will be filled from GlobalSign profile>
Subject State:	<will be filled from GlobalSign profile>
Subject City/Locality:	<will be filled from GlobalSign profile>
Subject Organization:	<will be filled from GlobalSign profile>
Subject Organizational Unit:	<will be filled from GlobalSign profile>
GlobalSign User Name: *	PAR208883_MyAccount
GlobalSign Password: *	••••••••••••••
GlobalSign Profile ID: *	MP202050996803

- ✓ Select product: ePKI Personal and S/MIME, or PersonalSign Business
- ✓ Select the validity period that matches your GlobalSign/TRUSTZONE license
- ✓ Set your GlobalSign/TRUSTZONE User Name
- ✓ Set your GlobalSign/TRUSTZONE Password
- ✓ Set your GlobalSign/TRUSTZONE Profile ID as found in the portal under ENTERPRISE PKI

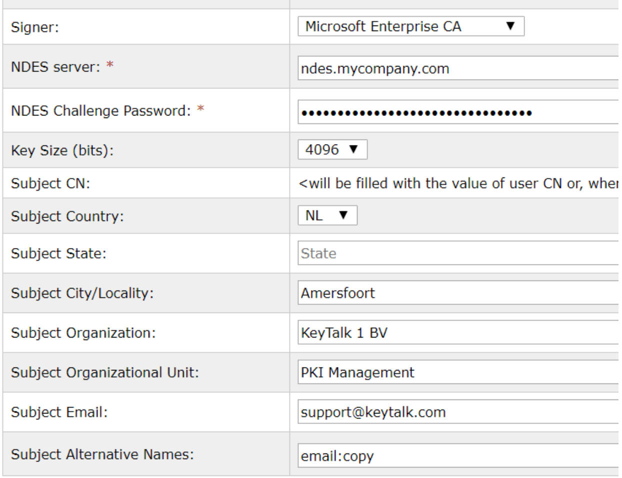
Step 4c: When using QuoVadis - DigiCert:



Signer:	QuoVadis ▼
QuoVadis Product:	Internal S/MIME ▼
QuoVadis Profile:	Production ▼
Key Size (bits):	4096 ▼
QuoVadis Policy Template ID: *	7967
Validity (years):	1 ▼
QuoVadis Account Name: *	My Company Account Name
QuoVadis Account Organisation: *	KeyTalk 1 BV
QuoVadis Account Country:	NL ▼
QuoVadis Admin Email: *	support@keytalk.com
QuoVadis Requestor Email: *	myrequestor@keytalk.com
QuoVadis API Signing Certificate and Key: *	Please upload PFX file containing signing certificate and key ••••••••• ⓘ Choose File API_Web_Ser...ng_Cert.p12
Subject CN:	<will be filled with the value of user CN or, when not defined,

- ✓ Select QuoVadis Product: Internal S/MIME
- ✓ Set your QuoVadis Policy Template ID
- ✓ Set your QuoVadis certificate validity period
- ✓ Set your QuoVadis Account Name
- ✓ Set your QuoVadis Account Organization
- ✓ Set your QuoVadis Account Country
- ✓ Set your QuoVadis Admin Email
- ✓ Set your QuoVadis Requestor Email
- ✓ Upload your QuoVadis API Signing Certificate and Key (p12 file)
- ✓ Upload your QuoVadis API Signing Certificate password

Step 4d: When using Microsoft AD Certificate Server:



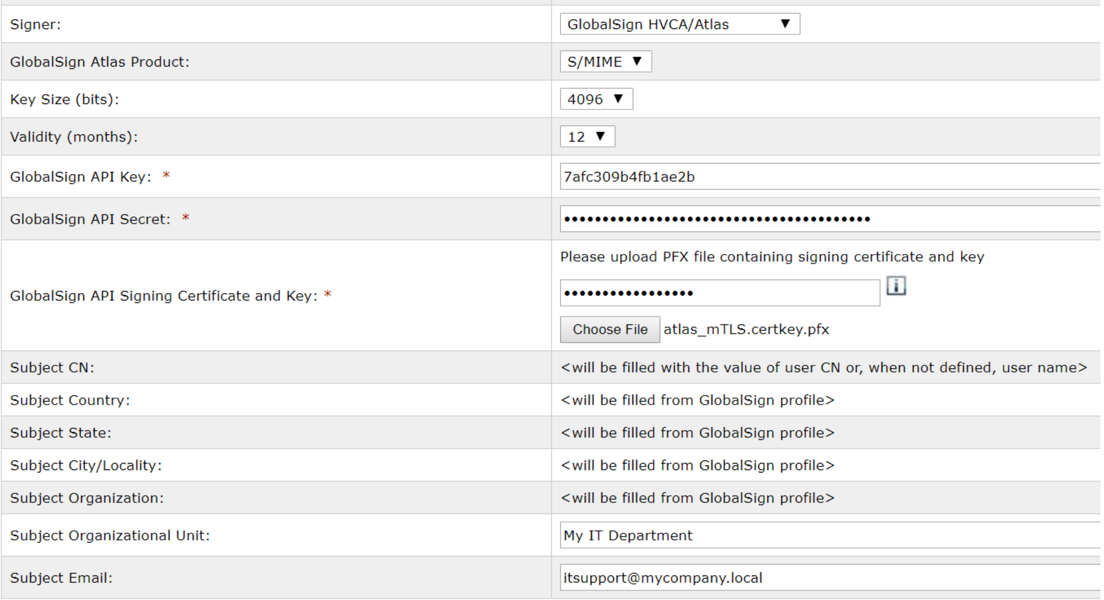
Signer:	Microsoft Enterprise CA ▼
NDES server: *	ndes.mycompany.com
NDES Challenge Password: *	•••••••••••••••••••••••••••••••
Key Size (bits):	4096 ▼
Subject CN:	<will be filled with the value of user CN or, wher
Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Subject Alternative Names:	email:copy

- ✓ Set your NDES server with a static challenge.
  If you haven't set NDES, follow https://www.youtube.com/watch?v=JOwoMJmgi2g

- ✓ Set Subject Alternative Names to: email:copy

  c

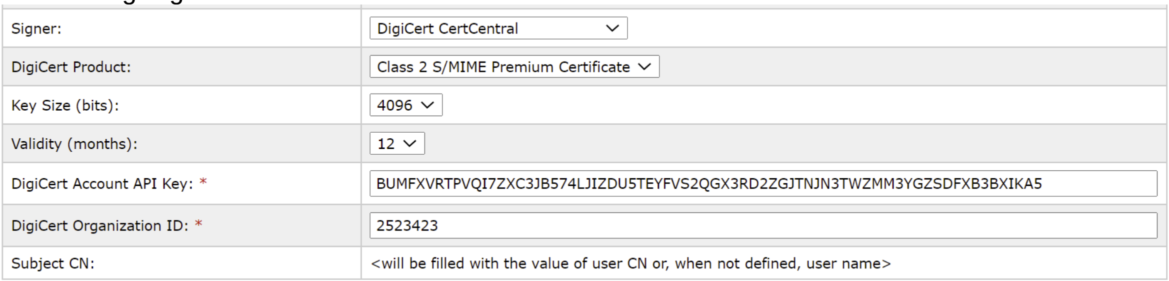
Step 4e: When using GlobalSign Atlas High Volume CA:



Signer:	GlobalSign HVCA/Atlas ▼
GlobalSign Atlas Product:	S/MIME ▼
Key Size (bits):	4096 ▼
Validity (months):	12 ▼
GlobalSign API Key: *	7afc309b4fb1ae2b
GlobalSign API Secret: *	••
GlobalSign API Signing Certificate and Key: *	Please upload PFX file containing signing certificate and key ••••••••••••••••• Choose File atlas_mTLS.certkey.pfx
Subject CN:	<will be filled with the value of user CN or, when not defined, user name>
Subject Country:	<will be filled from GlobalSign profile>
Subject State:	<will be filled from GlobalSign profile>
Subject City/Locality:	<will be filled from GlobalSign profile>
Subject Organization:	<will be filled from GlobalSign profile>
Subject Organizational Unit:	My IT Department
Subject Email:	itsupport@mycompany.local

- ✓ Select product: S/MIME
- ✓ Select the validity period that matches your GlobalSign Atlas license
- ✓ Set your GlobalSign Atlas API key
- ✓ Set your GlobalSign Atlas API secret
- ✓ Upload your GlobalSign Atlas API signing certificate and key
- ✓ Provide the GlobalSign Atlas API signing certificate and key install password
- ✓ Ensure the firewall allows traffic to emea.api.hvca.globalsign.com:8443

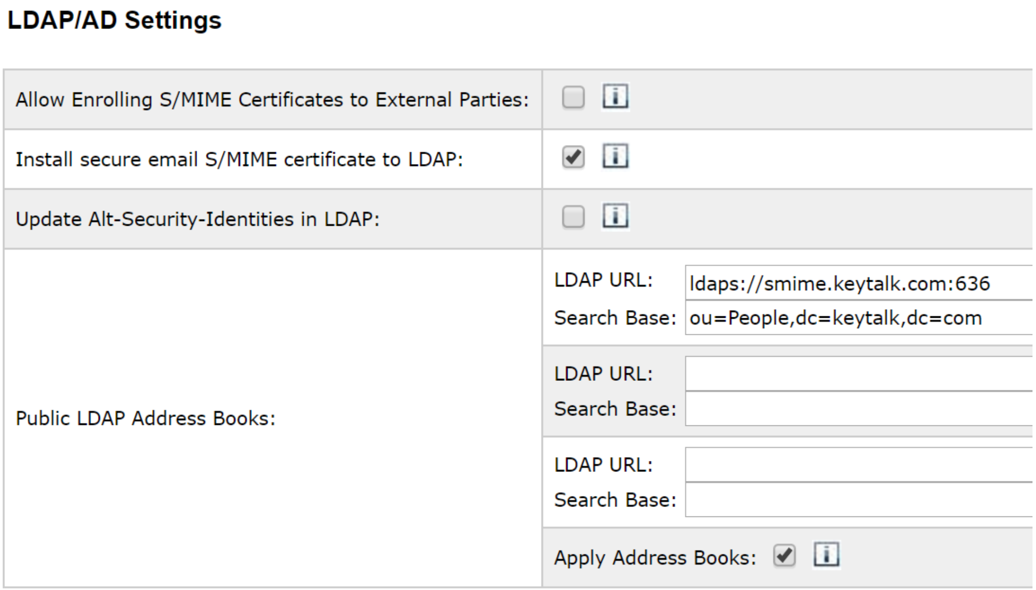
Step 4d: When using Digicert CertCentral:



Signer:	DigiCert CertCentral ⌄
DigiCert Product:	Class 2 S/MIME Premium Certificate ⌄
Key Size (bits):	4096 ⌄
Validity (months):	12 ⌄
DigiCert Account API Key: *	BUMFXVRTPVQI7ZXC3JB574LJIZDU5TEYFVS2QGX3RD2ZGJTNJN3TWZMM3YGZSDFXB3BXIKA5
DigiCert Organization ID: *	2523423
Subject CN:	<will be filled with the value of user CN or, when not defined, user name>

- ✓ Select product: Class 2 or Class 1 S/MIME
- ✓ Select the validity period that matches your CertCentral settings
- ✓ Set your DigiCert API key
- ✓ Set your DigiCert Org ID when applicable

Step 5: Configure writing issued S/MIME certificate to AD and/or LDAP:

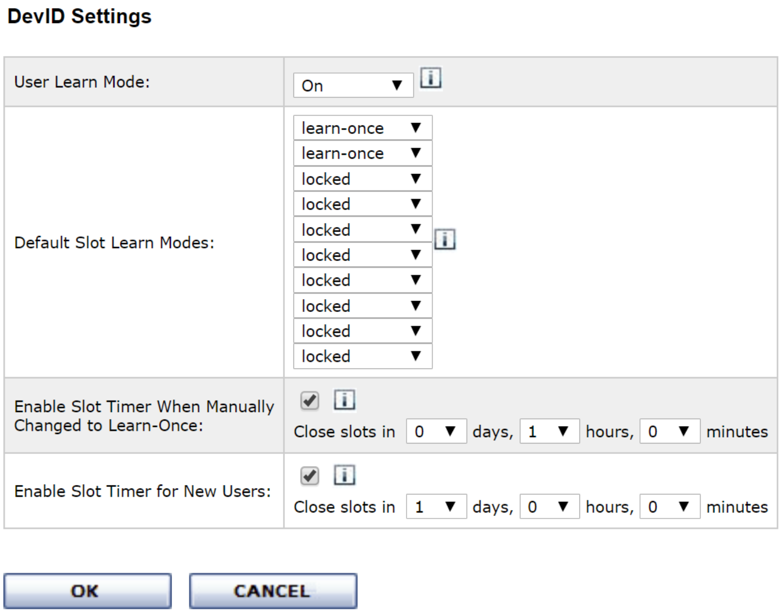


LDAP/AD Settings

Allow Enrolling S/MIME Certificates to External Parties:	□
Install secure email S/MIME certificate to LDAP:	☑
Update Alt-Security-Identities in LDAP:	□
Public LDAP Address Books:	LDAP URL: ldaps://smime.keytalk.com:636 Search Base: ou=People,dc=keytalk,dc=com LDAP URL: Search Base: LDAP URL: Search Base: Apply Address Books: ☑

- ✓ An issue S/MIME certificate can be written to your set AD as well as to your (KeyTalk) LDAP address book, by selecting: Install secure email S/MIME certificate to LDAP.
- ✓ Not required but often used: set your KeyTalk public LDAP URL and Search Base, so that the KeyTalk client can automatically configure it as an Outlook Address Book on Windows (requires Apply Address Books to be checked as well)

Step 6: Configure automated device learning mode, and amount of allowed devices.



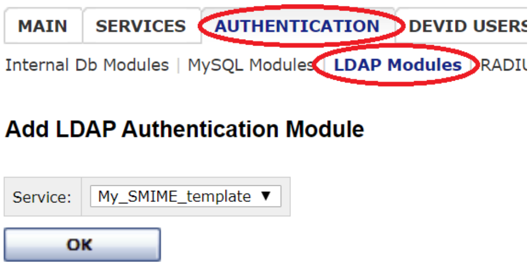
DevID Settings

User Learn Mode:	On
Default Slot Learn Modes:	learn-once learn-once locked locked locked locked locked locked locked locked
Enable Slot Timer When Manually Changed to Learn-Once:	☑ Close slots in 0 days, 1 hours, 0 minutes
Enable Slot Timer for New Users:	☑ Close slots in 1 days, 0 hours, 0 minutes

- ✓ Configure User Learn Mode to: On (this allows anyone with a valid AD account to use KeyTalk for certificate issuance)
- ✓ Configure slot 1 and 2 as : learn-once (this sets the first device to request a certificate to always be allowed, and enables a second device to also obtain the S/MIME certificate
- ✓ Set the timers. In the above example the second device is allowed to fetch a certificate within 24 hours after the first.

Step 7: Select OK to save the settings

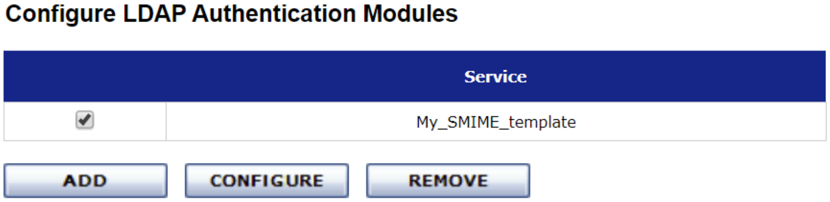
Step 8: Connect your Active Directory to the KeyTalk TEMPLATE certificate template



Add LDAP Authentication Module

Service:	My_SMIME_template ▼

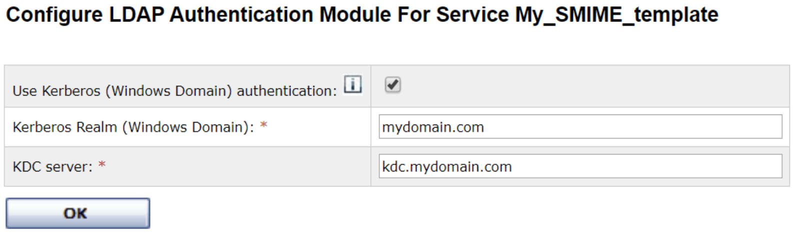
Step 9: Configure the added LDAP/AD connector



Configure LDAP Authentication Modules

	Service
☑	My_SMIME_template

Step 10: **Configure Kerberos**



Configure LDAP Authentication Module For Service My_SMIME_template

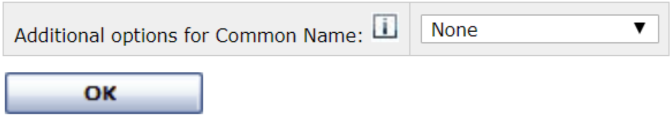
Use Kerberos (Windows Domain) authentication: ⓘ	☑
Kerberos Realm (Windows Domain): *	mydomain.com
KDC server: *	kdc.mydomain.com

- ✓ **Set your Windows domain**
- ✓ **Set your KDC server**

Save by selecting OK

When Kerberos is not configured, then the KeyTalk client/app will always ask for a username/password.

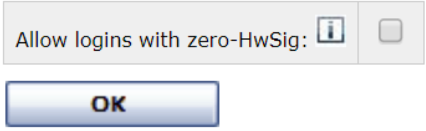
Step 11: Configure Common Name options



Additional options for Common Name: ⓘ	None ▼

- ✓ Set to None

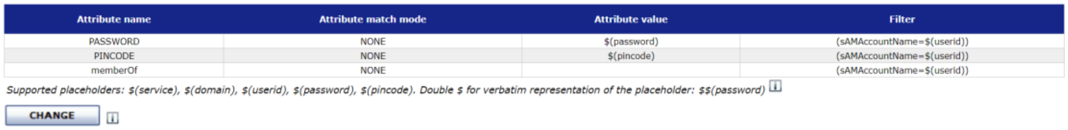
Step 12: Configure Hardware Signature requirement



Allow logins with zero-HwSig: ⓘ	□

- ✓ Leave unchecked (this enforces a non-zero value from a KeyTalk client/app when calculating a HardwareSignature)

Step 13: Configure AD/LDAP parameter Match Settings

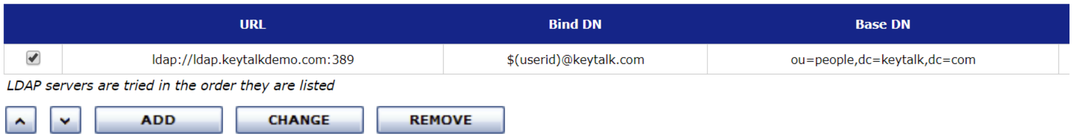


Attribute name	Attribute match mode	Attribute value	Filter
PASSWORD	NONE	$(password)	(sAMAccountName=$(userid))
PINCODE	NONE	$(pincode)	(sAMAccountName=$(userid))
memberOf	NONE		(sAMAccountName=$(userid))

Supported placeholders: $(service), $(domain), $(userid), $(password), $(pincode). Double $ for verbatim representation of the placeholder: $$(password)

Optionally configure this part to enforce a user being part of a specific Security Group (CHANGE -> memberof)

Step 14: Configure AD/LDAP connection settings

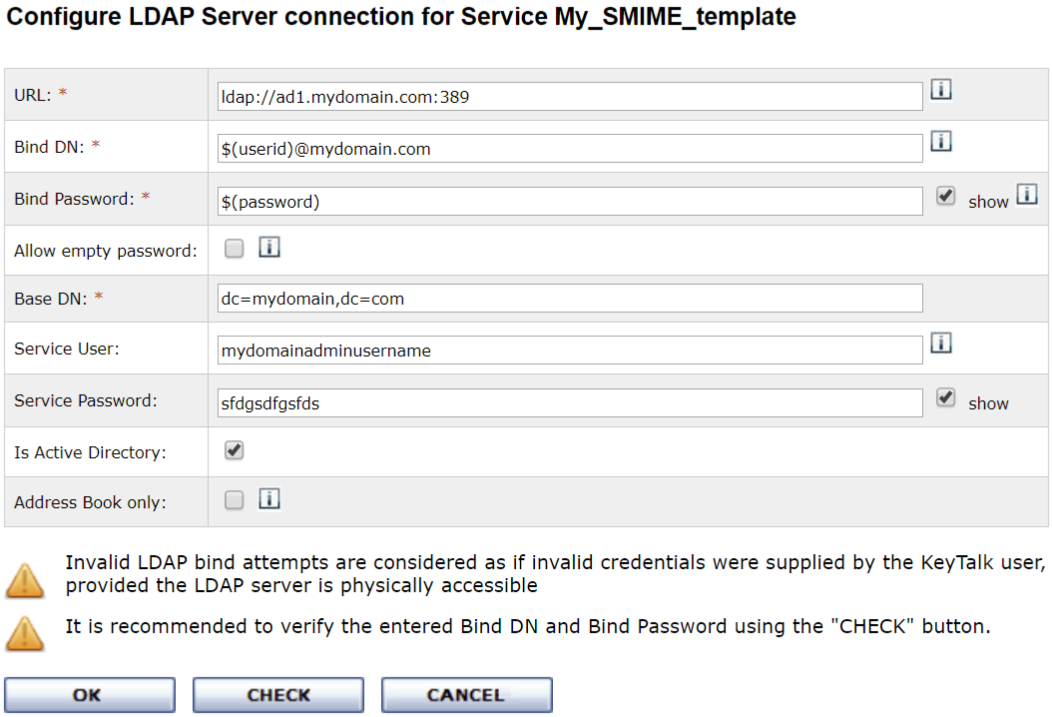


	URL	Bind DN	Base DN
☑	ldap://ldap.keytalkdemo.com:389	$(userid)@keytalk.com	ou=people,dc=keytalk,dc=com

LDAP servers are tried in the order they are listed

Select the default LDAP and select CHANGE

Step 15: Configure AD/LDAP



Configure LDAP Server connection for Service My_SMIME_template

URL: *	ldap://ad1.mydomain.com:389
Bind DN: *	$(userid)@mydomain.com
Bind Password: *	$(password) ☑ show
Allow empty password:	□
Base DN: *	dc=mydomain,dc=com
Service User:	mydomainadminusername
Service Password:	sfdgsdfgsfds ☑ show
Is Active Directory:	☑
Address Book only:	□

Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

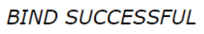
It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

Set your:

- ✓ AD URL using ldap:// and port 389
- ✓ Bind DN , its usually $(userid)@mydomain.com (substitute with your own domain)
- ✓ Bind Password set to $(password)
- ✓ Base DN to dc=mydomain,dc=com (substitute with your own domain and tld)
- ✓ TEMPLATE User: this is an AD domain admin capable of writing certificates to the attribute UserCertificate
- ✓ TEMPLATE Password: the password belonging to your set TEMPLATE User
- ✓ Checkmark Is Active Directory

Select OK to save

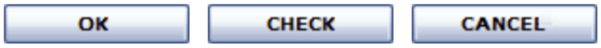
Step 16: Repeat step 14, now select CHECK to see if the connector works properly

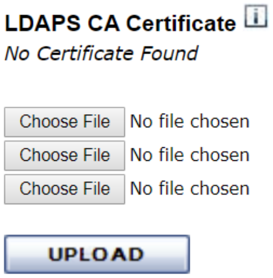


BIND SUCCESSFUL

If BIND fails, check the LOGS -> WEBUI LOG

Step 17: Configure secure LDAP/AD connection
Repeat step 14:





LDAPS CA Certificate

No Certificate Found

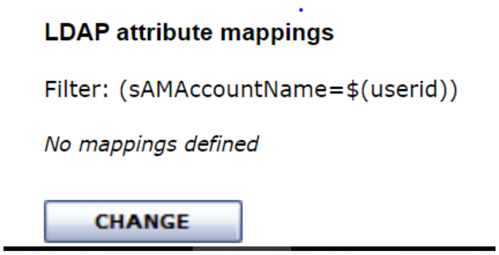
Choose File No file chosen
Choose File No file chosen
Choose File No file chosen
UPLOAD

✓ Upload in DER or PEM your AD/LDAP LDAPS issuing CA certificate.
This will result in KeyTalk server trusting your specific CA, enabling TLS over LDAPS

Step 18: Update your set AD/LDAP url to include LDAPS:// and port 636

Step 19: Repeat step 16 to verify if the LDAPS connection is working correctly

Step 20: Repeat step 9, scroll to the bottom of the page:



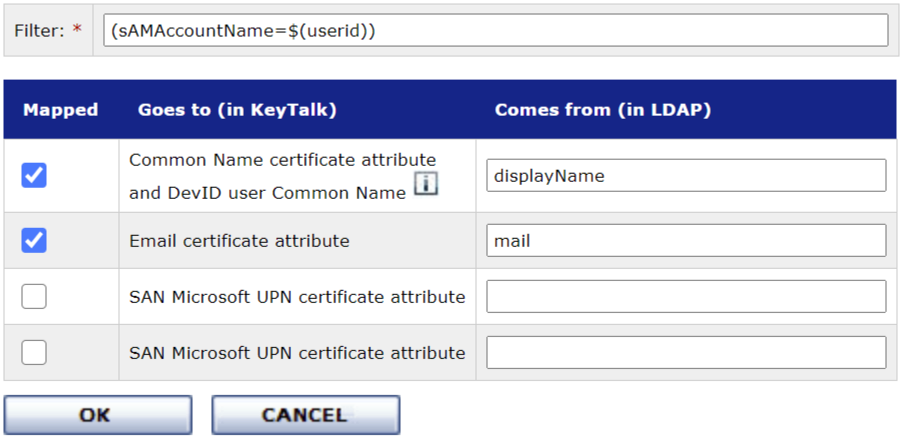
LDAP attribute mappings

Filter: (sAMAccountName=$(userid))

No mappings defined

Change the LDAP attribute mappings

Step 21: Configure AD attribute to certificate field mapping



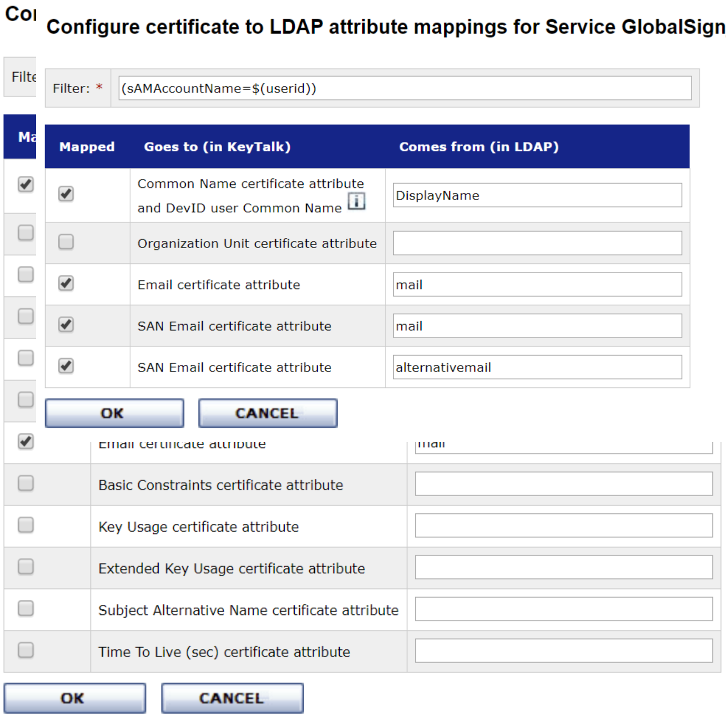
Filter: * (sAMAccountName=$(userid))

Mapped	Goes to (in KeyTalk)	Comes from (in LDAP)
☑	Common Name certificate attribute and DevID user Common Name	displayName
☑	Email certificate attribute	mail
☐	SAN Microsoft UPN certificate attribute	
☐	SAN Microsoft UPN certificate attribute	

The above shows when using a Public CA such as QuoVadis, TRUSTZONE, or GlobalSign.

- ✓ Set your AD filter to match the attribute used as the username (often sAMAccountName)
- ✓ Set Email certificate attribute to : mail (your AD mail attribute)
- ✓ Set Common Name certificate attribute to: DisplayName (not setting it is also an option then the username of the user is used as the issued S/MIME certificate CN value

Alternatively, the chosen CA provider might support more elaborate SAN certificate mappings to support for example multiple email addresses:



Configure certificate to LDAP attribute mappings for Service GlobalSign_Atlas

Filter: * (sAMAccountName=$(userid))

Mapped	Goes to (in KeyTalk)	Comes from (in LDAP)
☑	Common Name certificate attribute and DevID user Common Name	DisplayName
□	Organization Unit certificate attribute	
☑	Email certificate attribute	mail
☑	SAN Email certificate attribute	mail
☑	SAN Email certificate attribute	alternativemail

☑	Email certificate attribute	mail
□	Basic Constraints certificate attribute	
□	Key Usage certificate attribute	
□	Extended Key Usage certificate attribute	
□	Subject Alternative Name certificate attribute	
□	Time To Live (sec) certificate attribute	

The above shows when using a Private CA such as KeyTalk or Microsoft AD CS.

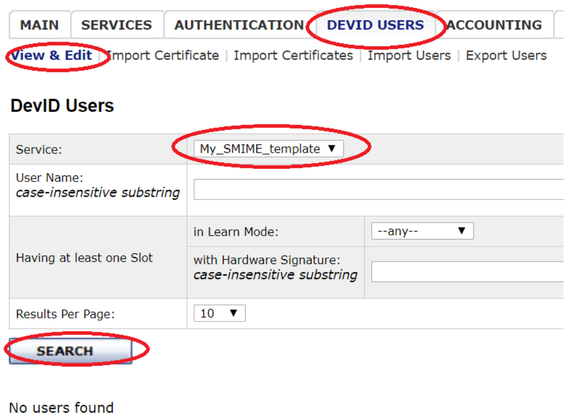
- ✓ Set your AD filter to match the attribute used as the username (often sAMAccountName)
- ✓ Set Email certificate attribute to: mail (your AD mail attribute)
- ✓ Set Common Name certificate attribute to: DisplayName (not setting it is also an option then the username of the user is used as the issued S/MIME certificate CN value

Select OK to save the settings.

Step 22a: The configuration is now ready to issue S/MIME certificates to any user matching the configured AD connector criteria.

The following manual steps are used to verify if automated certificate issuance works properly without deploying the KeyTalk client with its configuration file.

Go to DEVID USERS (the location where all issued certificates are administered for every user and end-point) :

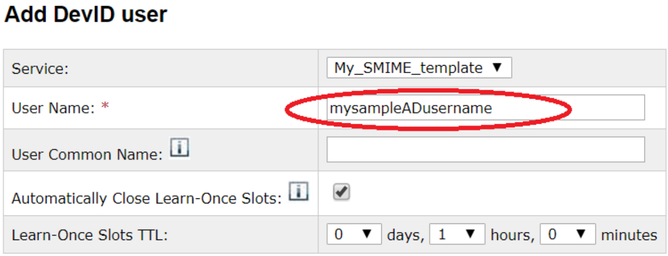


DevID Users

Service:	My_SMIME_template ▼		
User Name: *case-insensitive substring*			
Having at least one Slot	in Learn Mode:	--any-- ▼	
	with Hardware Signature: *case-insensitive substring*		
Results Per Page:	10 ▼		

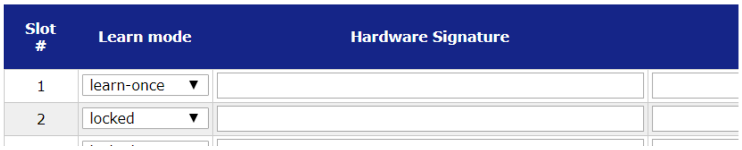
No users found

Step 22b: Add an existing AD user:



Add DevID user

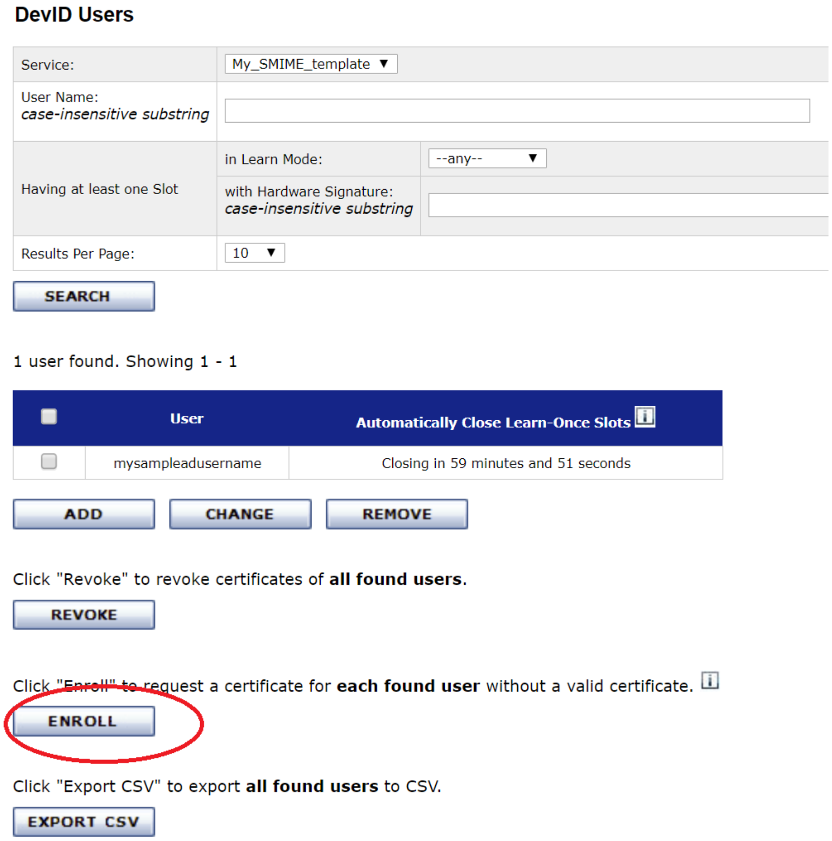
Service:	My_SMIME_template ▼
User Name: *	mysampleADusername
User Common Name: ℹ	
Automatically Close Learn-Once Slots: ℹ	☑
Learn-Once Slots TTL:	0 ▼ days, 1 ▼ hours, 0 ▼ minutes



Slot #	Learn mode	Hardware Signature	
1	learn-once ▼		
2	locked ▼		

And select OK

Step 22c: Enroll a certificate to the manual added user without a certificate:



DevID Users

Service:	My_SMIME_template ▼	
User Name: *case-insensitive substring*		
Having at least one Slot	in Learn Mode:	--any-- ▼
	with Hardware Signature: *case-insensitive substring*	
Results Per Page:	10 ▼	

1 user found. Showing 1 - 1

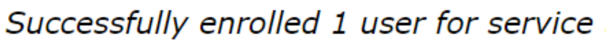
□	User	Automatically Close Learn-Once Slots ℹ
□	mysampleadusername	Closing in 59 minutes and 51 seconds

Click "Revoke" to revoke certificates of **all found users**.

Click "Enroll" to request a certificate for **each found user** without a valid certificate. ℹ

Click "Export CSV" to export **all found users** to CSV.

When successful you will see:

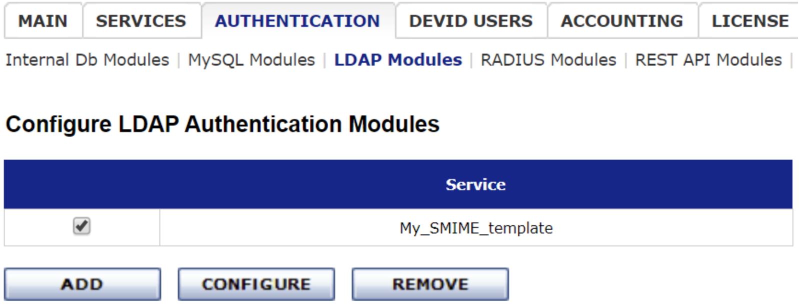


Successfully enrolled 1 user for service

If unsuccessful, check both CAD log and WEBUI log to see why it failed

Step 23a: Now configure KeyTalk to also write certificates to the optional KeyTalk LDAP Address Book.
Should you wish to skip the writing of S/MIME to the LDAP KeyTalk Address Book, continue to **step 24**.

Configure your LDAP/AD connector again:



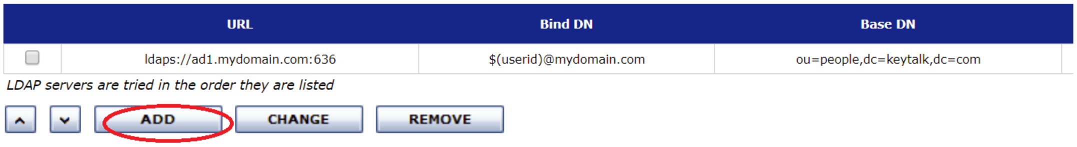
MAIN | SERVICES | **AUTHENTICATION** | DEVID USERS | ACCOUNTING | LICENSE

Internal Db Modules | MySQL Modules | **LDAP Modules** | RADIUS Modules | REST API Modules |

Configure LDAP Authentication Modules

	Service
☑	My_SMIME_template

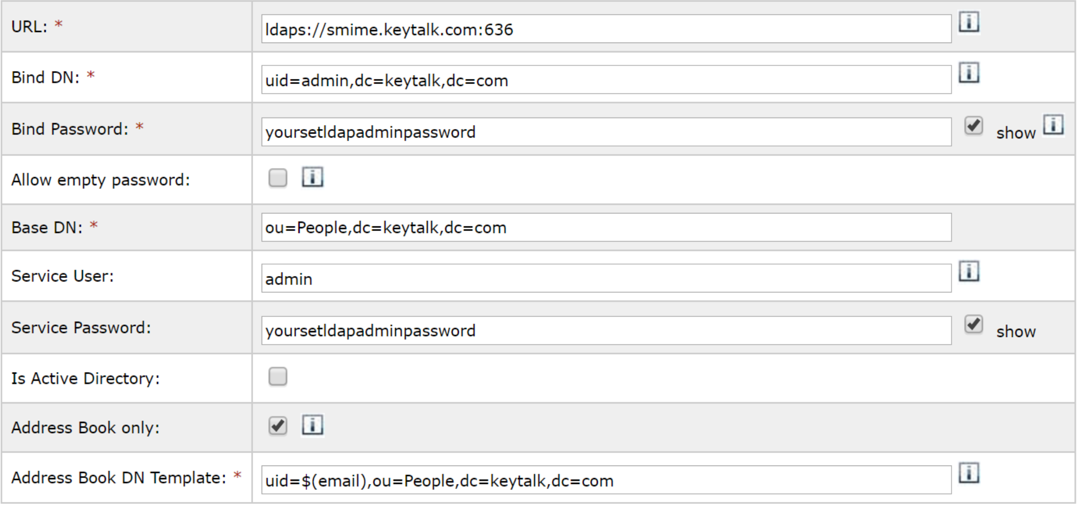
Step 23b: ADD an LDAP



	URL	Bind DN	Base DN
□	ldaps://ad1.mydomain.com:636	$(userid)@mydomain.com	ou=people,dc=keytalk,dc=com

LDAP servers are tried in the order they are listed

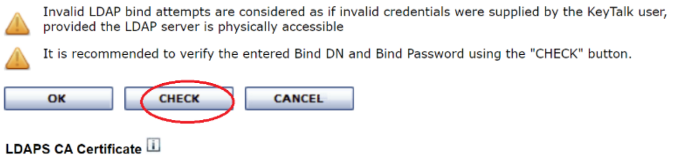
Step 23c: Configure the public KeyTalk S/MIME LDAP Address Book



URL: *	ldaps://smime.keytalk.com:636
Bind DN: *	uid=admin,dc=keytalk,dc=com
Bind Password: *	yoursetldapadminpassword ☑ show
Allow empty password:	□
Base DN: *	ou=People,dc=keytalk,dc=com
Service User:	admin
Service Password:	yoursetldapadminpassword ☑ show
Is Active Directory:	□
Address Book only:	☑
Address Book DN Template: *	uid=$(email),ou=People,dc=keytalk,dc=com

- ✓ Set your LDAP Addressbook url. Use ldap:// with port :389
- ✓ Set the Bind DN to: uid=admin,dc=keytalk,dc=com
- ✓ Set the Bind Password to the LDAP Admin password you configured on the LDAP
- ✓ Set the Base DN to: ou=People,dc=keytalk,dc=com
- ✓ Set TEMPLATE User to: admin
- ✓ Set TEMPLATE Password to: the LDAP Admin password you configured on the LDAP
- ✓ Check: Address Book only

Verify if the connection works using the CHECK button

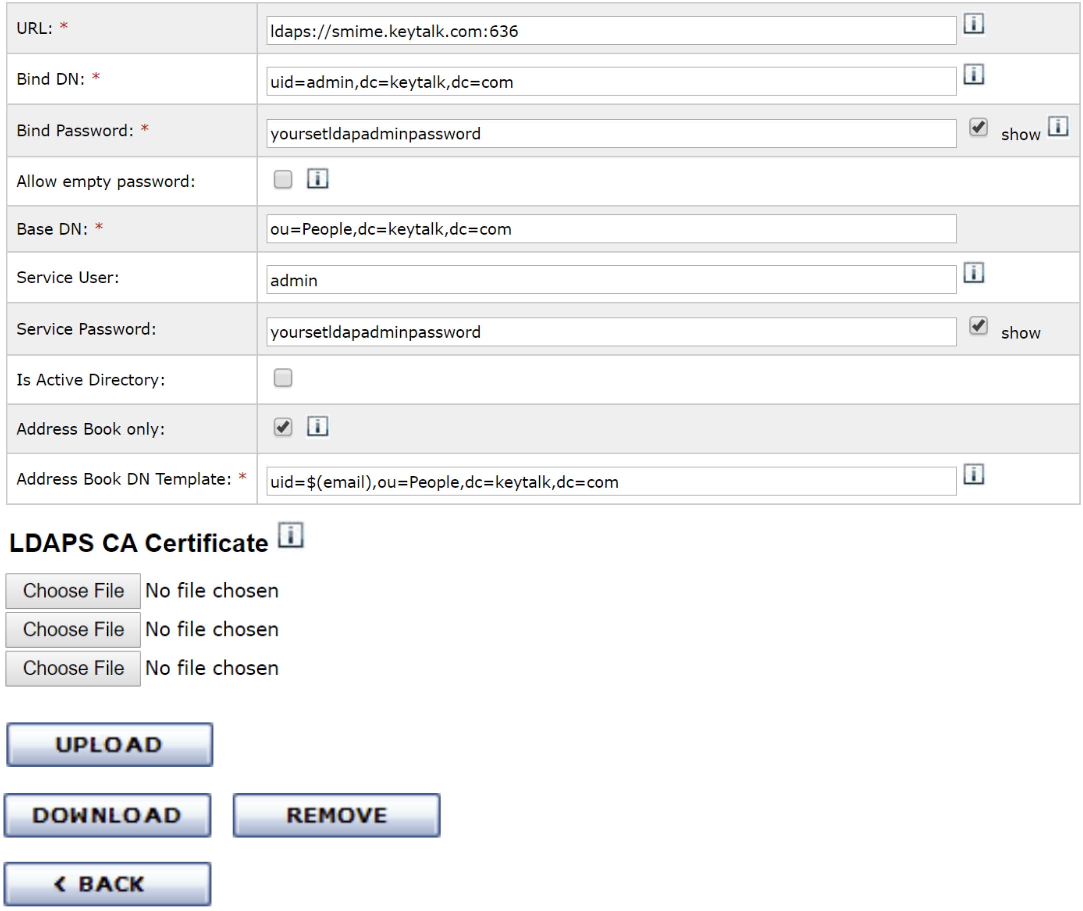


Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

LDAPS CA Certificate

Step 23d: Configure LDAPS for the Address Book



URL: *	ldaps://smime.keytalk.com:636
Bind DN: *	uid=admin,dc=keytalk,dc=com
Bind Password: *	yoursetldapadminpassword ☑ show
Allow empty password:	□
Base DN: *	ou=People,dc=keytalk,dc=com
Service User:	admin
Service Password:	yoursetldapadminpassword ☑ show
Is Active Directory:	□
Address Book only:	☑
Address Book DN Template: *	uid=$(email),ou=People,dc=keytalk,dc=com

LDAPS CA Certificate

Choose File No file chosen

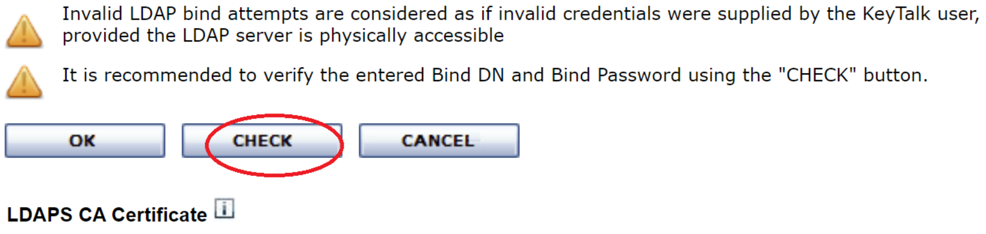
Choose File No file chosen

Choose File No file chosen

While its not required to configure TLS, it is advised , as it will secure your connection.

- ✓ Change your LDAP Addressbook url to use use ldaps:// with port :636
- ✓ Upload your KeyTalk LDAP SSL certificate issuing CA to create a trust for the KeyTalk server

Now verify if the LDAPS connection works properly using the CHECK button.

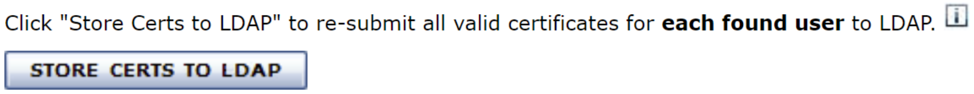


Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

LDAPS CA Certificate

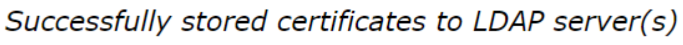
Step 23e: Verify if the certificate write to LDAP Address Book works properly
Go to: DEVID USERS, and select your S/MIME TEMPLATE template



Click "Store Certs to LDAP" to re-submit all valid certificates for **each found user** to LDAP.

- ✓ Select: STORE CERTS TO LDAP

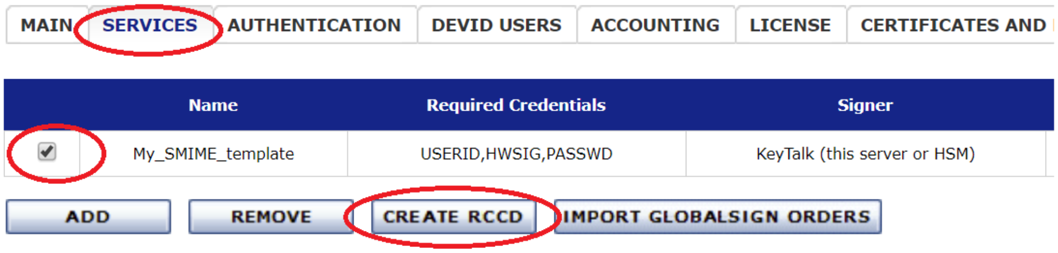
When it works properly you will see:



Successfully stored certificates to LDAP server(s)

If unsuccessful, check both AuthD log and WEBUI log to see why it failed

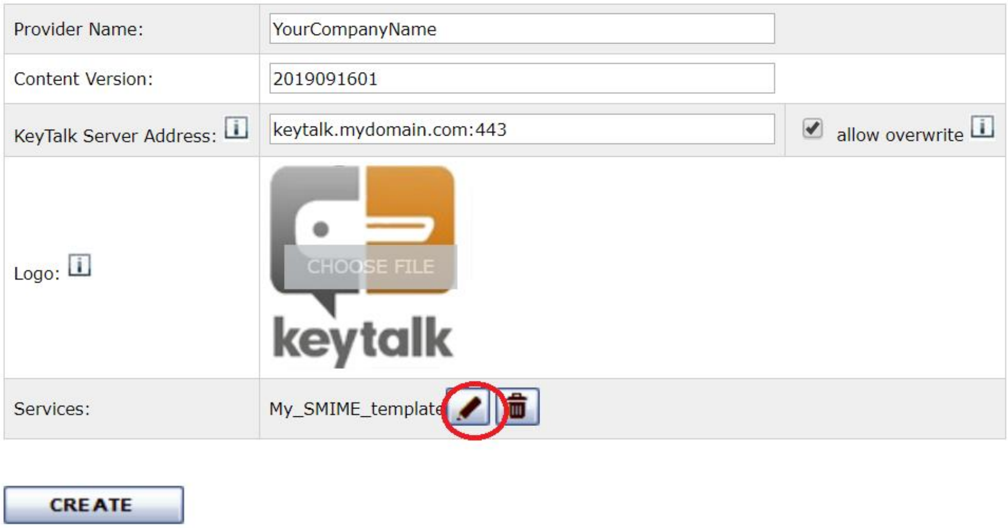
Step 24a: Create KeyTalk Real Client Configuration Data file.
This file is used to configure the KeyTalk agent



	Name	Required Credentials	Signer
☑	My_SMIME_template	USERID,HWSIG,PASSWD	KeyTalk (this server or HSM)

- ✓ Select the TEMPLATE you wish to include in the configuration file
- ✓ Select CREATE RCCD

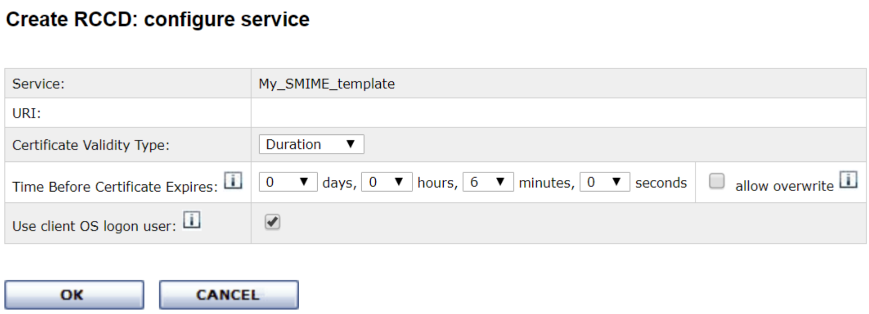
Step 24b: Create KeyTalk Real Client Configuration Data file.
This file is used to configure the KeyTalk agent



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address:	keytalk.mydomain.com:443	☑ allow overwrite
Logo:		
Services:	My_SMIME_template	

- ✓ Set KeyTalk Provider Name (this is the name under which multiple KeyTalk configurations are grouped when imported into a single client)
- ✓ Set KeyTalk Server Address. This is either the FQDN or the IP address. Ensure the this address is also present in the KeyTalk “Certificates and Keys” in the SAN of the “Client-Server certificate” ”(generic TLS use for most KeyTalk clients), and when applicable in the KeyTalk “Certificates and Keys” in the SAN of the “Trusted Mobile SSL” (required for Apple and Android devices, and usable for generic KeyTalk client TLS purposes)
- ✓ Set the port to 443 (generic use for KeyTalk client/app), or optionally to port 4443 when a Trusted Mobile SSL certificate and key has been uploaded.
- ✓ Optionally change the Logo to a 110x110 PNG
- ✓ Select the EDIT button next to the selected TEMPLATE

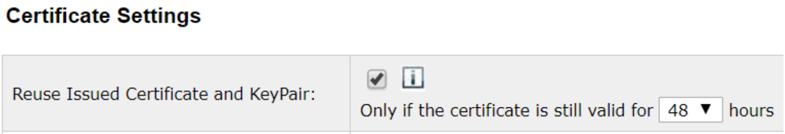
Step 24c: Configure KeyTalk Real Client Configuration Data meta data



Create RCCD: configure service

Service:	My_SMIME_template	
URI:		
Certificate Validity Type:	Duration ▼	
Time Before Certificate Expires:	0 ▼ days, 0 ▼ hours, 6 ▼ minutes, 0 ▼ seconds	□ allow overwrite
Use client OS logon user:	☑	

- ✓ Set the duration time, which determines the amount of time before the certificate needs to be replaced before it expires. This time should not exceed the TEMPLATE template time configured in:



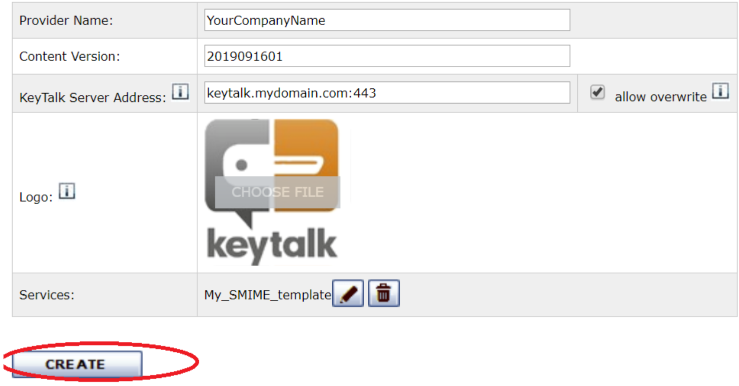
Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ Only if the certificate is still valid for 48 ▼ hours

- ✓ Set : Use client OS logon user. This ensures that the KeyTalk client/app uses the Windows user that is logged in. Unchecking it requires the user to enter their own username
- ✓ Set Allow Overwrite. Setting this option ensures a Windows user cannot easily remove the configuration data

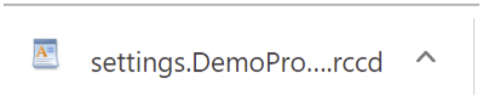
Select OK to save

Step 24d: Finalize the KeyTalk configuration file



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address:	keytalk.mydomain.com:443	☑ allow overwrite
Logo:	CHOOSE FILE	
Services:	My_SMIME_template	

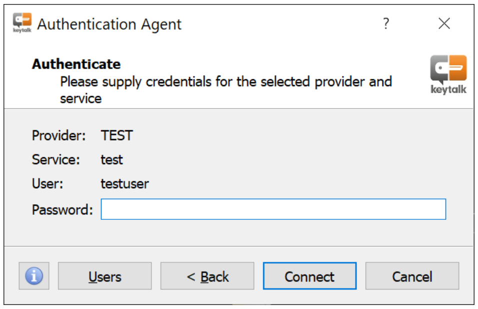
Step 24e: Rename the created RCCD file to a name you find suitable/shorter:



settings.DemoPro....rccd

This RCCD file is effectively a ZIP file, you can rename it to inspect its contents.
Its typically deployed together with the KeyTalk client (see the individual KeyTalk client/app manuals), or it is email to users as an attachment, or made available as a public download (the file does not contain any secret information)

Step 25: Authenticate from an installed and configured KeyTalk client using Kerberos, or username/password



Authentication Agent

Authenticate

Please supply credentials for the selected provider and service

Provider: TEST

Service: test

User: testuser

Password:

Note1: Kerberos is currently only supported in Windows 7-10, and Server 2012R2-2019

Note2: With each KeyTalk installation two Windows Scheduler Tasks are activated, which are used to periodically verify if certificate replacement needs to trigger.

Notes:

- ➢ While KeyTalk cannot provide support on software of other vendors, several KeyTalk customers have indicated the following mail clients work just fine with S/MIME and most also work with integrated LDAP Address Books, albeit these mail clients may require some manual configurations:
  - ✓ Windows: Outlook
  - ✓ Windows: Thunderbird
  - ✓ Linux: Thunderbird
  - ✓ Linux: Evolution
  - ✓ MacOSX: MacMail
  - ✓ MacOSX: Outlook for Mac
  - ✓ MacOSX: Thunderbird
  - ✓ iOS: Mail
  - ✓ iOS: MobileIron Email+ (requires MobileIron)
  - ✓ iOS: Outlook (requires Intune!)
  - ✓ Android: Samsung mail
  - ✓ Android: Ciphermail
  - ✓ Android: Nine - Email

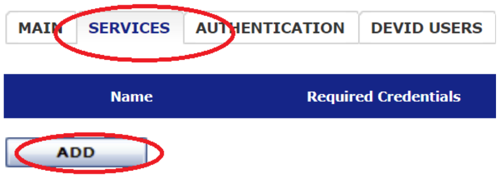
ANNEX D: Sample Third party S/MIME use-case configuration

This use-case sample assumes:

- ✓ the KeyTalk server has been configured already for the issuance of S/MIME certificates which also support client authentication (see ANNEX C)
- ✓ an LDAP Address Book is available. Do note that an LDAP Address Book is REQUIRED (but does not need to be the KeyTalk S/MIME LDAP as provided as part of the KeyTalk solution)

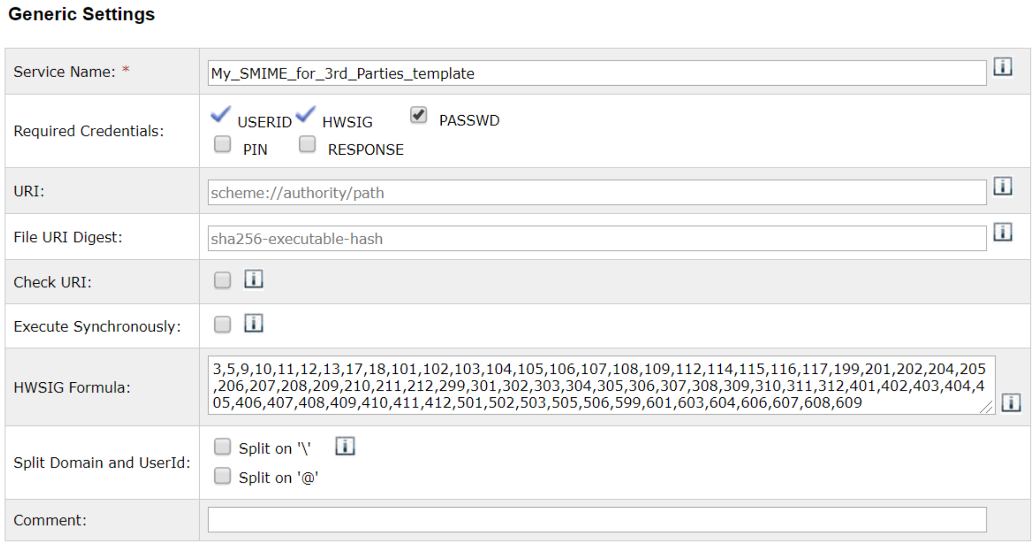
In this sample configuration, we'll go step by step over enabling the ability to request certificates for third party email addresses (ie people outside your company), using the KeyTalk Self TEMPLATE Portal.

Step 1: Configure a KeyTalk certificate template under TEMPLATES



Name	Required Credentials

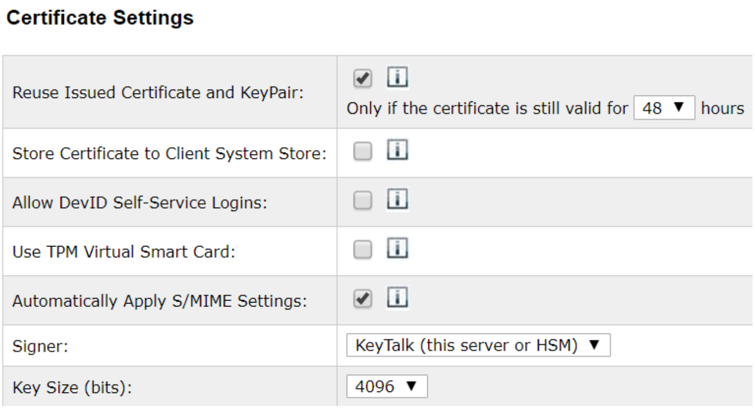
Step 2: Configure the TEMPLATE general settings, and ensure you select password. Optionally make appropriate changes to the HWSIG Formula hardware/software characteristics.



Generic Settings

Service Name: *	My_SMIME_for_3rd_Parties_template
Required Credentials:	✓ USERID ✓ HWSIG ☑ PASSWD □ PIN □ RESPONSE
URI:	scheme://authority/path
File URI Digest:	sha256-executable-hash
Check URI:	□
Execute Synchronously:	□
HWSIG Formula:	3,5,9,10,11,12,13,17,18,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,599,601,603,604,606,607,608,609
Split Domain and UserId:	□ Split on '\' □ Split on '@'
Comment:	

Step 3: Configure certificate settings

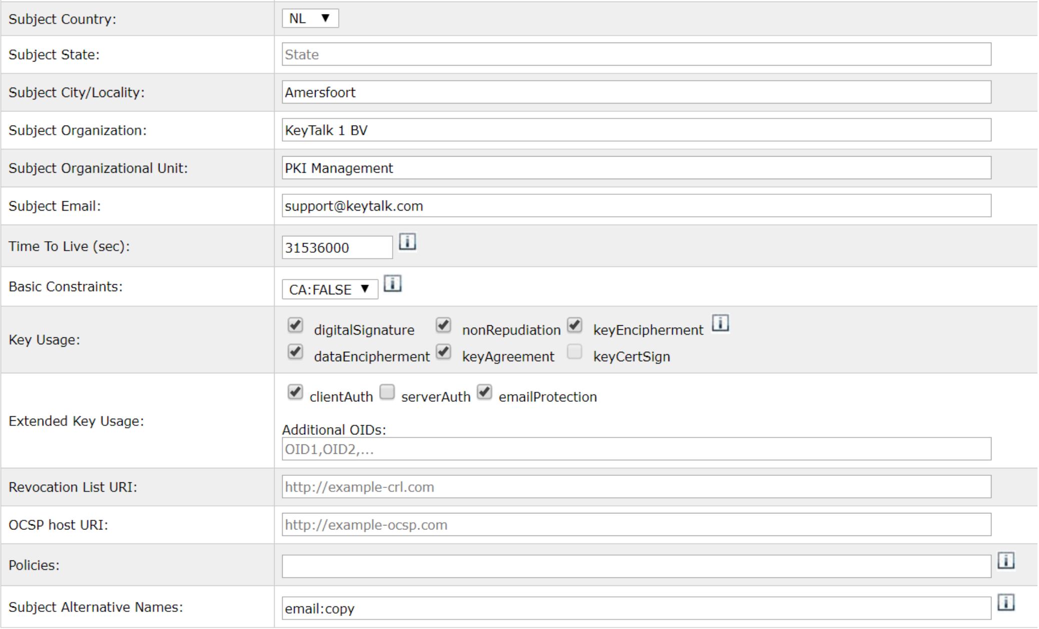


Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ Only if the certificate is still valid for 48 ▼ hours
Store Certificate to Client System Store:	□
Allow DevID Self-Service Logins:	□
Use TPM Virtual Smart Card:	□
Automatically Apply S/MIME Settings:	☑
Signer:	KeyTalk (this server or HSM) ▼
Key Size (bits):	4096 ▼

- ✓ Ensure Reuse issued certificate and KeyPair is checked and set to any value
- ✓ Optionally set Auto Apply S/MIME Settings is checked
- ✓ Set your CA source and key-size

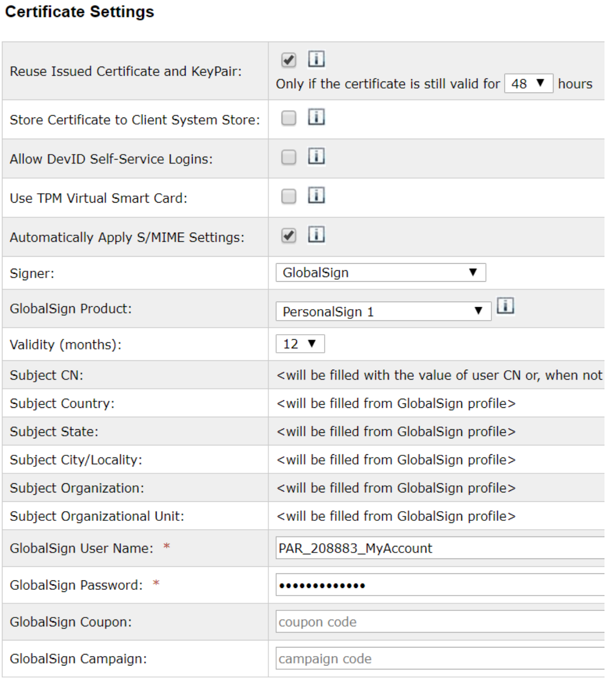
Step 4a: When using KeyTalk Private CA:



Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Time To Live (sec):	31536000 ℹ
Basic Constraints:	CA:FALSE ▼ ℹ
Key Usage:	☑ digitalSignature ☑ nonRepudiation ☑ keyEncipherment ℹ ☑ dataEncipherment ☑ keyAgreement ☐ keyCertSign
Extended Key Usage:	☑ clientAuth ☐ serverAuth ☑ emailProtection Additional OIDs: OID1,OID2,...
Revocation List URI:	http://example-crl.com
OCSP host URI:	http://example-ocsp.com
Policies:	ℹ
Subject Alternative Names:	email:copy ℹ

- ✓ Optionally set the subject information to match your own company. Arguably he third party will not represent your company, so often its left empty
- ✓ Set Time to Live to (sec): 31536000 (1 year), or use any other value you want.
- ✓ Set Basic Constaints to CA:FALSE
- ✓ Set Extended Key Usage to: clientAuth, emailProtection
- ✓ Set Subject Alternative Names to: email:copy (this ensures that the used email address is copied into the SAN)

Step 4b: When using GlobalSign/TRUSTZONE:



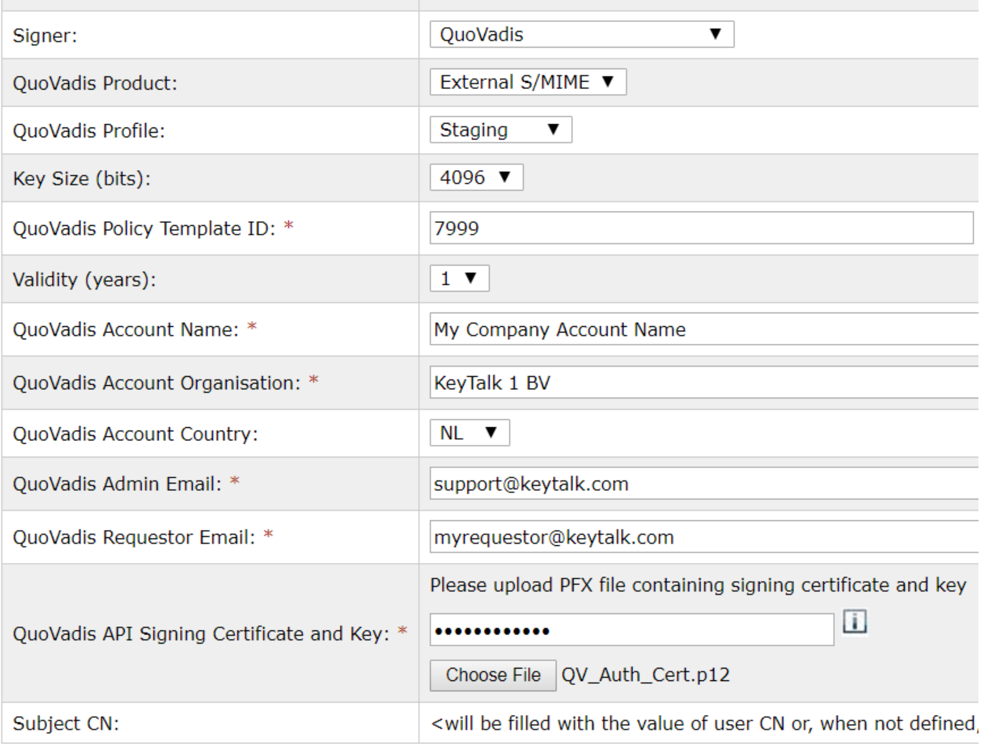
Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ ℹ Only if the certificate is still valid for 48 ▼ hours
Store Certificate to Client System Store:	☐ ℹ
Allow DevID Self-Service Logins:	☐ ℹ
Use TPM Virtual Smart Card:	☐ ℹ
Automatically Apply S/MIME Settings:	☑ ℹ
Signer:	GlobalSign ▼
GlobalSign Product:	PersonalSign 1 ▼ ℹ
Validity (months):	12 ▼
Subject CN:	<will be filled with the value of user CN or, when not
Subject Country:	<will be filled from GlobalSign profile>
Subject State:	<will be filled from GlobalSign profile>
Subject City/Locality:	<will be filled from GlobalSign profile>
Subject Organization:	<will be filled from GlobalSign profile>
Subject Organizational Unit:	<will be filled from GlobalSign profile>
GlobalSign User Name: *	PAR_208883_MyAccount
GlobalSign Password: *	•••••••••••••
GlobalSign Coupon:	coupon code
GlobalSign Campaign:	campaign code

- ✓ Select product: PersonalSign 1, or PersonalSign Express
- ✓ Select the validity period that matches your GlobalSign/TRUSTZONE license
- ✓ Set your GlobalSign/TRUSTZONE User Name
- ✓ Set your GlobalSign/TRUSTZONE Password

- ✓ Set your GlobalSign/TRUSTZONE Profile ID as found in the portal under ENTERPRISE PKI

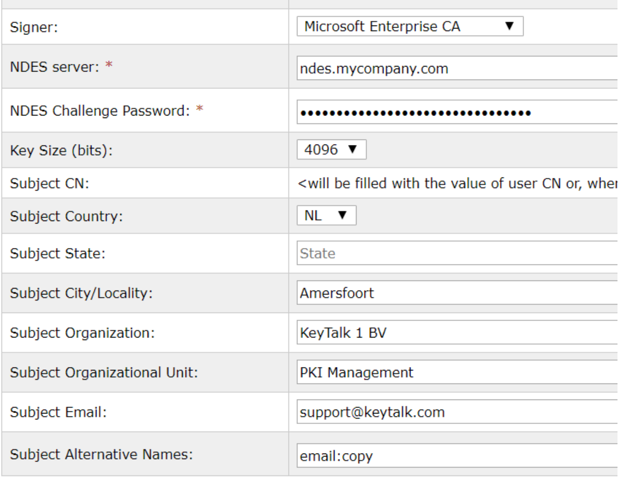
Step 4c: When using QuoVadis - DigiCert:



Signer:	QuoVadis ▼
QuoVadis Product:	External S/MIME ▼
QuoVadis Profile:	Staging ▼
Key Size (bits):	4096 ▼
QuoVadis Policy Template ID: *	7999
Validity (years):	1 ▼
QuoVadis Account Name: *	My Company Account Name
QuoVadis Account Organisation: *	KeyTalk 1 BV
QuoVadis Account Country:	NL ▼
QuoVadis Admin Email: *	support@keytalk.com
QuoVadis Requestor Email: *	myrequestor@keytalk.com
QuoVadis API Signing Certificate and Key: *	Please upload PFX file containing signing certificate and key •••••••••••• Choose File QV_Auth_Cert.p12
Subject CN:	<will be filled with the value of user CN or, when not defined,

- ✓ Select QuoVadis Product: External S/MIME
- ✓ Set your QuoVadis Policy Template ID
- ✓ Set your QuoVadis certificate validity period
- ✓ Set your QuoVadis Account Name
- ✓ Set your QuoVadis Account Organization
- ✓ Set your QuoVadis Account Country
- ✓ Set your QuoVadis Admin Email
- ✓ Set your QuoVadis Requestor Email
- ✓ Upload your QuoVadis API Signing Certificate and Key (p12 file)
- ✓ Upload your QuoVadis API Signing Certificate password

Step 4d: When using Microsoft AD Certificate Server:

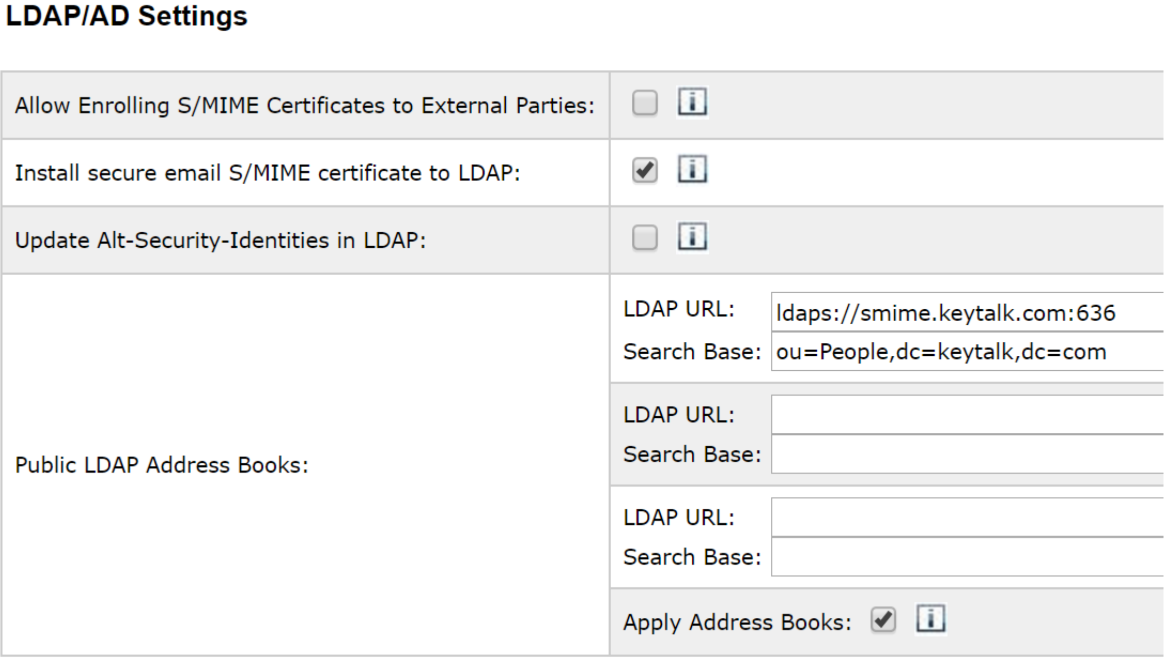


Signer:	Microsoft Enterprise CA ▼
NDES server: *	ndes.mycompany.com
NDES Challenge Password: *	••••••••••••••••••••••••••••••••
Key Size (bits):	4096 ▼
Subject CN:	<will be filled with the value of user CN or, wher
Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Subject Alternative Names:	email:copy

- ✓ Set your NDES server with a static challenge.
  If you haven’t set NDES, follow https://www.youtube.com/watch?v=JOwoMJmgi2g

- ✓ Set Subject Alternative Names to: email:copy

Step 5: Configure writing issued S/MIME certificate to target LDAP Address Book:

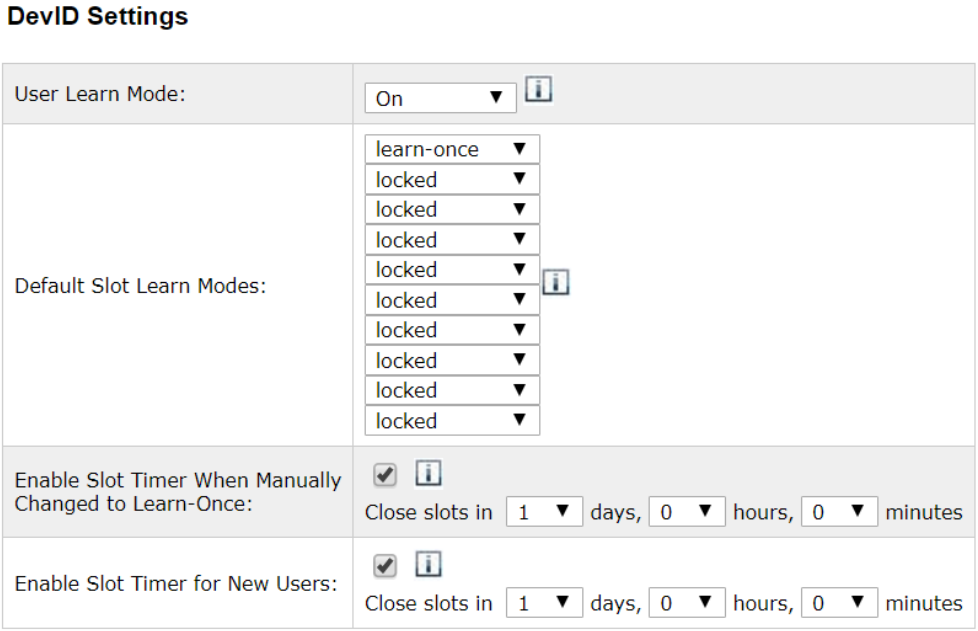


LDAP/AD Settings

Allow Enrolling S/MIME Certificates to External Parties:	☐
Install secure email S/MIME certificate to LDAP:	☑
Update Alt-Security-Identities in LDAP:	☐
Public LDAP Address Books:	LDAP URL: ldaps://smime.keytalk.com:636 Search Base: ou=People,dc=keytalk,dc=com LDAP URL: Search Base: LDAP URL: Search Base: Apply Address Books: ☑

- ✓ An issue S/MIME certificate needs to be written to your set (KeyTalk) LDAP address book, by selecting: Install secure email S/MIME certificate to LDAP.
- ✓ Set your KeyTalk public LDAP URL and Search Base, so that the KeyTalk client can automatically configure it as an Outlook Address Book on Windows (requires Apply Address Books to be checked as well).
  The LDAP URL must follow the format: ldap://<url>:port or ldaps://<url>:port

Step 6: Configure automated device learning mode:



DevID Settings

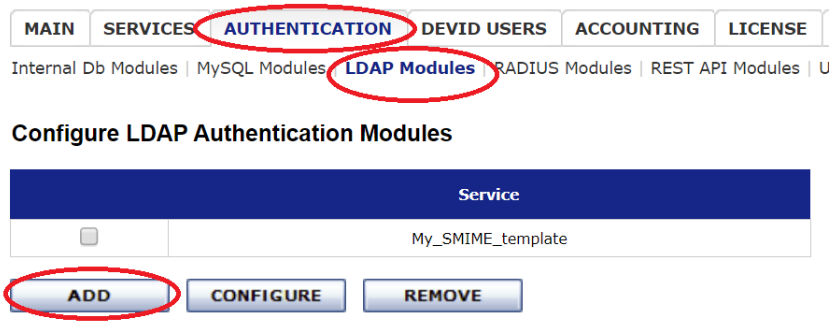
User Learn Mode:	On ▼
Default Slot Learn Modes:	learn-once ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼
Enable Slot Timer When Manually Changed to Learn-Once:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes
Enable Slot Timer for New Users:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes

- ✓ Configure User Learn Mode to: On
- ✓ Configure slot 1 as : learn-once

- ✓ Set the timers.

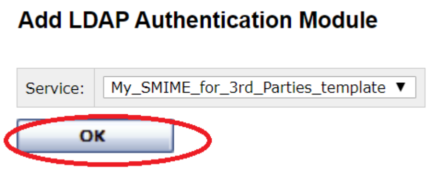
Step 7: Select OK to save the settings

Step 8: Connect your LDAP Address Book to the KeyTalk TEMPLATE certificate template


MAIN
SERVICES
AUTHENTICATION
DEVID USERS
ACCOUNTING
LICENSE
Internal Db Modules | MySQL Modules | LDAP Modules | RADIUS Modules | REST API Modules | U
ADD
CONFIGURE
REMOVE

Configure LDAP Authentication Modules

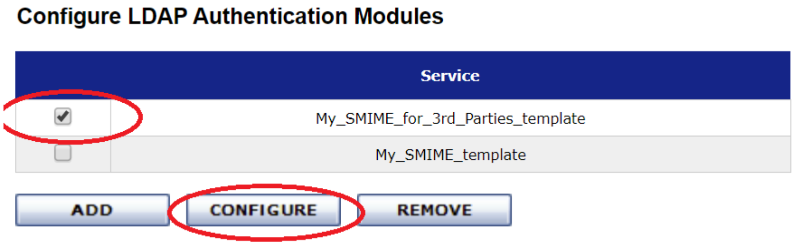
	Service
□	My_SMIME_template



Add LDAP Authentication Module

Service: My_SMIME_for_3rd_Parties_template ▼

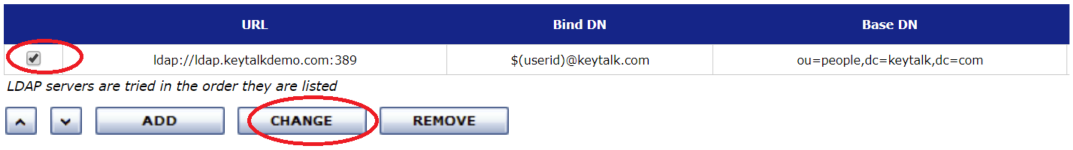
Step 9: Configure the added LDAP/AD connector



Configure LDAP Authentication Modules

	Service
☑	My_SMIME_for_3rd_Parties_template
□	My_SMIME_template

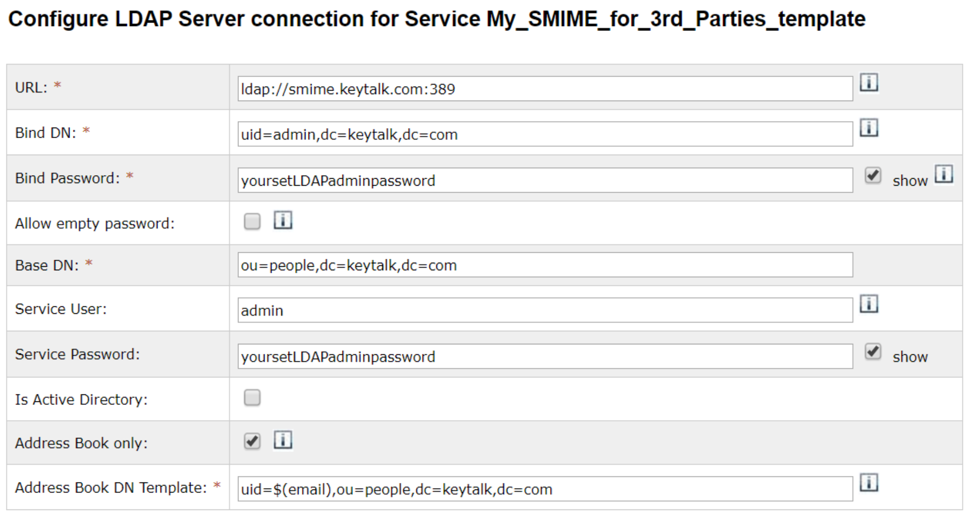
Step 10: Edit the default LDAP



	URL	Bind DN	Base DN
☑	ldap://ldap.keytalkdemo.com:389	$(userid)@keytalk.com	ou=people,dc=keytalk,dc=com

LDAP servers are tried in the order they are listed

Step 11: **Configure the public KeyTalk S/MIME LDAP Address Book connector**

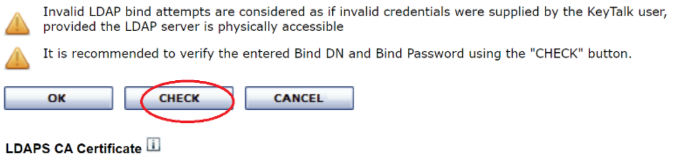


Configure LDAP Server connection for Service My_SMIME_for_3rd_Parties_template

URL: *	ldap://smime.keytalk.com:389
Bind DN: *	uid=admin,dc=keytalk,dc=com
Bind Password: *	yoursetLDAPadminpassword ☑ show
Allow empty password:	☐
Base DN: *	ou=people,dc=keytalk,dc=com
Service User:	admin
Service Password:	yoursetLDAPadminpassword ☑ show
Is Active Directory:	☐
Address Book only:	☑
Address Book DN Template: *	uid=$(email),ou=people,dc=keytalk,dc=com

- ✓ Set your LDAP Addressbook url. Use ldap:// with port :389
- ✓ Set the Bind DN to: uid=admin,dc=keytalk,dc=com
- ✓ Set the Bind Password to the LDAP Admin password you configured on the LDAP
- ✓ Set the Base DN to: ou=People,dc=keytalk,dc=com
- ✓ Set TEMPLATE User to: admin
- ✓ Set TEMPLATE Password to: the LDAP Admin password you configured on the LDAP
- ✓ Check: Address Book only

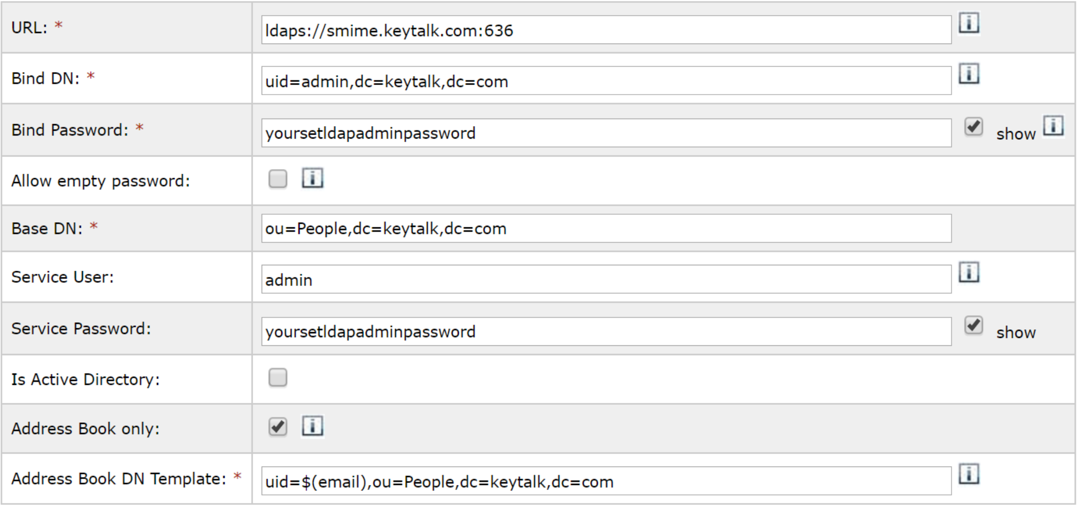
Verify if the connection works using the CHECK button



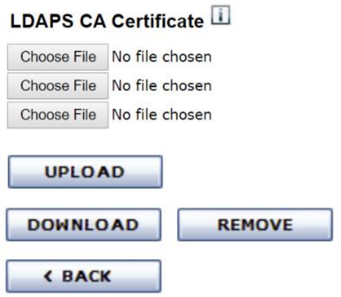
⚠ Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

⚠ It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

Step 12: Configure LDAPS for the Address Book



URL: *	ldaps://smime.keytalk.com:636
Bind DN: *	uid=admin,dc=keytalk,dc=com
Bind Password: *	yoursetldapadminpassword ☑ show
Allow empty password:	☐
Base DN: *	ou=People,dc=keytalk,dc=com
Service User:	admin
Service Password:	yoursetldapadminpassword ☑ show
Is Active Directory:	☐
Address Book only:	☑
Address Book DN Template: *	uid=$(email),ou=People,dc=keytalk,dc=com



LDAPS CA Certificate

Choose File No file chosen

Choose File No file chosen

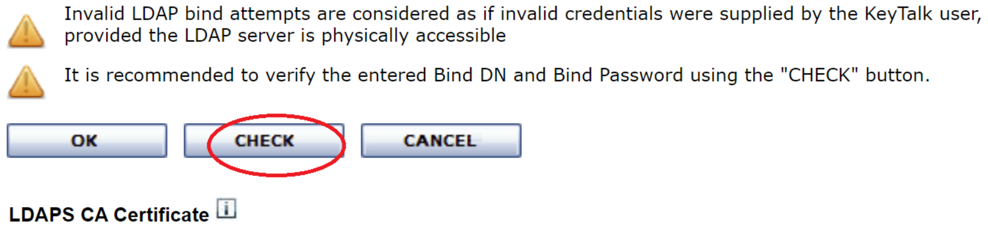
Choose File No file chosen

Use these exact Bind DN, Base DN and Address Book DN template settings as these are fixed values when using the KeyTalk LDAP S/MIME Address Book.

While its not required to configure TLS, it is advised , as it will secure your connection.

- ✓ Change your LDAP Addressbook url to use use ldaps:// with port :636
- ✓ Upload your KeyTalk LDAP SSL certificate issuing CA to create a trust for the KeyTalk server

Now verify if the LDAPS connection works properly using the CHECK button.

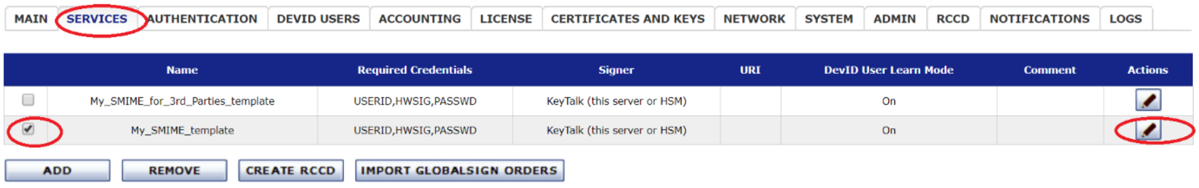


Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

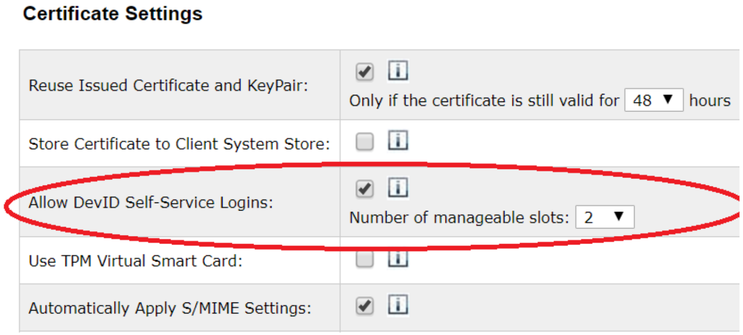
It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

LDAPS CA Certificate

Step 13a: Configure the original S/MIME certificate issuance TEMPLATE template (ANNEX C), to allow for use of Self TEMPLATE Portal access



	Name	Required Credentials	Signer	URI	DevID User Learn Mode	Comment	Actions
☐	My_SMIME_for_3rd_Parties_template	USERID,HWSIG,PASSWD	KeyTalk (this server or HSM)		On		✎
☑	My_SMIME_template	USERID,HWSIG,PASSWD	KeyTalk (this server or HSM)		On		✎

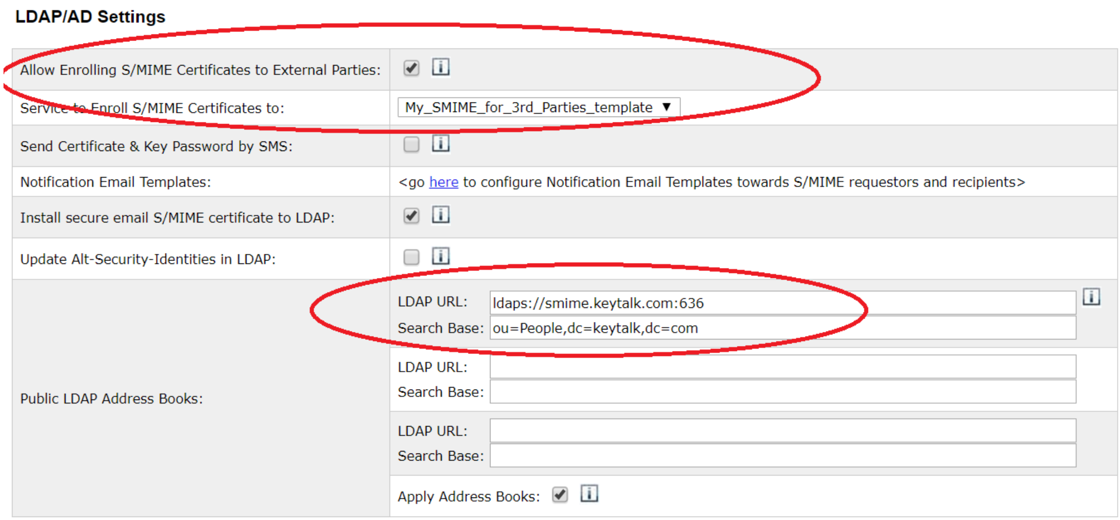


Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ ⓘ Only if the certificate is still valid for 48 ▼ hours
Store Certificate to Client System Store:	☐ ⓘ
Allow DevID Self-Service Logins:	☑ ⓘ Number of manageable slots: 2 ▼
Use TPM Virtual Smart Card:	☐ ⓘ
Automatically Apply S/MIME Settings:	☑ ⓘ

- ✓ Check “Allow DevID Self-TEMPLATE Logins
- ✓ Optionally set the amount of devices a user may manage from the Self TEMPLATE Portal

Step 13b: Enable the ability to request third party certificates using the Self-TEMPLATE Portal

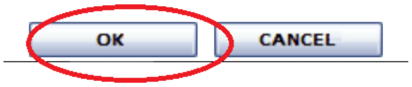


LDAP/AD Settings

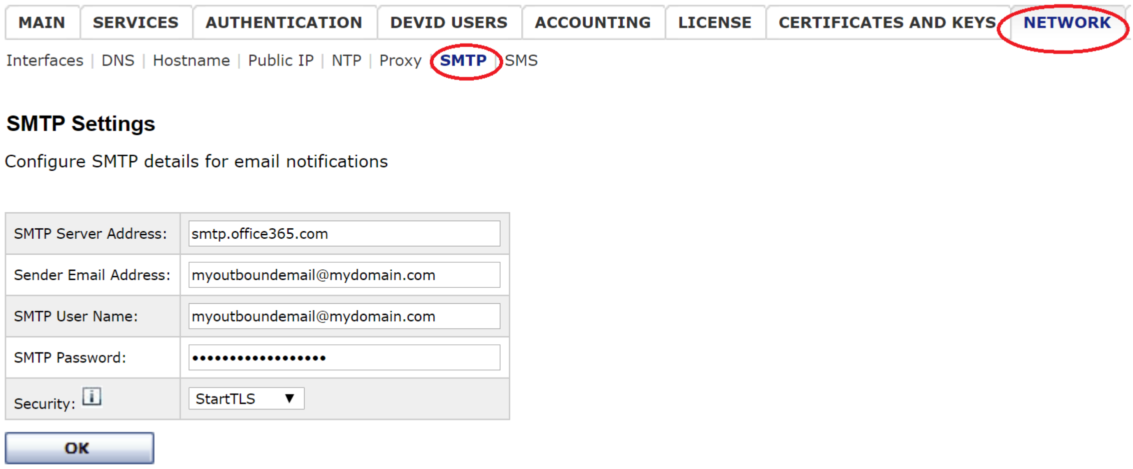
Allow Enrolling S/MIME Certificates to External Parties:	☑
Service to Enroll S/MIME Certificates to:	My_SMIME_for_3rd_Parties_template ▼
Send Certificate & Key Password by SMS:	□
Notification Email Templates:	<go here to configure Notification Email Templates towards S/MIME requestors and recipients>
Install secure email S/MIME certificate to LDAP:	☑
Update Alt-Security-Identities in LDAP:	□
Public LDAP Address Books:	LDAP URL: ldaps://smime.keytalk.com:636 Search Base: ou=People,dc=keytalk,dc=com LDAP URL: Search Base: LDAP URL: Search Base: Apply Address Books: ☑

- ✓ Check “Allow Enrolling S/MIME Certificates to External Parties
- ✓ Select the third party S/MIME issuance TEMPLATE template
- ✓ Set the configured LDAP Address Book as primary, secondary or tertiary entry
- ✓ Optionally ENFORCE the sending of the certificate private key installation password by SMS. *While less convenient for the third party recipient and for your users (as they are required to know the mobile number of the recipient), sending the installation password through a side channel, instead of in the same email / same email based channel, improves overall security and might be required based on corporate policy/compliance requirements. Currently Twillio is support, contact KeyTalk for support of other SMS gateway solutions.*

Step 13c: Save the changes at the bottom of the page:



Step 14a: Configure outbound email for the KeyTalk server



SMTP Settings

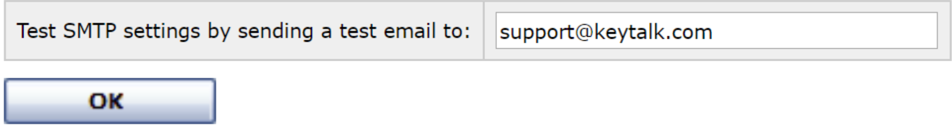
Configure SMTP details for email notifications

SMTP Server Address:	smtp.office365.com
Sender Email Address:	myoutboundemail@mydomain.com
SMTP User Name:	myoutboundemail@mydomain.com
SMTP Password:	••••••••••••••••••
Security:	StartTLS ▼

- ✓ Set the SMTP server address
- ✓ Set the Sender Email address
- ✓ Set the SMTP User Name (when applicable)
- ✓ Set the SMTP Password (when applicable)
- ✓ Set the SMTP security connection

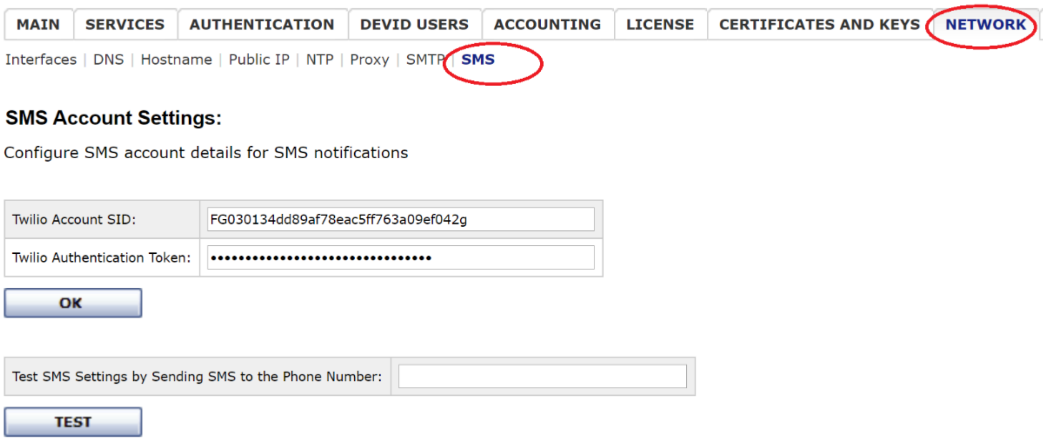
Save by selecting OK

Step 14b: Verify the outbound email to work properly



Test SMTP settings by sending a test email to: support@keytalk.com

Step 14c: Optionally configure the SMS gateway (see step 13b)



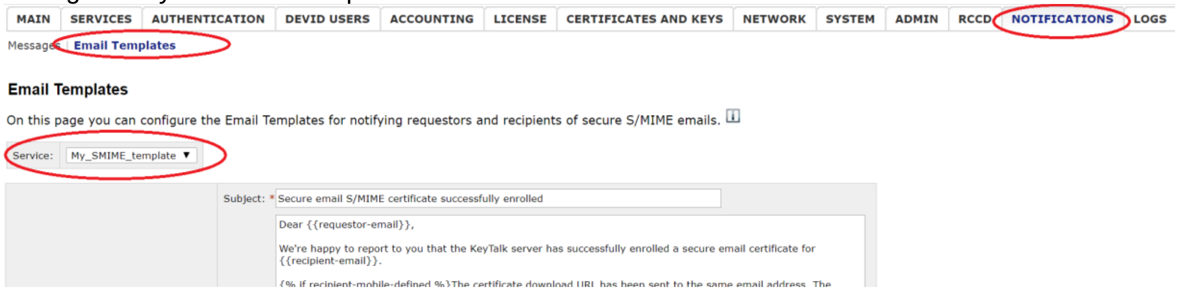
SMS Account Settings:

Configure SMS account details for SMS notifications

Twilio Account SID:	FG030134dd89af78eac5ff763a09ef042g
Twilio Authentication Token:	••••••••••••••••••••••••••••••

Test SMS Settings by Sending SMS to the Phone Number:

Step 15: Configure KeyTalk email templates.



Email Templates

On this page you can configure the Email Templates for notifying requestors and recipients of secure S/MIME emails.

Service: My_SMIME_template ▼

Subject: * Secure email S/MIME certificate successfully enrolled

Dear {{requestor-email}},

We're happy to report to you that the KeyTalk server has successfully enrolled a secure email certificate for {{recipient-email}}.

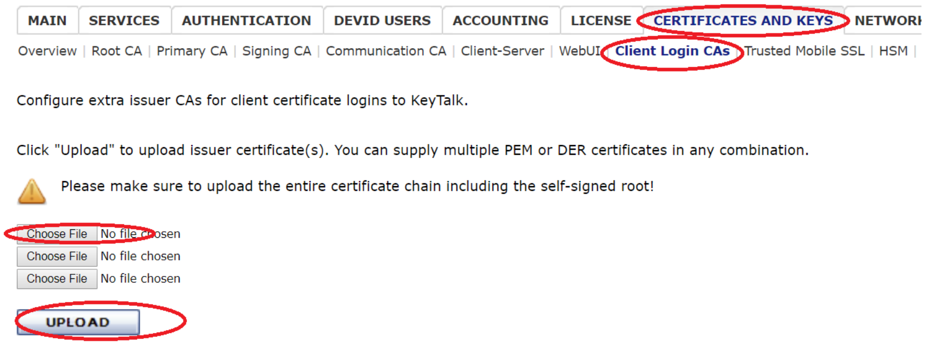
{% if recipient-mobile-defined %}The certificate download URL has been sent to the same email address. The

Based on the configured step 14a outbound email, the KeyTalk server will send email messages to your corporate users who request S/MIME certificates for third parties, and to those for whom an S/MIME certificate has been requested.

KeyTalk comes pre-loaded with fully working factory default email templates for every use-case covered in the request process. These templates can be set for each primary S/MIME TEMPLATE certificate template. So you do not set the templates for the slaved third party S/SMIME TEMPLATE template.

You can change the Subject and the Message Body accordingly in this section. So you are not bound to English, or the actual content.

Step 16: Configure KeyTalk to trust your chosen S/MIME CA-source.



Configure extra issuer CAs for client certificate logins to KeyTalk.

Click "Upload" to upload issuer certificate(s). You can supply multiple PEM or DER certificates in any combination.

⚠ Please make sure to upload the entire certificate chain including the self-signed root!

Choose File No file chosen
Choose File No file chosen
Choose File No file chosen

UPLOAD

By default the KeyTalk server will only trust its own locally configured private CA.
Therefor if you're issuing S/MIME and authentication certificates to your corporate users using the KeyTalk private CA, there is no need to configure this part and you can continue to step 16a.

When you're issuing S/MIME and authentication certificates to your corporate users using a non-KeyTalk-private-CA, such as Microsoft ADCS, or GlobalSign/TrustZone/QuoVadis etc, then you will need to configure KeyTalk to trust the used CA(s).

Simply upload the issuing CA in DER or PEM format for each of your KeyTalk instances. For the public CA's that KeyTalk already integrates with, you can find the most recent issuing CA file in your KeyTalk server download.

Step 16a: Enable Self-TEMPLATE Portal and KeyTalk Management certificate based strong authentication

The Self-TEMPLATE portal of KeyTalk enables your corporate users to request certificates for third party recipients using https://<yourkeytalkerverfqdn>:3000 over TLS 1.3

Because corporate users will get access as authorized regular users, as well as your KeyTalk Admins with leveraged authorization, currently the KeyTalk system requires certificate based strong authentication (ie mutual SSL authentication over TLS 1.3)

This requires an “admin” user (ie the local KeyTalk sys admin) to exist as a user with a unique client authentication certificate with a matching CN=admin, and a value in the O and the OU field, as issued under a trusted issuing CA.

You could opt to NOT have a local sys admin defined, and only use Cluster Admins, which will work with any defined CN value in a client authentication certificate.

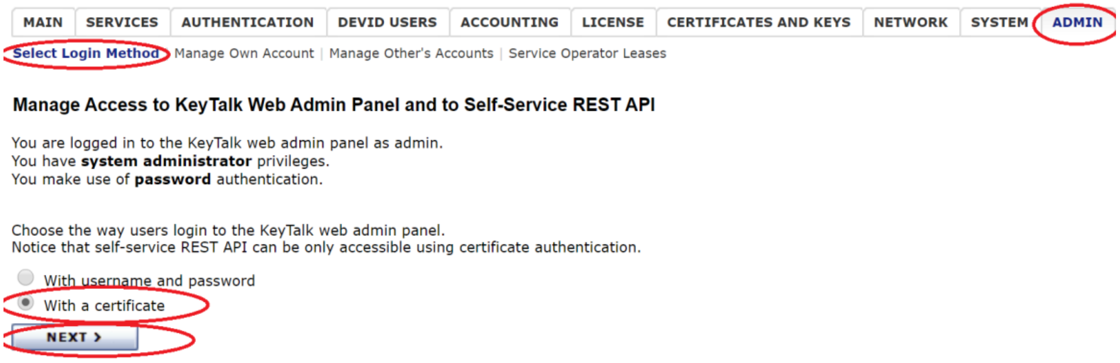
There are 4 methods of obtaining such a trusted admin client authentication certificate:

1) Issue it under KeyTalk private CA using your AD authentication connector, based on an AD defined “Admin” user, (see chapter 13.2), OR
2) Issue it under KeyTalk private CA using your AD authentication connector, based on any existing user, but then using CN certificate field mapping based on an AD attribute that will contain the name “admin”, (see chapter 13.2), OR
3) Issue it under KeyTalk private CA using KeyTalk’s internal DB connector, based on an internalDB defined “admin” user, (see chapter 13.1), OR
4) Issue it under another CA, whereby you set your KeyTalk server to trust certificates issued under this particular CA (see chapter

When using options 1-3, KeyTalk recommends using short-lived certificates for your KeyTalk (delegated) admins. ie a validity period of 1-12 hours, possibly extending up to 1 week validity.
As the additional issuance of KeyTalk (delegated) admin logins will likely affect your license count, kindly contact sales@keytalk.com, or your KeyTalk vendor, to see how your existing license can be compensated for this additional certificate issuance.

Before continuing to step 16b, ensure you are able to fetch a client authentication certificate using the KeyTalk client/app, or have a client authentication certificate available whereby its issuing CA is trusted by your KeyTalk environment.

Step 16b: Set enforced certificate based authentication to your KeyTalk environment.

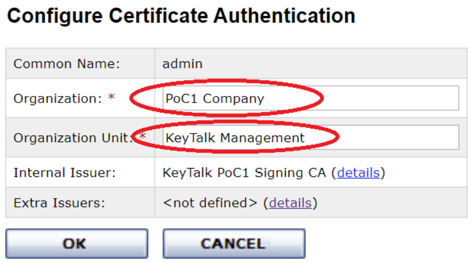


Manage Access to KeyTalk Web Admin Panel and to Self-Service REST API

You are logged in to the KeyTalk web admin panel as admin.
You have **system administrator** privileges.
You make use of **password** authentication.

Choose the way users login to the KeyTalk web admin panel.
Notice that self-service REST API can be only accessible using certificate authentication.

○ With username and password
● With a certificate

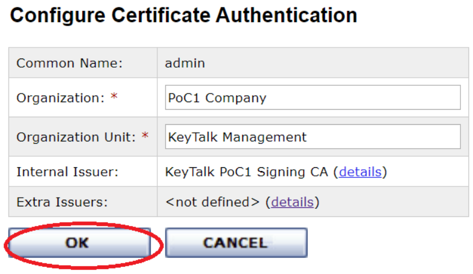


Configure Certificate Authentication

Common Name:	admin
Organization: *	PoC1 Company
Organization Unit: *	KeyTalk Management
Internal Issuer:	KeyTalk PoC1 Signing CA (details)
Extra Issuers:	<not defined> (details)

- ✓ Set the Organization name as found in the client authentication certificate’s O field for the Local Sys Admin access
- ✓ Set the Organizational Unit name as found in the client authentication certificate’s OU field for the Local Sys Admin access

Step 16c: Activate enforced certificate based authentication for each KeyTalk server instance:



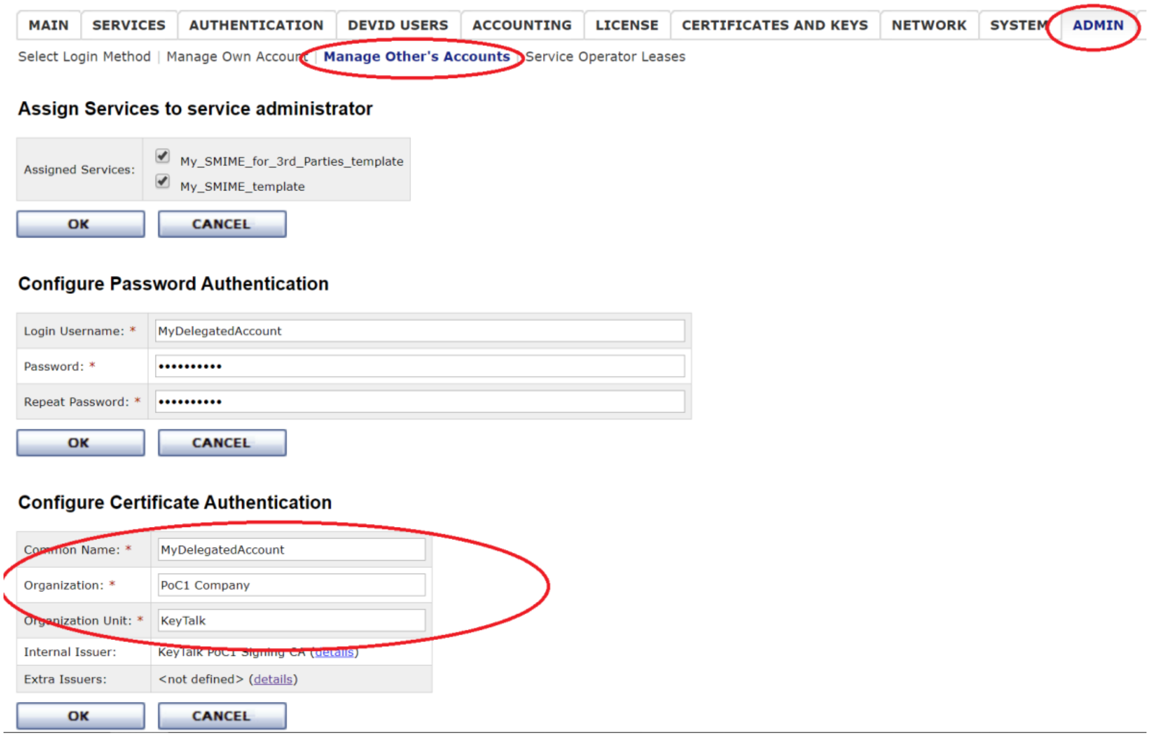
Configure Certificate Authentication

Common Name:	admin
Organization: *	PoC1 Company
Organization Unit: *	KeyTalk Management
Internal Issuer:	KeyTalk PoC1 Signing CA (details)
Extra Issuers:	<not defined> (details)

This will trigger a Daemon reboot of the KeyTalk server, after which an appropriate client certificate will be requested to login with an appropriate configured authorization.

If you're locked yourself out of the system, you can undo the enforced certificate authentication by accessing SSH or the CLI of the KeyTalk server and using:
/usr/local/bin/keytalk/www/reset-admin-passwd
This will set the default sys admin login to : admin/change!

Step 16d: Configure KeyTalk delegated Admin accounts for certificate authentication



Assign Services to service administrator

Assigned Services:	☑ My_SMIME_for_3rd_Parties_template ☑ My_SMIME_template

Configure Password Authentication

Login Username: *	MyDelegatedAccount
Password: *	••••••••••
Repeat Password: *	••••••••••

Configure Certificate Authentication

Common Name: *	MyDelegatedAccount
Organization: *	PoC1 Company
Organization Unit: *	KeyTalk
Internal Issuer:	KeyTalk PoC1 Signing CA (details)
Extra Issuers:	<not defined> (details)

If other users/roles have been defined in KeyTalk, configure the appropriate certificate based authentication field matching settings.

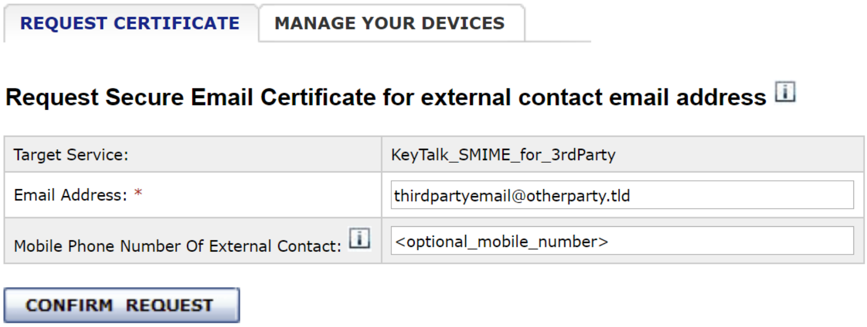
Contrary to the local sys admin, all other account settings are synched with other KeyTalk server instances in case you are running a KeyTalk High Availability cluster.

Important note: KeyTalk will always assign the highest authorization role to a matching certificate.
So if someone's certificate details match those of a "regular" user, and those of an (delegated) admin role, the admin interface will prevail over the user Self-TEMPLATE portal interface.

Step 17a: **Test the Self-TEMPLATE portal, third party S/MIME certificate request**
Ensure you have issued a corporate S/MIME certificate using the KeyTalk client based on the ANNEX C KeyTalk TEMPLATE template.

Step 17b: **Visit https://<yourkeytalkserverurl>:3000 and select your S/MIME client authentication certificate to login.**

Step 17c: **Request third party S/MIME certificate through the portal.**



Request Secure Email Certificate for external contact email address ⓘ

Target Service:	KeyTalk_SMIME_for_3rdParty
Email Address: *	thirdpartyemail@otherparty.tld
Mobile Phone Number Of External Contact: ⓘ	<optional_mobile_number>

- ✓ Enter the third party email address you wish to request an S/MIME certificate for. It's highly recommended to use a third party email address that you control, so you can verify both ends of the process, ie as as requestor and as a target recipient.
- ✓ When SMS based password sending is enforced, also enter the mobile number
- ✓ Select CONFIRM REQUEST

When all is configured properly, both on your KeyTalk server side, and on your CA-provider side, a confirmation should appear.

If something goes wrong:

- Close the browser (not the tab, but the full browser in order to clear the SSL cache)
- Open the browser and visit your KeyTalk Admin interface, and authenticate using a KeyTalk trusted (delegated) admin certificate
- View the WebUI log, and CAD log to find the cause of the problem

Step 17d: Depending on the selected third party S/MIME CA provider, different emails and processes are triggered.
Keep a close eye on your email as both corporate requestor and the email of the third party recipient.
Emails sent by the KeyTalk system can be changed (see step 15).
Emails sent by the non-KeyTalk CA, can only be changed by those controlling this CA environment.

Step 17e: When the third party S/MIME certificate has been delivered to the recipient, validate if the certificate got registered in your configured LDAP Address Book.
Either login to the LDAP using an LDAP manager, or in case of the KeyTalk S/MIME LDAPS Address book you can also visit https://<myldapurl> and do a lookup based on the email address.
If it didn't register, verify in the KeyTalk AuthD and RDD log why it failed.

Step 17f: Validate if your Windows Outlook/MacMail has been configured by the KeyTalk client to make use of the LDAP addressbook

Step 17g: Use your Windows Outlook, or MacMail, to write an encrypted and signed email to the target third party email address, and send it.

When all goes well, the email gets send properly.

If things don't go well, it mostly happens because the email client is unable to read the LDAP Address Book, and therefor is unable to read the target email's public S/MIME details. This mostly happens due to LDAP as a protocol not being allows Firewall-rule wise, or because of a DNS issue client side, or because of a misconfiguration of the LDAP Address Book in Windows Outlook, or on Apple Mac as an LDAP profile. (Apple Mac is pretty strict in TLS validation for example)

Step 17h: Validate as the third party recipient that you can decrypt the received message (step 17g) using the received and installed S/MIME certificate and private key.

Step 17i: Validate as the third party recipient, that you can reply to the received encrypted email in an encrypted and digitally signed manner.

Notes:

- ➢ While KeyTalk cannot provide support on software of other vendors, several KeyTalk customers have indicated the following mail clients work just fine with S/MIME and most also work with integrated LDAP Address Books, albeit these mail clients may require some manual configurations:
  - ✓ Windows: Outlook
  - ✓ Windows: Thunderbird
  - ✓ Linux: Thunderbird
  - ✓ Linux: Evolution
  - ✓ MacOSX: MacMail
  - ✓ MacOSX: Outlook for Mac
  - ✓ MacOSX: Thunderbird
  - ✓ iOS: Mail
  - ✓ iOS: MobileIron Email+ (requires MobileIron)
  - ✓ iOS: Outlook (requires Intune!)
  - ✓ Android: Samsung mail
  - ✓ Android: Ciphermail
  - ✓ Android: Nine - Email

ANNEX E: Sample end-user machine certificate use-case configuration

Machine certificates for end-point devices are often used for 802.1x EAP/TLS network authentication.

KeyTalk supports several use-cases to issue/renew/(de)install end-point machine certificates.

Most common scenario's include:

1) Machine Certificate issuance based on a user's authentication credentials (such as Kerberos) against an AD, after which, based on the user's machine identification, the machine certificate is requested/generated, pushed and installed on the end-points system or personal certificate store (optionally requiring the CSR to be generated client side with or without TPM)

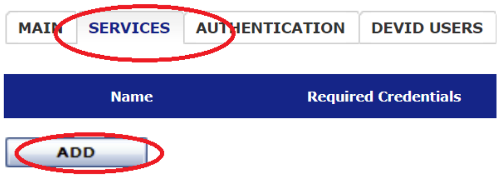
2) Machine Certificate issuance based on an end-points authentication credentials (such as username/password) against an internal Db

3) Machine Certificate issuance based on an end-points sole request against an internal Db

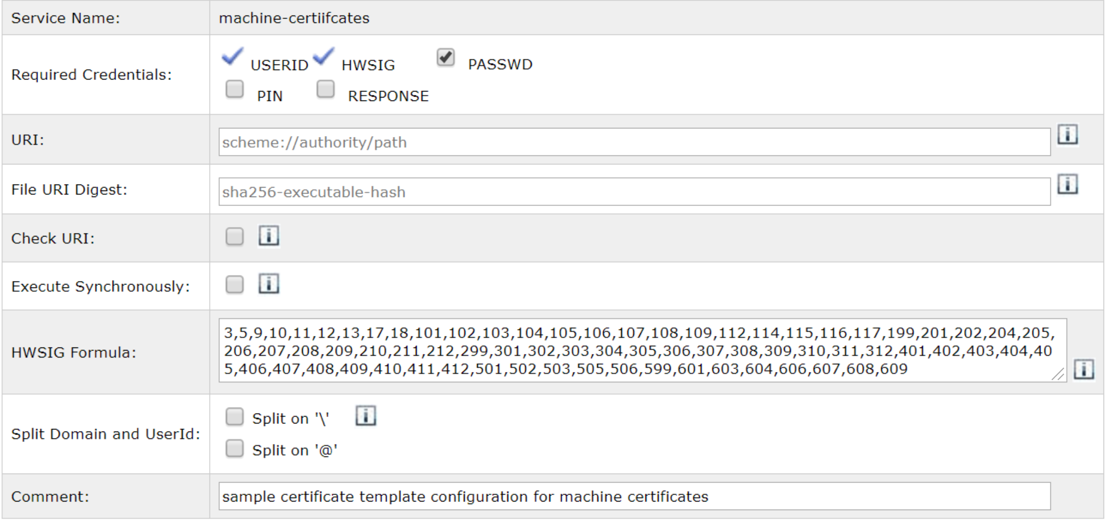
This End-user Kerberos/AD username-password based use-case (scenario 1) example assumes:

- ✓ the KeyTalk server has been generically configured already
- ✓ an AD is configured

Step 1: Configure a KeyTalk certificate template under TEMPLATES

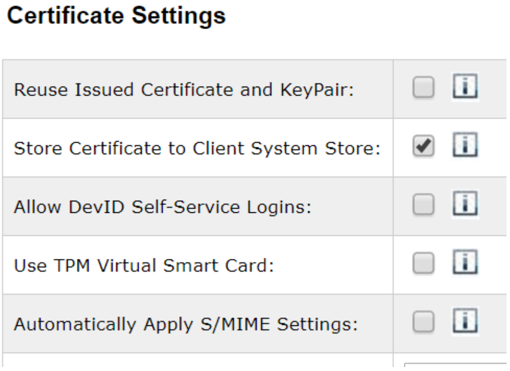


Step 2: Configure the TEMPLATE general settings, and ensure you select password. Optionally make appropriate changes to the HWSIG Formula hardware/software characteristics.



Service Name:	machine-certiifcates
Required Credentials:	✓ USERID ✓ HWSIG ☑ PASSWD □ PIN □ RESPONSE
URI:	scheme://authority/path
File URI Digest:	sha256-executable-hash
Check URI:	□
Execute Synchronously:	□
HWSIG Formula:	3,5,9,10,11,12,13,17,18,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,599,601,603,604,606,607,608,609
Split Domain and UserId:	□ Split on '\' □ Split on '@'
Comment:	sample certificate template configuration for machine certificates

Step 3: Configure certificate settings

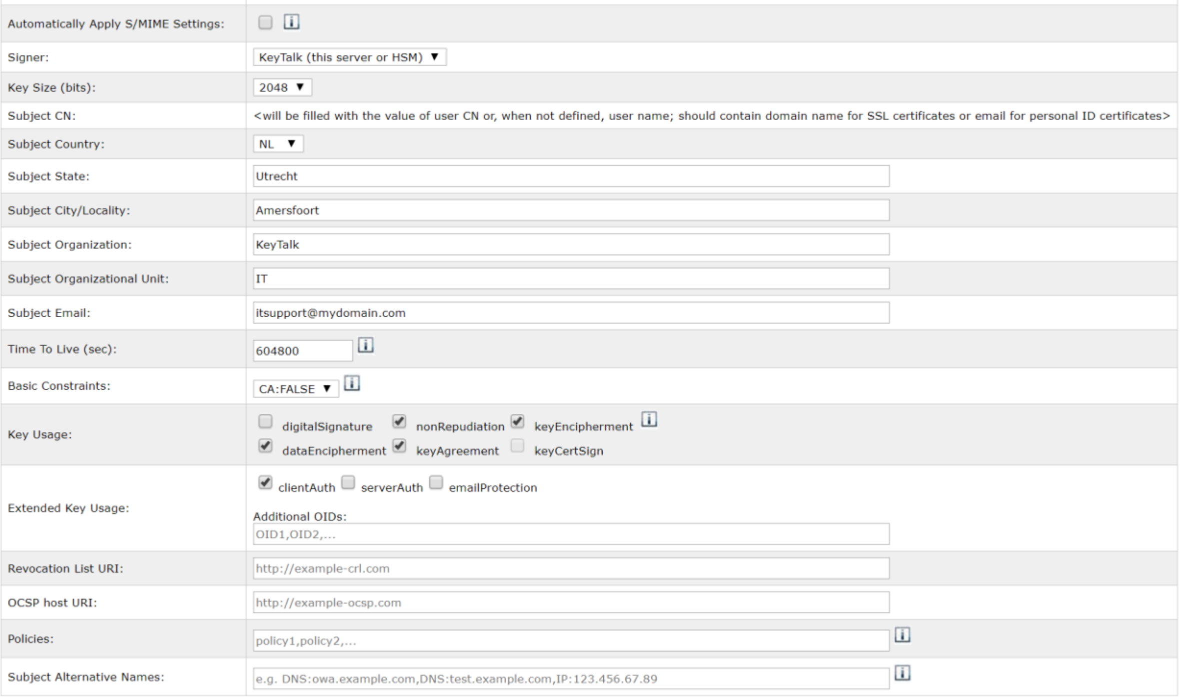


Certificate Settings

Reuse Issued Certificate and KeyPair:	☐
Store Certificate to Client System Store:	☑
Allow DevID Self-Service Logins:	☐
Use TPM Virtual Smart Card:	☐
Automatically Apply S/MIME Settings:	☐

- ✓ Set Store Certificate to Client System Store

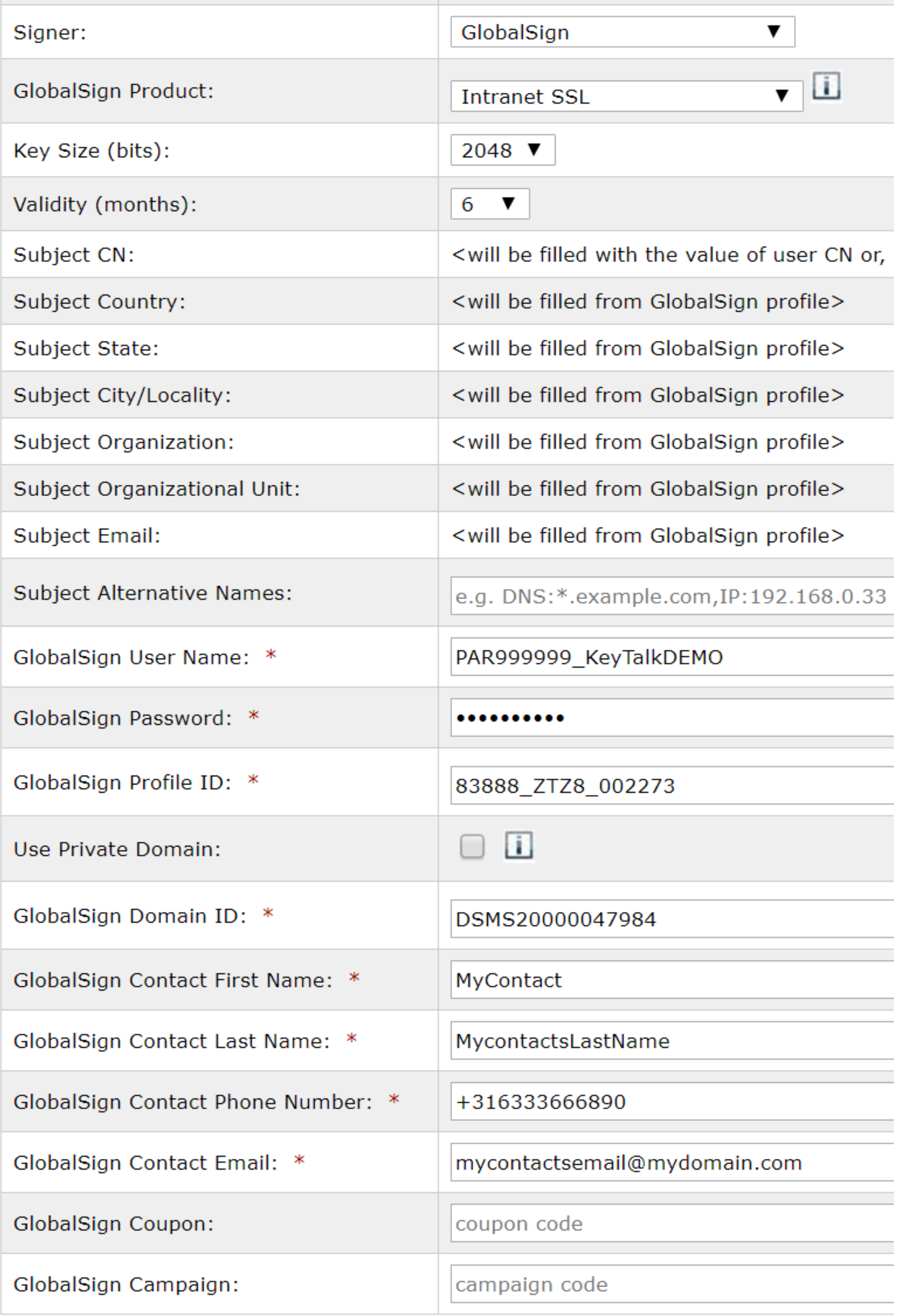
Step 4a: When using KeyTalk Private CA:



Automatically Apply S/MIME Settings:	☐
Signer:	KeyTalk (this server or HSM) ▼
Key Size (bits):	2048 ▼
Subject CN:	<will be filled with the value of user CN or, when not defined, user name; should contain domain name for SSL certificates or email for personal ID certificates>
Subject Country:	NL ▼
Subject State:	Utrecht
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk
Subject Organizational Unit:	IT
Subject Email:	itsupport@mydomain.com
Time To Live (sec):	604800
Basic Constraints:	CA:FALSE ▼
Key Usage:	☐ digitalSignature ☑ nonRepudiation ☑ keyEncipherment ☑ dataEncipherment ☑ keyAgreement ☐ keyCertSign
Extended Key Usage:	☑ clientAuth ☐ serverAuth ☐ emailProtection Additional OIDs: OID1,OID2,...
Revocation List URI:	http://example-crl.com
OCSP host URI:	http://example-ocsp.com
Policies:	policy1,policy2,...
Subject Alternative Names:	e.g. DNS:owa.example.com,DNS:test.example.com,IP:123.456.67.89

- ✓ Optionally set the subject information to match your own company. Arguably he third party will not represent your company, so often its left empty
- ✓ Set Time to Live to (sec): 604800 (1 week), or use any other value you want.
- ✓ Set Basic Constaints to CA:FALSE
- ✓ Set Extended Key Usage to: clientAuth

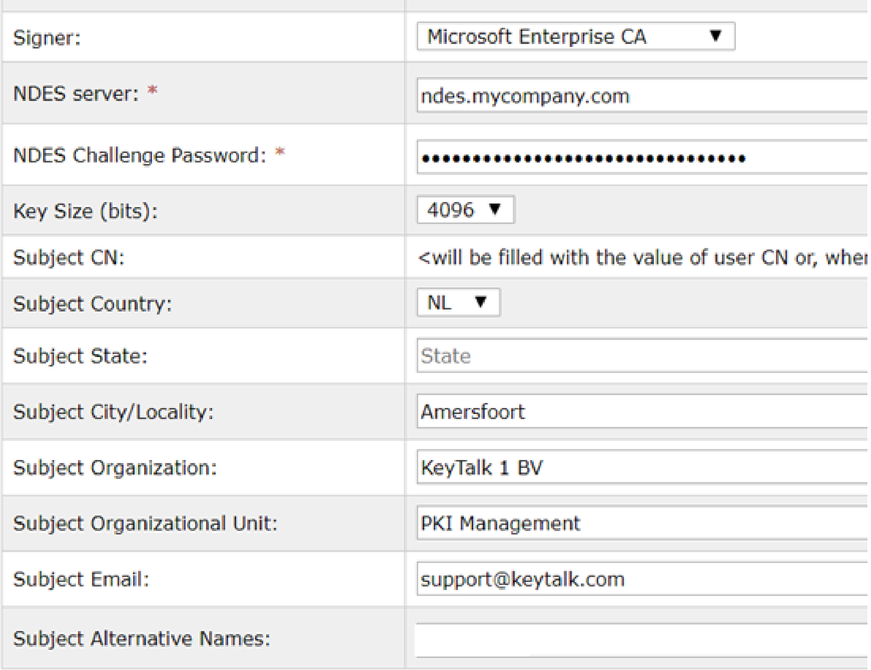
Step 4b: When using GlobalSign/TRUSTZONE:



Signer:	GlobalSign ▼
GlobalSign Product:	Intranet SSL ▼
Key Size (bits):	2048 ▼
Validity (months):	6 ▼
Subject CN:	<will be filled with the value of user CN or,
Subject Country:	<will be filled from GlobalSign profile>
Subject State:	<will be filled from GlobalSign profile>
Subject City/Locality:	<will be filled from GlobalSign profile>
Subject Organization:	<will be filled from GlobalSign profile>
Subject Organizational Unit:	<will be filled from GlobalSign profile>
Subject Email:	<will be filled from GlobalSign profile>
Subject Alternative Names:	e.g. DNS:*.example.com,IP:192.168.0.33
GlobalSign User Name: *	PAR999999_KeyTalkDEMO
GlobalSign Password: *	•••••••••••
GlobalSign Profile ID: *	83888_ZTZ8_002273
Use Private Domain:	☐
GlobalSign Domain ID: *	DSMS20000047984
GlobalSign Contact First Name: *	MyContact
GlobalSign Contact Last Name: *	MycontactsLastName
GlobalSign Contact Phone Number: *	+316333666890
GlobalSign Contact Email: *	mycontactsemail@mydomain.com
GlobalSign Coupon:	coupon code
GlobalSign Campaign:	campaign code

- ✓ Select product: PersonalSign 1, or PersonalSign Express
- ✓ Select the validity period that matches your GlobalSign/TRUSTZONE license
- ✓ Set your GlobalSign/TRUSTZONE User Name
- ✓ Set your GlobalSign/TRUSTZONE Password
- ✓ Set your GlobalSign/TRUSTZONE Profile ID as found in the portal under ENTERPRISE PKI

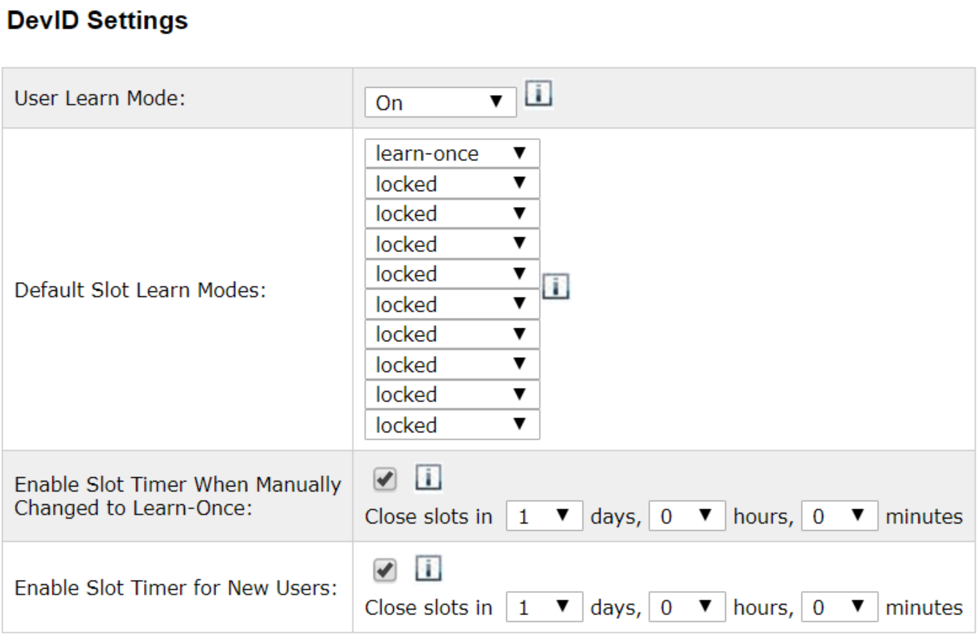
Step 4c: When using Microsoft AD Certificate Server:



Signer:	Microsoft Enterprise CA ▼
NDES server: *	ndes.mycompany.com
NDES Challenge Password: *	••••••••••••••••••••••••••••••••
Key Size (bits):	4096 ▼
Subject CN:	<will be filled with the value of user CN or, whe
Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Subject Alternative Names:	

✓ Set your NDES server with a static challenge.
If you haven't set NDES, follow https://www.youtube.com/watch?v=JOwoMJmgi2g

Step 5: Configure automated device learning mode:



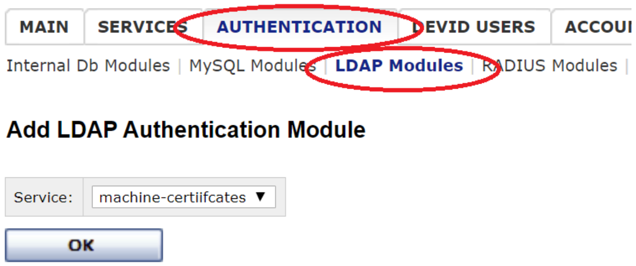
DevID Settings

User Learn Mode:	On ▼
Default Slot Learn Modes:	learn-once ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼
Enable Slot Timer When Manually Changed to Learn-Once:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes
Enable Slot Timer for New Users:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes

✓ Configure User Learn Mode to: On
✓ Configure slot 1 as : learn-once
✓ Set the timers

Step 7: Select OK to save the settings

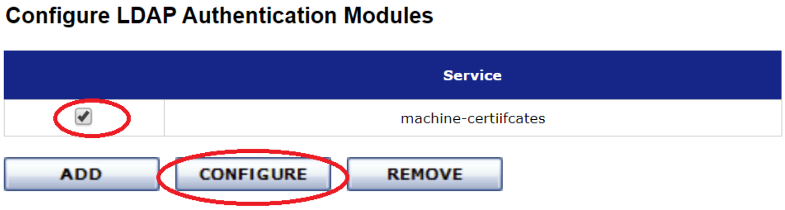
Step 8: Connect your (Azure) Active Directory to the KeyTalk TEMPLATE certificate template


MAIN
SERVICES
AUTHENTICATION
DEVID USERS
ACCOU
Internal Db Modules | MySQL Modules | LDAP Modules | RADIUS Modules |
OK

Add LDAP Authentication Module

Service: machine-certiifcates ▼

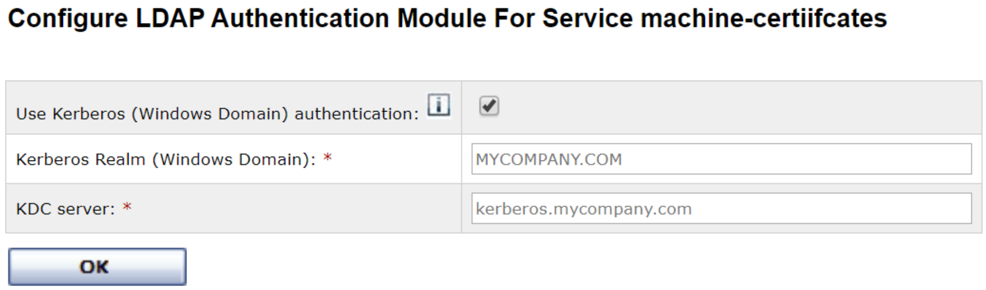
Step 9: Configure the added LDAP/AD connector



Configure LDAP Authentication Modules

☑	Service
☑	machine-certiifcates

Step 10: **Configure Kerberos**



Configure LDAP Authentication Module For Service machine-certiifcates

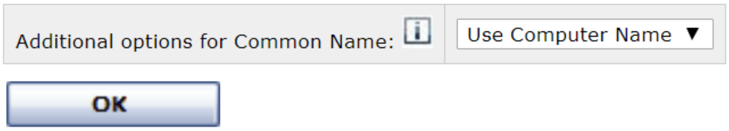
Use Kerberos (Windows Domain) authentication:	☑
Kerberos Realm (Windows Domain): *	MYCOMPANY.COM
KDC server: *	kerberos.mycompany.com

- ✓ **Set your Windows domain**
- ✓ **Set your KDC server**

Save by selecting OK

When Kerberos is not configured, then the KeyTalk client/app will always ask for a username/password given these sample configuration settings

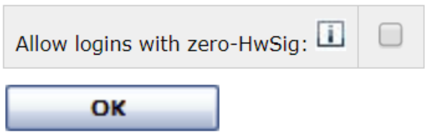
Step 11: Configure Common Name options



Additional options for Common Name: Use Computer Name ▼

- ✓ Set to Use Computer Name

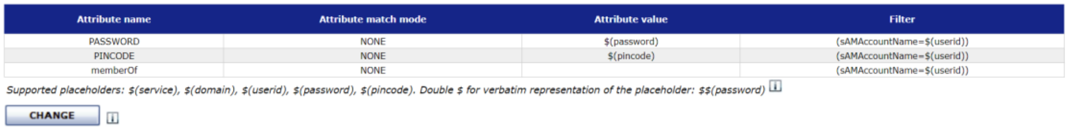
Step 12: Configure Hardware Signature requirement



Allow logins with zero-HwSig: ☐

- ✓ Leave unchecked (this enforces a non-zero value from a KeyTalk client/app when calculating a HardwareSignature)

Step 13: Configure AD/LDAP parameter Match Settings

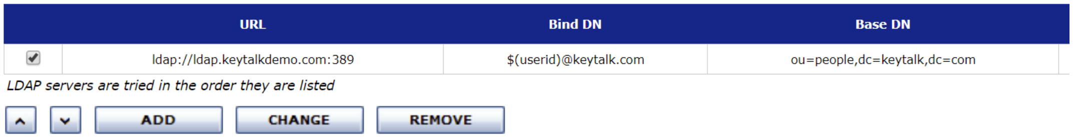


Attribute name	Attribute match mode	Attribute value	Filter
PASSWORD	NONE	$(password)	(sAMAccountName=$(userid))
PINCODE	NONE	$(pincode)	(sAMAccountName=$(userid))
memberOf	NONE		(sAMAccountName=$(userid))

Supported placeholders: $(service), $(domain), $(userid), $(password), $(pincode). Double $ for verbatim representation of the placeholder: $$(password)

Optionally configure this part to enforce a user being part of a specific Security Group (CHANGE -> memberof)

Step 14: Configure AD/LDAP connection settings

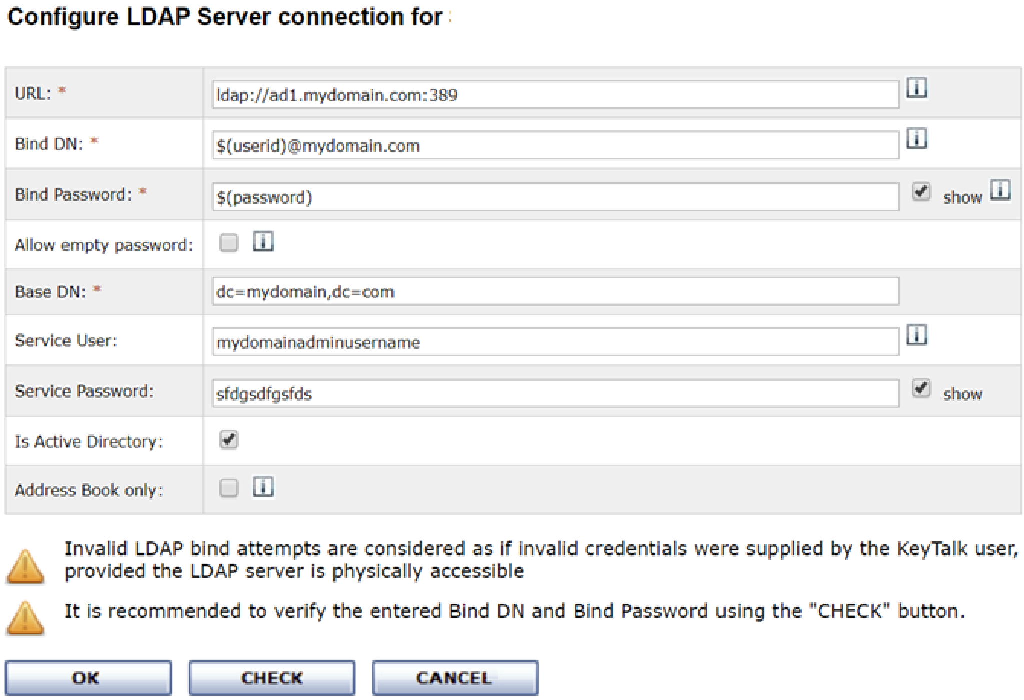


	URL	Bind DN	Base DN
☑	ldap://ldap.keytalkdemo.com:389	$(userid)@keytalk.com	ou=people,dc=keytalk,dc=com

LDAP servers are tried in the order they are listed

Select the default LDAP and select CHANGE

Step 15: Configure AD/LDAP



Configure LDAP Server connection for

URL: *	ldap://ad1.mydomain.com:389
Bind DN: *	$(userid)@mydomain.com
Bind Password: *	$(password) ☑ show
Allow empty password:	☐
Base DN: *	dc=mydomain,dc=com
Service User:	mydomainadminusername
Service Password:	sfdgsdfgsfds ☑ show
Is Active Directory:	☑
Address Book only:	☐

Invalid LDAP bind attempts are considered as if invalid credentials were supplied by the KeyTalk user, provided the LDAP server is physically accessible

It is recommended to verify the entered Bind DN and Bind Password using the "CHECK" button.

Set your:

- ✓ AD URL using ldap:// and port 389
- ✓ Bind DN , its usually $(userid)@mydomain.com (substitute with your own domain)
- ✓ Bind Password set to $(password)
- ✓ Base DN to dc=mydomain,dc=com (substitute with your own domain and tld)
- ✓ TEMPLATE User: this is an AD domain admin capable of writing certificates to the attribute UserCertificate
- ✓ TEMPLATE Password: the password belonging to your set TEMPLATE User
- ✓ Checkmark Is Active Directory

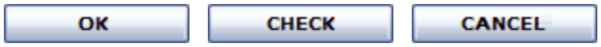
Select OK to save

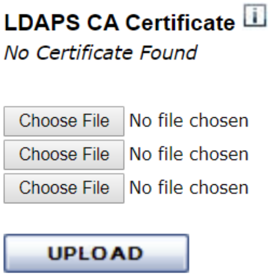
Step 16: Repeat step 14, now select CHECK to see if the connector works properly

BIND SUCCESSFUL

If BIND fails, check the LOGS -> WEBUI LOG

Step 17: Configure secure LDAP/AD connection
Repeat step 14:





LDAPS CA Certificate ⓘ

No Certificate Found

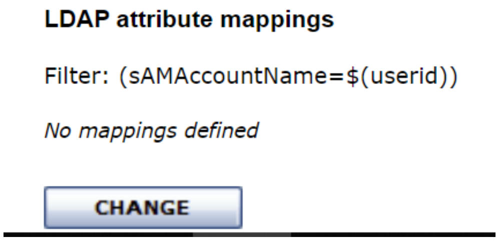
Choose File No file chosen
Choose File No file chosen
Choose File No file chosen

- ✓ Upload in DER or PEM your AD/LDAP LDAPS issuing CA certificate.
  This will result in KeyTalk server trusting your specific CA, enabling TLS over LDAPS

Step 18: Update your set AD/LDAP url to include LDAPS:// and port 636

Step 19: Repeat step 16 to verify if the LDAPS connection is working correctly

Step 20: Repeat step 9, scroll to the bottom of the page:



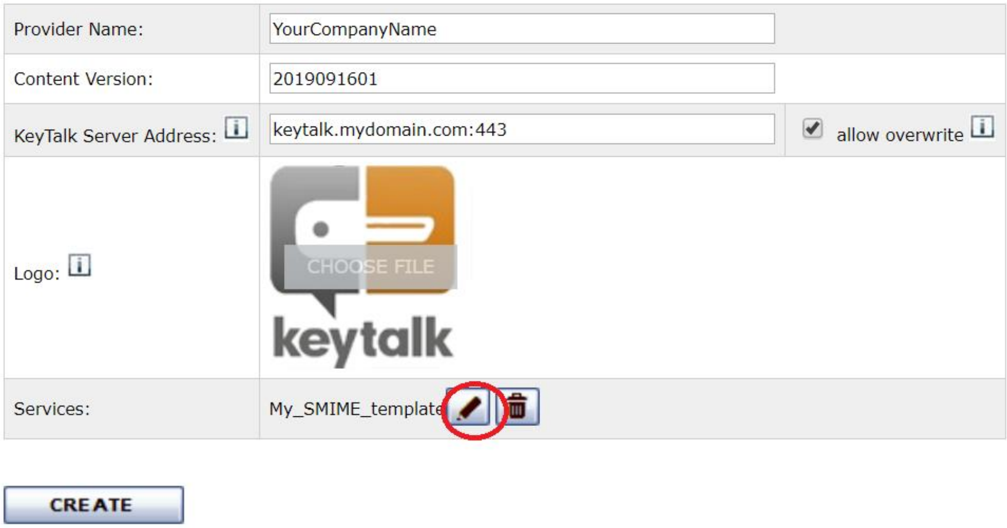
LDAP attribute mappings

Filter: (sAMAccountName=$(userid))

No mappings defined

Change the LDAP attribute mappings

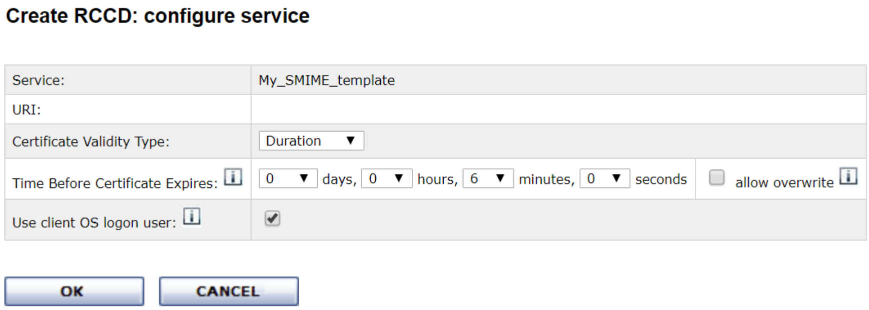
Step 21: Create KeyTalk Real Client Configuration Data file.
This file is used to configure the KeyTalk agent



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address: ⓘ	keytalk.mydomain.com:443	☑ allow overwrite ⓘ
Logo: ⓘ		
Services:	My_SMIME_template	

- ✓ Set KeyTalk Provider Name (this is the name under which multiple KeyTalk configurations are grouped when imported into a single client)
- ✓ Set KeyTalk Server Address. This is either the FQDN or the IP address. Ensure the this address is also present in the KeyTalk “Certificates and Keys” in the SAN of the “Client-Server certificate” ”(generic TLS use for most KeyTalk clients), and when applicable in the KeyTalk “Certificates and Keys” in the SAN of the “Trusted Mobile SSL” (required for Apple and Android devices, and usable for generic KeyTalk client TLS purposes)
- ✓ Set the port to 443 (generic use for KeyTalk client/app), or optionally to port 4443 when a Trusted Mobile SSL certificate and key has been uploaded.
- ✓ Optionally change the Logo to a 110x110 PNG
- ✓ Select the EDIT button next to the selected TEMPLATE

Step 22: Configure KeyTalk Real Client Configuration Data meta data



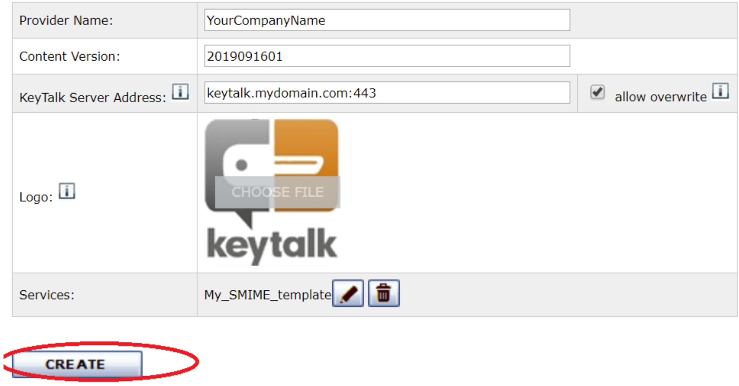
Create RCCD: configure service

Service:	My_SMIME_template	
URI:		
Certificate Validity Type:	Duration ▼	
Time Before Certificate Expires: ⓘ	0 ▼ days, 0 ▼ hours, 6 ▼ minutes, 0 ▼ seconds	☐ allow overwrite ⓘ
Use client OS logon user: ⓘ	☑	

- ✓ Set the duration time, which determines the amount of time before the certificate needs to be replaced before it expires.
- ✓ Set : Use client OS logon user. This ensures that the KeyTalk client/app uses the Windows user that is logged in. Unchecking it requires the user to enter their own username
- ✓ Set Allow Overwrite. Setting this option ensures a Windows user cannot easily remove the configuration data

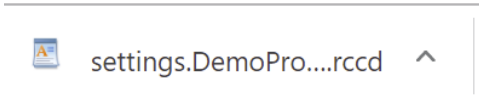
Select OK to save

Step 23: Finalize the KeyTalk configuration file



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address: ⓘ	keytalk.mydomain.com:443	☑ allow overwrite ⓘ
Logo: ⓘ	CHOOSE FILE	
Services:	My_SMIME_template	

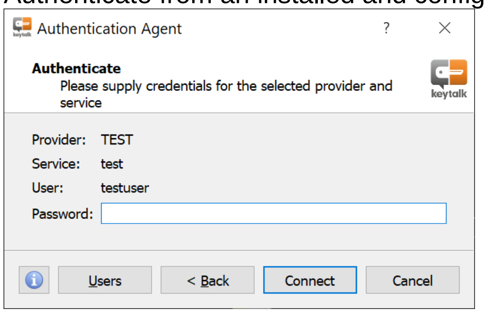
Step 24: Rename the created RCCD file to a name you find suitable/shorter:



settings.DemoPro....rccd

This RCCD file is effectively a ZIP file, you can rename it to inspect its contents.
Its typically deployed together with the KeyTalk client (see the individual KeyTalk client/app manuals), or it is email to users as an attachment, or made available as a public download (the file does not contain any secret information)

Step 25: Authenticate from an installed and configured KeyTalk client using Kerberos, or username/password


Authentication Agent

Authenticate
Please supply credentials for the selected provider and service

Provider: TEST
Service: test
User: testuser
Password:

Note1: Kerberos is currently only supported in Windows 7-10, and Server 2012R2-2019

Note2: With each KeyTalk installation two Windows Scheduler Tasks are activated, which are used to periodically verify if certificate replacement needs to trigger. Kindly read the KeyTalk Windows Client for more details

Verify if the certificate was indeed installed in the Windows System Certificate Store
Using the command MMC

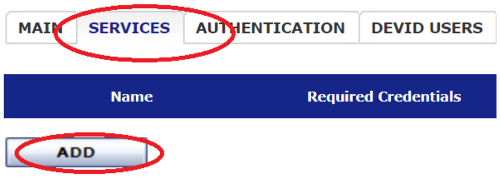
ANNEX F: Sample (web)server certificate use-case configuration

KeyTalk supports the issuance/renewal/(de)installation and automated binding of end-point server certificates using its KeyTalk client on these target servers. Alternatively an admin can generate and manually download the certificate and key-pair to manually install it on a target server.

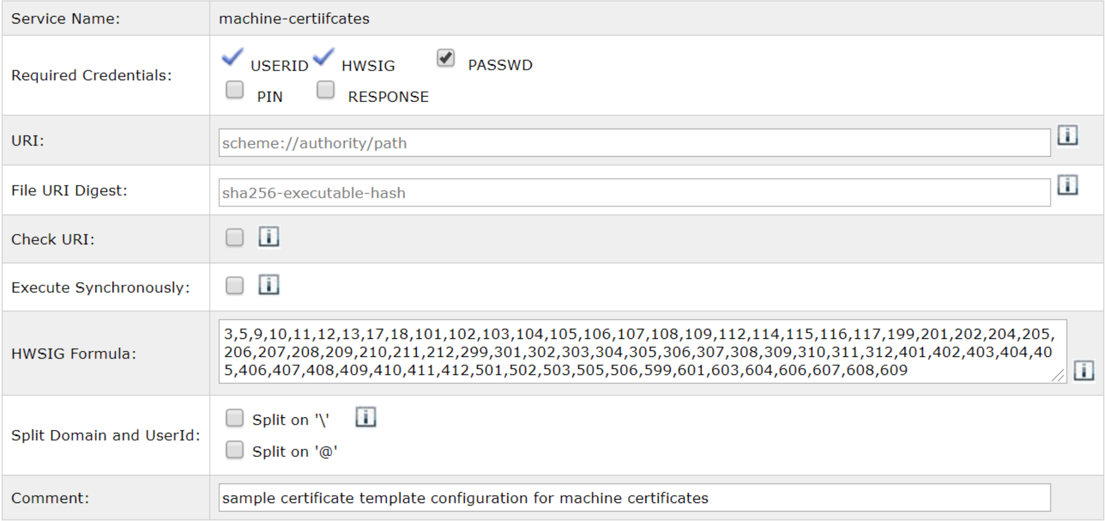
This sample (web)server certificate use-case assumes:

- ✓ the KeyTalk server has been generically configured already
- ✓ the KeyTalk internalDB is used as the primary source of server identities and authentication (contrary to for example an AD and Kerberos authentication
- ✓ No SNI applies (though it is supported)

Step 1: Configure a KeyTalk certificate template under TEMPLATES

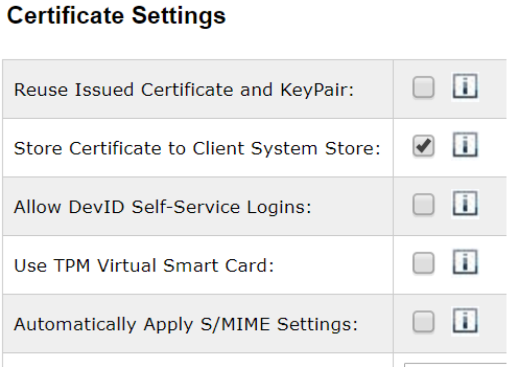


Step 2: Configure the TEMPLATE general settings.
Should you want the server to authenticate using more than just its hardware characteristics based preshared secret, select “password”.
Optionally make appropriate changes to the HWSIG Formula hardware/software characteristics.



Service Name:	machine-certiifcates
Required Credentials:	✓ USERID ✓ HWSIG ☑ PASSWD ☐ PIN ☐ RESPONSE
URI:	scheme://authority/path
File URI Digest:	sha256-executable-hash
Check URI:	☐
Execute Synchronously:	☐
HWSIG Formula:	3,5,9,10,11,12,13,17,18,101,102,103,104,105,106,107,108,109,112,114,115,116,117,199,201,202,204,205,206,207,208,209,210,211,212,299,301,302,303,304,305,306,307,308,309,310,311,312,401,402,403,404,405,406,407,408,409,410,411,412,501,502,503,505,506,599,601,603,604,606,607,608,609
Split Domain and UserId:	☐ Split on '\' ☐ Split on '@'
Comment:	sample certificate template configuration for machine certificates

Step 3: Configure certificate settings

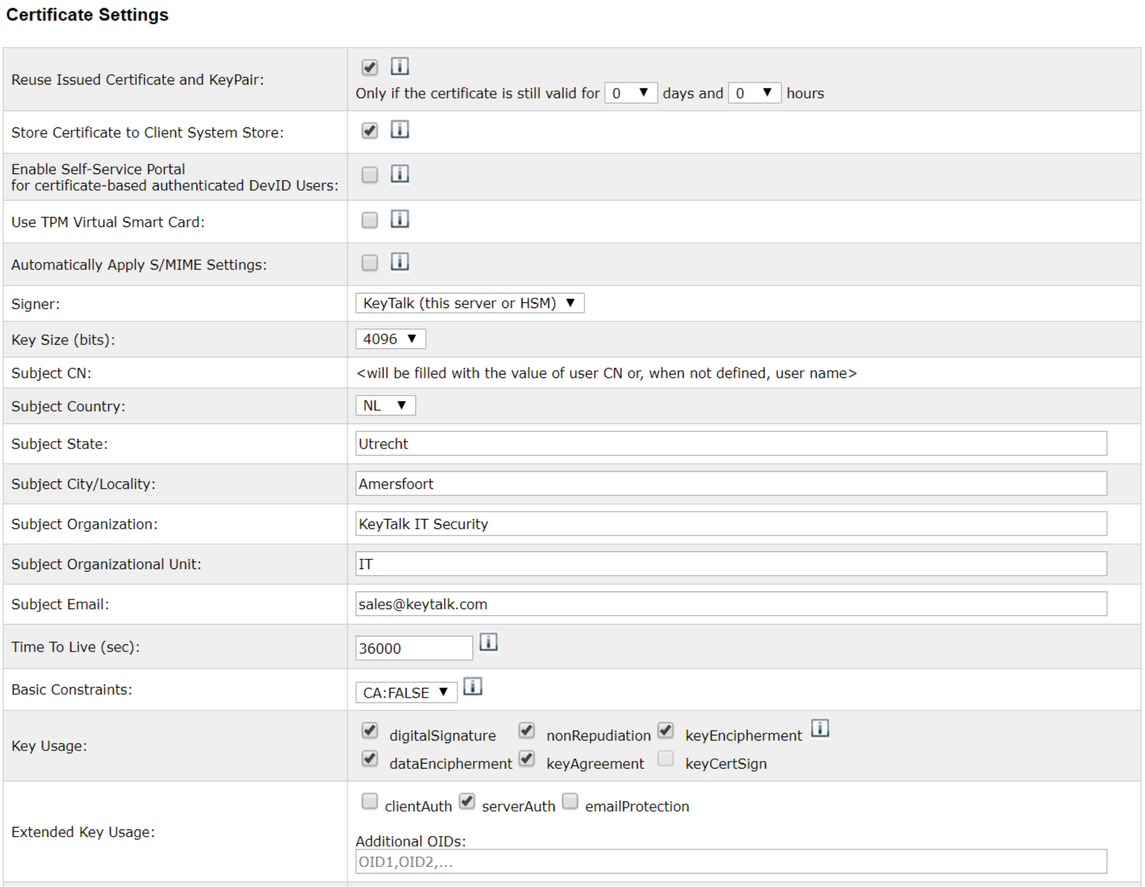


Certificate Settings

Reuse Issued Certificate and KeyPair:	□ ⓘ
Store Certificate to Client System Store:	☑ ⓘ
Allow DevID Self-Service Logins:	□ ⓘ
Use TPM Virtual Smart Card:	□ ⓘ
Automatically Apply S/MIME Settings:	□ ⓘ

- ✓ Set Store Certificate to Client System Store
- ✓ Optionally set “Reuse Issued Certificate and Keypair” as it will ensure you can automatically reissue the previously issued certificate and key in case it gets lost on the server or if it’s a wildcard and needs to be reused on other servers

Step 4a: When using KeyTalk Private CA:

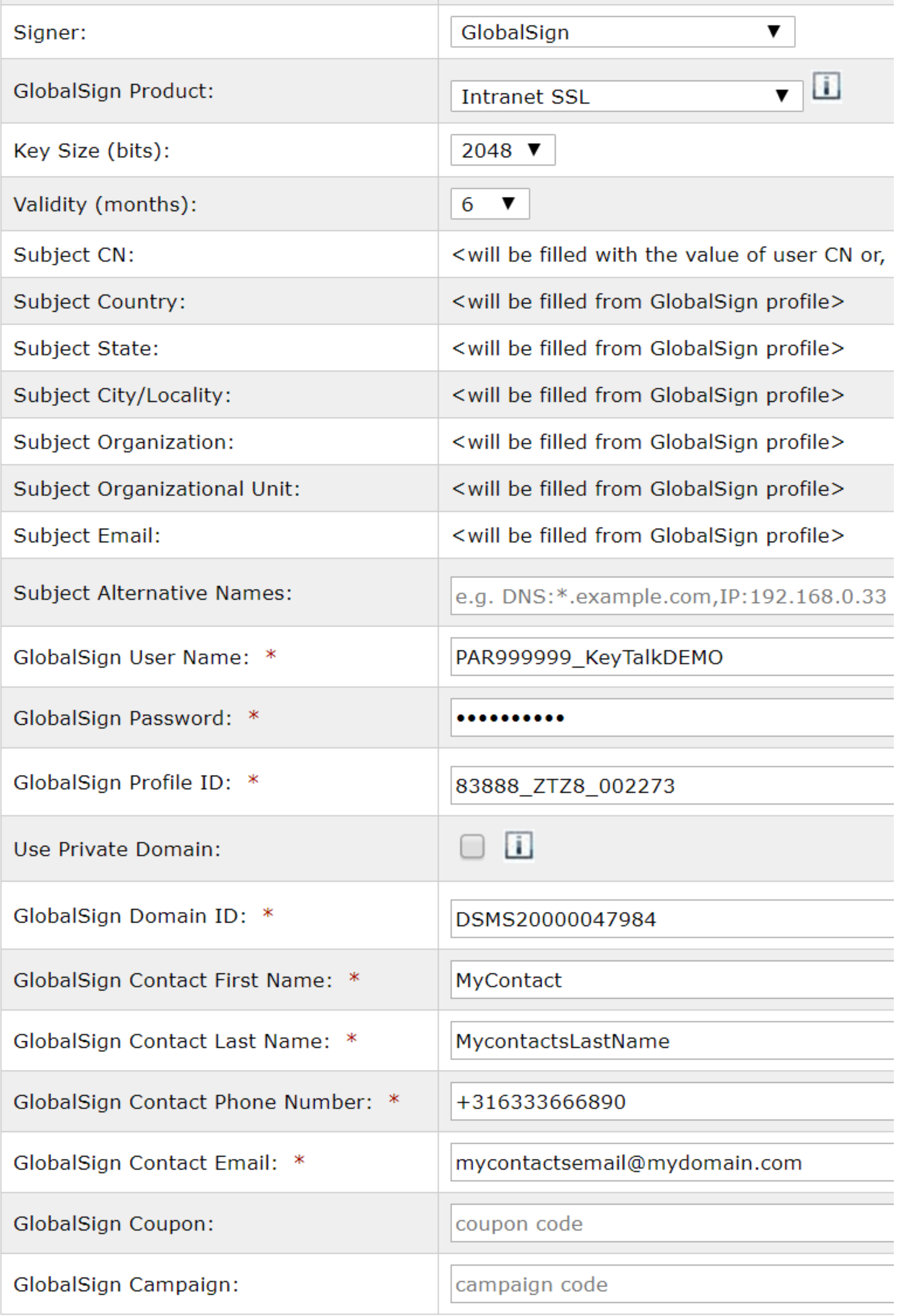


Certificate Settings

Reuse Issued Certificate and KeyPair:	☑ ⓘ Only if the certificate is still valid for 0 ▼ days and 0 ▼ hours
Store Certificate to Client System Store:	☑ ⓘ
Enable Self-Service Portal for certificate-based authenticated DevID Users:	□ ⓘ
Use TPM Virtual Smart Card:	□ ⓘ
Automatically Apply S/MIME Settings:	□ ⓘ
Signer:	KeyTalk (this server or HSM) ▼
Key Size (bits):	4096 ▼
Subject CN:	<will be filled with the value of user CN or, when not defined, user name>
Subject Country:	NL ▼
Subject State:	Utrecht
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk IT Security
Subject Organizational Unit:	IT
Subject Email:	sales@keytalk.com
Time To Live (sec):	36000 ⓘ
Basic Constraints:	CA:FALSE ▼ ⓘ
Key Usage:	☑ digitalSignature ☑ nonRepudiation ☑ keyEncipherment ⓘ ☑ dataEncipherment ☑ keyAgreement □ keyCertSign
Extended Key Usage:	□ clientAuth ☑ serverAuth □ emailProtection Additional OIDs: OID1,OID2,...

- ✓ Optionally set the subject information to match your own company/department
- ✓ Set Time to Live to (sec): 604800 (1 week), or use any other value you want
- ✓ Set Basic Constaints to CA:FALSE
- ✓ Set Extended Key Usage to: serverAuth

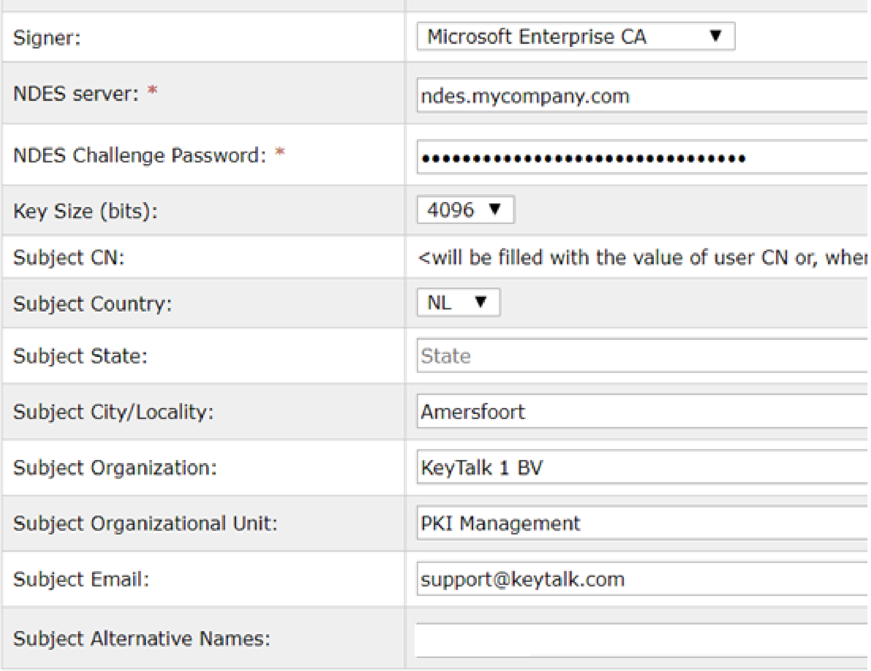
Step 4b: When using GlobalSign/TRUSTZONE:



Signer:	GlobalSign ▼
GlobalSign Product:	Intranet SSL ▼
Key Size (bits):	2048 ▼
Validity (months):	6 ▼
Subject CN:	<will be filled with the value of user CN or,
Subject Country:	<will be filled from GlobalSign profile>
Subject State:	<will be filled from GlobalSign profile>
Subject City/Locality:	<will be filled from GlobalSign profile>
Subject Organization:	<will be filled from GlobalSign profile>
Subject Organizational Unit:	<will be filled from GlobalSign profile>
Subject Email:	<will be filled from GlobalSign profile>
Subject Alternative Names:	e.g. DNS:*.example.com,IP:192.168.0.33
GlobalSign User Name: *	PAR999999_KeyTalkDEMO
GlobalSign Password: *	••••••••••
GlobalSign Profile ID: *	83888_ZTZ8_002273
Use Private Domain:	□
GlobalSign Domain ID: *	DSMS20000047984
GlobalSign Contact First Name: *	MyContact
GlobalSign Contact Last Name: *	MycontactsLastName
GlobalSign Contact Phone Number: *	+316333666890
GlobalSign Contact Email: *	mycontactsemail@mydomain.com
GlobalSign Coupon:	coupon code
GlobalSign Campaign:	campaign code

- ✓ Select product: IntranetSSL or Organization SSL or EV SSL
- ✓ Select the validity period that matches your GlobalSign/TRUSTZONE license
- ✓ Set your GlobalSign/TRUSTZONE User Name
- ✓ Set your GlobalSign/TRUSTZONE Password
- ✓ Set your GlobalSign/TRUSTZONE Profile ID as found in the GCC portal
- ✓ Set your contact details
- ✓ Optionally set your coupon or campaign code

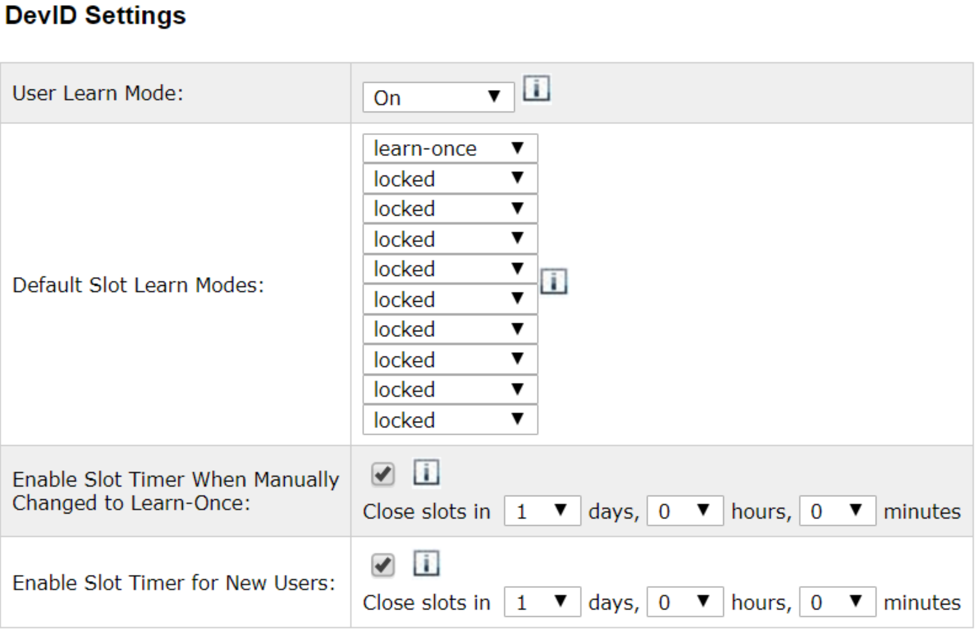
Step 4c: When using Microsoft AD Certificate Server:



Signer:	Microsoft Enterprise CA ▼
NDES server: *	ndes.mycompany.com
NDES Challenge Password: *	••••••••••••••••••••••••••••••••
Key Size (bits):	4096 ▼
Subject CN:	<will be filled with the value of user CN or, wher
Subject Country:	NL ▼
Subject State:	State
Subject City/Locality:	Amersfoort
Subject Organization:	KeyTalk 1 BV
Subject Organizational Unit:	PKI Management
Subject Email:	support@keytalk.com
Subject Alternative Names:	

- ✓ Set your NDES server with a static challenge.
  If you haven't set NDES, follow https://www.youtube.com/watch?v=JOwoMJmgi2g

Step 5: Configure automated device learning mode:



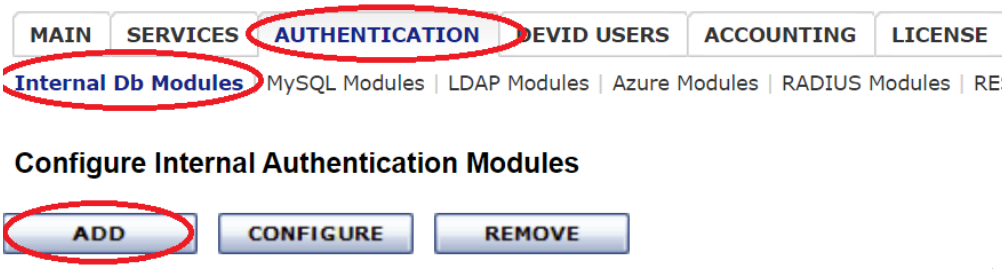
DevID Settings

User Learn Mode:	On ▼
Default Slot Learn Modes:	learn-once ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼ locked ▼
Enable Slot Timer When Manually Changed to Learn-Once:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes
Enable Slot Timer for New Users:	☑ Close slots in 1 ▼ days, 0 ▼ hours, 0 ▼ minutes

- ✓ Configure User Learn Mode to: On
- ✓ Configure slot 1 as : learn-once (ie issue to 1 server only)
- ✓ Set the timers

Step 7: Select OK to save the settings

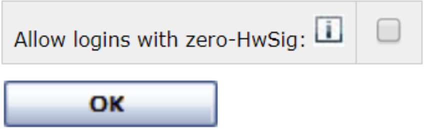
Step 8: Connect the Internal Db Module to the KeyTalk TEMPLATE certificate template



Configure Internal Authentication Modules

Step 9: Configure the added Internal DB Module

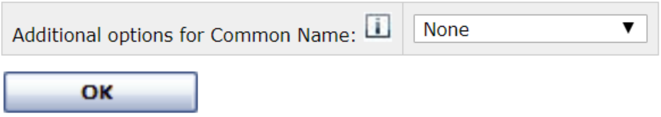
Step 10: Configure Hardware Signature requirement



Allow logins with zero-HwSig: □

- ✓ Leave unchecked (this enforces a non-zero value from a KeyTalk client/app when calculating a HardwareSignature)

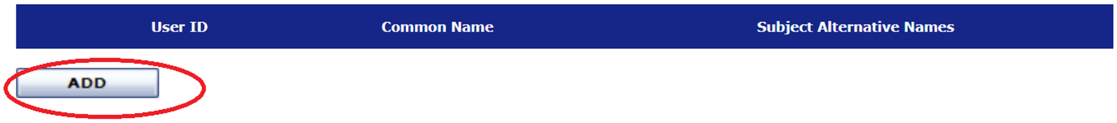
Step 11: **Configure Common Name options**



Additional options for Common Name: None ▼

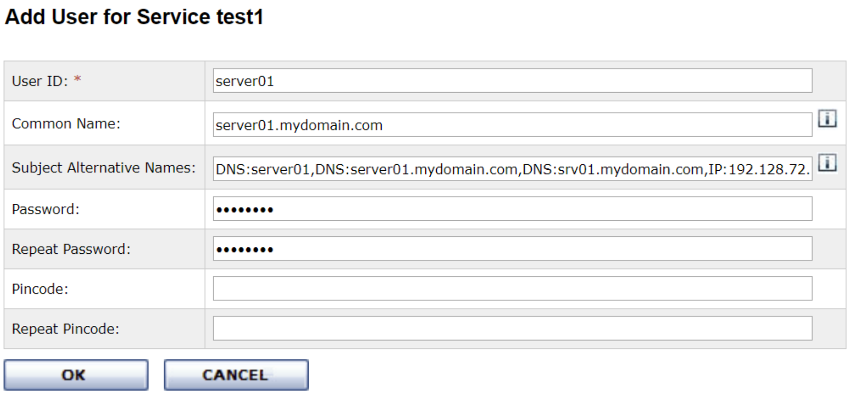
- ✓ Leave it to None

Step 12: **Add your first or Nth server end-point**



User ID	Common Name	Subject Alternative Names

Step 13: **Define the server end-point details**



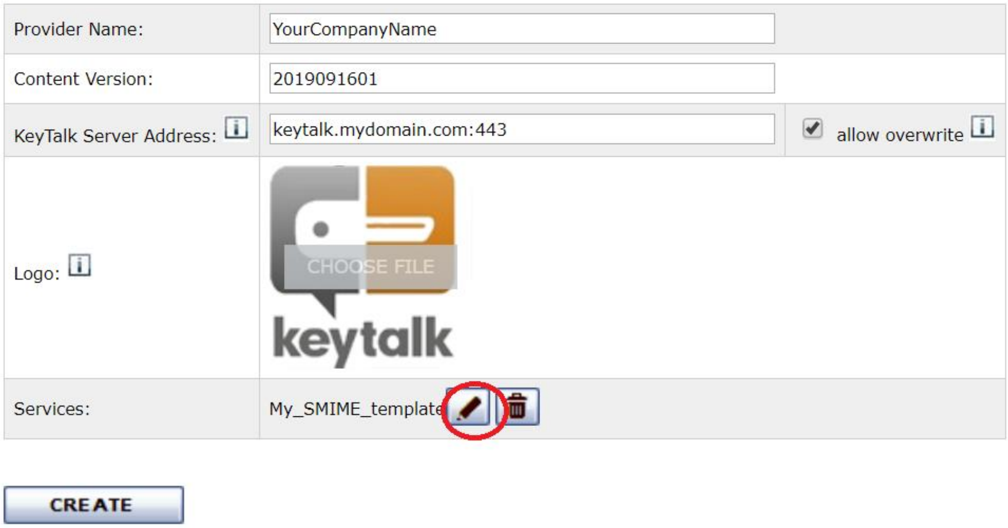
Add User for Service test1

User ID: *	server01
Common Name:	server01.mydomain.com
Subject Alternative Names:	DNS:server01,DNS:server01.mydomain.com,DNS:srv01.mydomain.com,IP:192.128.72.
Password:	••••••••
Repeat Password:	••••••••
Pincode:	
Repeat Pincode:	

- ✓ Configure the server user-ID (used to authenticate from the server client to KeyTalk CLM
- ✓ Define the Common Name
- ✓ Define the SAN
- ✓ Optionally set a password

Step 14: **Repeat steps 12-13 to add more servers**

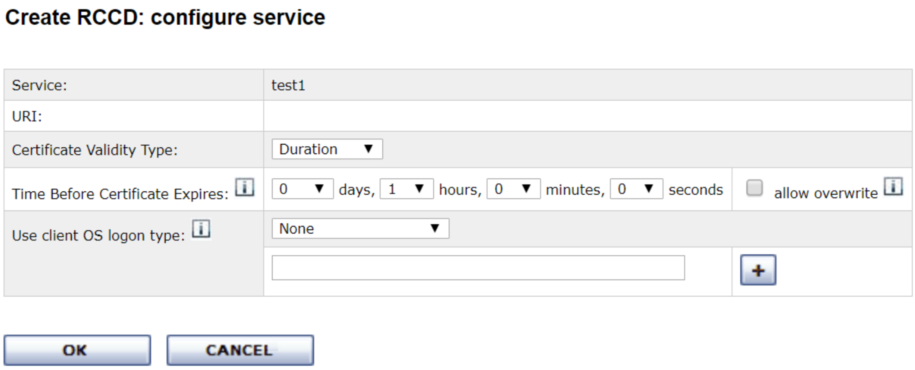
Step 15: **Create KeyTalk Real Client Configuration Data file.**
This file is used to configure the KeyTalk agent for every server that's been defined



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address:	keytalk.mydomain.com:443	☑ allow overwrite
Logo:	CHOOSE FILE	
Services:	My_SMIME_template	

- ✓ Set KeyTalk Provider Name (this is the name under which multiple KeyTalk configurations are grouped when imported into a single client)
- ✓ Set KeyTalk Server Address. This is either the FQDN or the IP address. Ensure the this address is also present in the KeyTalk "Certificates and Keys" in the SAN of the "Client-Server certificate"
- ✓ Set the port to 443 (generic use for KeyTalk client/app), or optionally to port 4443 when a Trusted Mobile SSL certificate and key has been uploaded.
- ✓ Optionally change the Logo to a 110x110 PNG of your choice
- ✓ Select the EDIT button next to the selected TEMPLATE

Step 16: **Configure KeyTalk Real Client Configuration Data meta data**



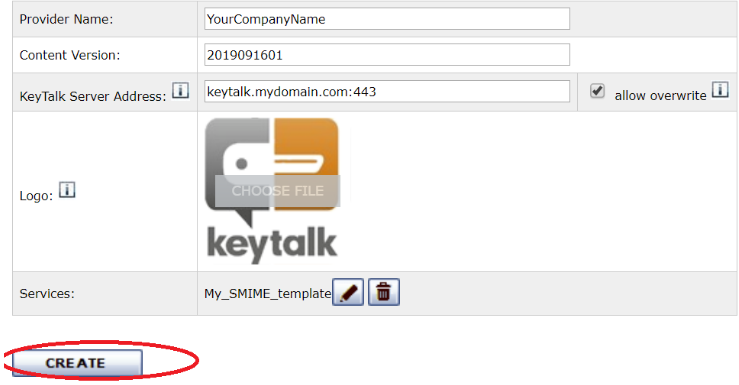
Create RCCD: configure service

Service:	test1	
URI:		
Certificate Validity Type:	Duration	
Time Before Certificate Expires:	0 days, 1 hours, 0 minutes, 0 seconds	□ allow overwrite
Use client OS logon type:	None	

- ✓ Set the duration time, which determines the amount of time before the certificate needs to be replaced before it expires.
- ✓ Set "Use client OS logon user" to None

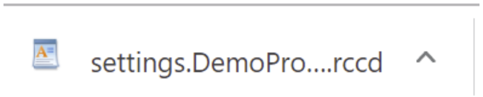
Select OK to save

Step 17: Finalize the KeyTalk configuration file



Provider Name:	YourCompanyName	
Content Version:	2019091601	
KeyTalk Server Address:	keytalk.mydomain.com:443	☑ allow overwrite
Logo:	CHOOSE FILE	
Services:	My_SMIME_template	

Step 18: Rename the created RCCD file to a name you find suitable/shorter:



This RCCD file is effectively a ZIP file, you can rename it to inspect its contents.
Its typically deployed together with the KeyTalk client (see the individual KeyTalk client/app manuals), or it is email to users as an attachment, or made available as a public download (the file does not contain any secret information)

Step 19: **Manually or mass deploy the KeyTalk client and manually or auto configure it**

For Windows servers with IIS install the KeyTalk client , import the RCCD file, and start the KeyTalk Configuration Manager with Administrator privileges and create a certificate task (most left menu option). For further details see the Windows client manual.

For Linux servers with Apache/TomCat etc, install the KeyTalk client in accordance to the Linux OS specific manual included in the client download, import the RCCD and follow the steps in the Linux manual.

ANNEX G: Migrating from KeyTalk 5.8.14 to KeyTalk 6.x

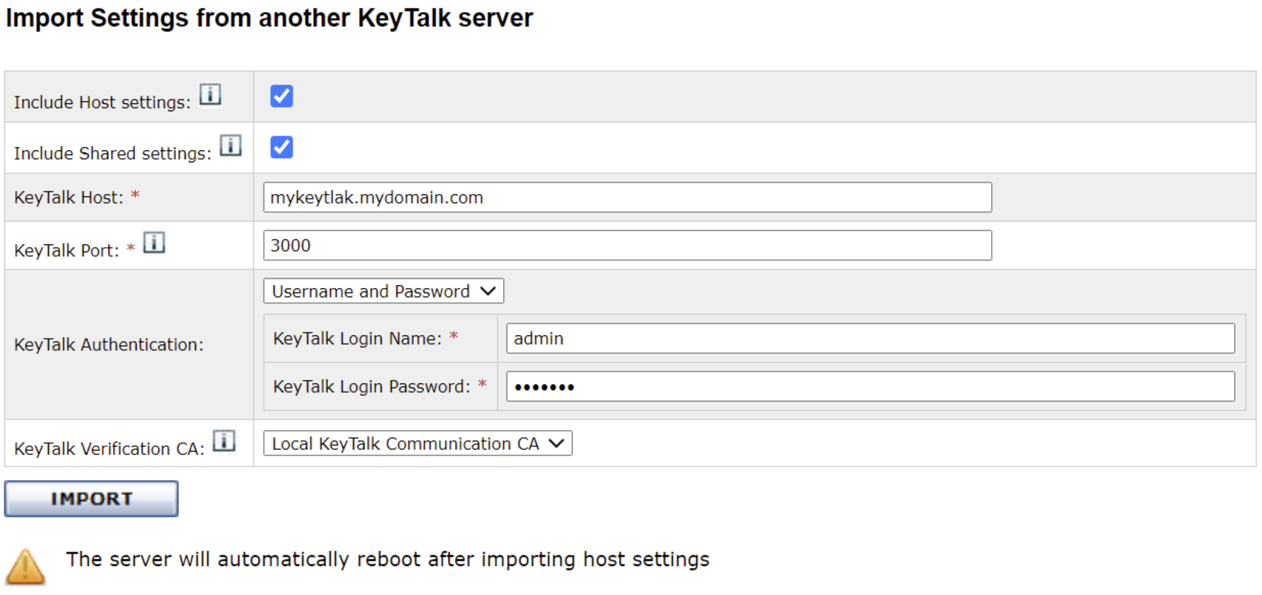
Should you already have an existing KeyTalk CKMS 5.x , make sure you backup the current version, and upgrade it first to the latest KeyTalk 5 firmware version: 5.8.14.
The latest firmware version can be found here: https://downloads.keytalk.com/downloads/upgrade/keytalk.svr.fw-5.8.14-production.dat

When you have an externally configured KeyTalk MySQL Db, it will remain compatible with KeyTalk 6.1.0. Simply transfer the network settings as the shared data remains in the MySQL instance.
However to ensure future firmware updates are properly processed, ensure your MySQL OS instance is fully updated (sudo apt update / sudo apt upgrade) and ensure the MySQL DB can handle files of at least 512 MB

There are two methods to migrate to KeyTalk 6.x :

1) When your KeyTalk 5.8.14 environment is relatively small:
   Step 1: Create a full settings backup of KeyTalk 5.8.14
   Step 2: Deploy your KeyTalk 6.x instance and provide basic networking configurations
   Step 3: If the settings backup is less than 500MB, upload it into KeyTalk 6.x
   Step 4: Finalize your DNS and LoadBalancer configurations
   Step 5: Apply KeyTalk 6.x firmware upgrades when available (www.keytalk.com/support)

2) When your KeyTalk 5.8.14 environment is relatively large (ie settings backup is larger than 500MB)
   Step 1: Create a full settings backup of KeyTalk 5.8.14
   Step 2: Deploy your KeyTalk 6.x instance and provide basic networking configurations
   Step 3: In KeyTalk 5.8.14 go to: SYSTEM -> SETTINGS, and configure a KeyTalk authentication username and password
   Step 4: In KeyTalk 5.8.14 go to: CERTIFICATES AND KEYS -> Communication CA -> Download the certificate as PEM (do NOT include the private key)
   Step 5: Ensure firewall/network wise that KeyTalk 6.x is allowed to communicate to KeyTalk 5.8.14 over port 3000
   Step 5: In KeyTalk 6.x, go to: SYSTEM -> SETTINGS, and configure the same KeyTalk authentication username and password as configured for KeyTalk 5.8.14
   Step 6: In KeyTalk 6.x, go to: SYSTEM -> SETTINGS, and upload the KeyTalk 5.8.14 Communication CA PEM certificate
   Step 7: Select : IMPORT
   Step 8: When the import is done, finalize your DNS and LoadBalancer configurations
   Step 9: Apply KeyTalk 6.x firmware upgrades when available (www.keytalk.com/support)



Import Settings from another KeyTalk server

Include Host settings:	☑	
Include Shared settings:	☑	
KeyTalk Host: *	mykeytlak.mydomain.com	
KeyTalk Port: *	3000	
KeyTalk Authentication:	Username and Password	
	KeyTalk Login Name: *	admin
	KeyTalk Login Password: *	•••••••
KeyTalk Verification CA:	Local KeyTalk Communication CA	

The server will automatically reboot after importing host settings