# Configuration Manager MicroCA

BOSCH

**en** Operation Manual

# Table of contents

# 1 Introduction

## 1.1 About this manual

This manual should help you to manage certificates using the Configuration Manager MicroCA feature.
This document assumes that the reader is familiar with the system and the other programs integrated into the system.

## 1.2 Conventions in this document

The following symbols and notations are used to draw attention to special situations:

**Notice!**
This symbol indicates special features and provides tips and information for easier, more convenient use of the software.

Terms that you can find in the program, such as menu options, commands or text in the user interface, are written in **bold**.

## 1.3 Additional documentation

**Documentation and software for Bosch Security Systems products can be found in the online product catalogue as follows:**

- Open any browser > enter www.boschsecurity.com > select your region and your country > start a search for your product > select the product in the search results to show the existing files.

# 2 System overview

## 2.1 Background information

The Configuration Manager MicroCA functionality eases the management of small to medium systems deploying certificate device authentication and certificate-based user authentication.
Each certificate consists of the following parts:

– A publicly available certificate with the public key
– A corresponding private key

For highest level of security, the private key must be concealed in hardware, a physical key store, typically performed by a Trusted Platform Module (TPM) chip. For this purpose, Bosch cameras include a TPM chip. Use a USB or smart card crypto token for MicroCA use to guarantee exclusive ownership.
For test purposes, or in case of low expectations on measures against stolen keys, you may also store the private key and certificate on a standard USB flash stick as PKCS12 file.

**Notice!**
Weak protection by PKCS12 implementations
Malware on the PC may create an unnoticed copy and crack the PIN due to weak encryption of most PKCS12 implementations. Never use PKCS12 implementations in security-critical applications.

**Very high protection through certificate-based authentication**
Certificate based authentication allows you to create closed systems with very high protection against malicious access. This certification mechanism allows you to set up distributed camera systems that reach security level 3 of FIPS-140-2 standard.
However, note that before the initial creation of certificates on the devices no technical means can hinder so-called man in the middle attacks. Preferably use a secure environment to roll-out the initial certificates to your devices.

## 2.2 Initializing the MicroCA

The MicroCA functionality in the Configuration Manager program is an easy-to-use tiny certificate authority (CA).
Create the CA certificate as follows:

– Provide the certificate information
– Select the location as token or file
– Select a key strength

After the CA certificate is created, it can be immediately used for signing other certificates.
When using a file-based CA certificate make sure to store it on a USB flash stick kept in a safe place. We also recommend that you create a security copy to reduce the risk of losing your CA certificate.
Preferably, use a USB token or smart card. Check the release notes for a list of supported crypto hardware.

## 2.3 Creating signed device certificates

The most secure mechanism to generate device certificates is a two-step approach.

– First generate a signing request on the device. This step creates a private key within the device's crypto hardware module, guaranteeing exclusive access by the device.
– In a second step the signing request is downloaded, signed and uploaded back to the device.

For signing, you will need your MicroCA crypto token or USB drive, and you need to enter the MicroCA PIN to authorize its use.

## 2.4 Creating User Token

User tokens are dedicated hardware, like a smart card or a crypto USB stick, which carry a user certificate for authentication.
Smart cards are well-known devices for user authentication, although in principle you may deploy any other certificate technology for this purpose. The following items must be configured:

- Load a root certificate to the devices and assign it the client trust role.
- Create a smart card for each user with the username as so-called “common name”.
- Enable certificate-based login on the devices.
- On the devices configure the login role for each of the users.

# 3 Configuring MicroCA

## 3.1 Smart Token

**Creating a Smart Token**

1. In Configuration Manager program, navigate to **Preferences** > **Applications** > Configuration Manager.
2. Click the **Security** tab.

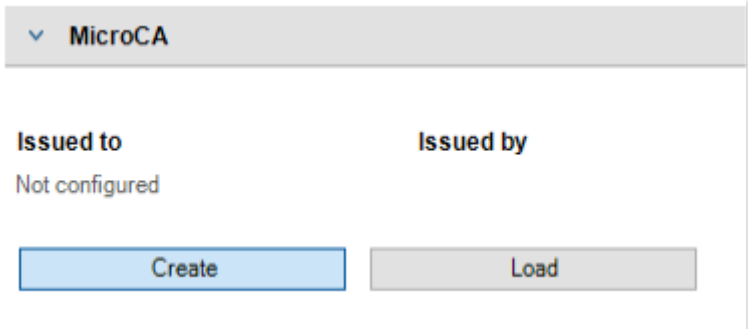


**Figure 3.1: Configuration page with no CA configured**

3. Click **Create**. The **Create CA** dialog box is displayed.

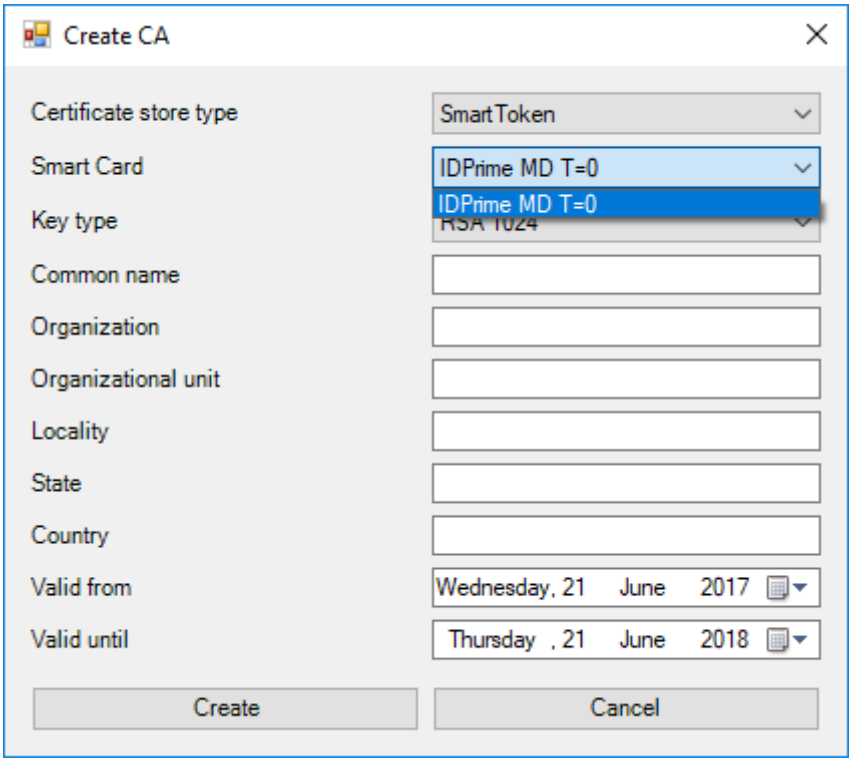


**Figure 3.2: Create CA dialog box.**

4. In the **Certificate store type** list, click **Smart Token**.
5. In the **Common name** box, enter a meaningful name for the new Certificate Authority.
6. In the **Valid from** and **Valid until** lists, click the desired start and end date.
   **Note:** Since the MicroCA functionality has no provisions to prolong validity, make sure that you select an appropriate period of time.
7. Fill out the remaining fields. In larger installations, this information will help you to identify the authority.

8. In the **Key type** list, select an entry.
   The list contains different key sizes and two different key types: the classical RSA type and the ECDSA type, a so-called Diffie-Hellman exchange type. While RSA is much more common, Diffie-Hellman has lower computational overhead. Although mixing both types on different tokens is possible, we recommend that you use the same type for all tokens.
   **Note:** Higher numbers reflect higher levels of security. For example, RSA 2048 is more secure than RSA 1024, but requires more computation time.

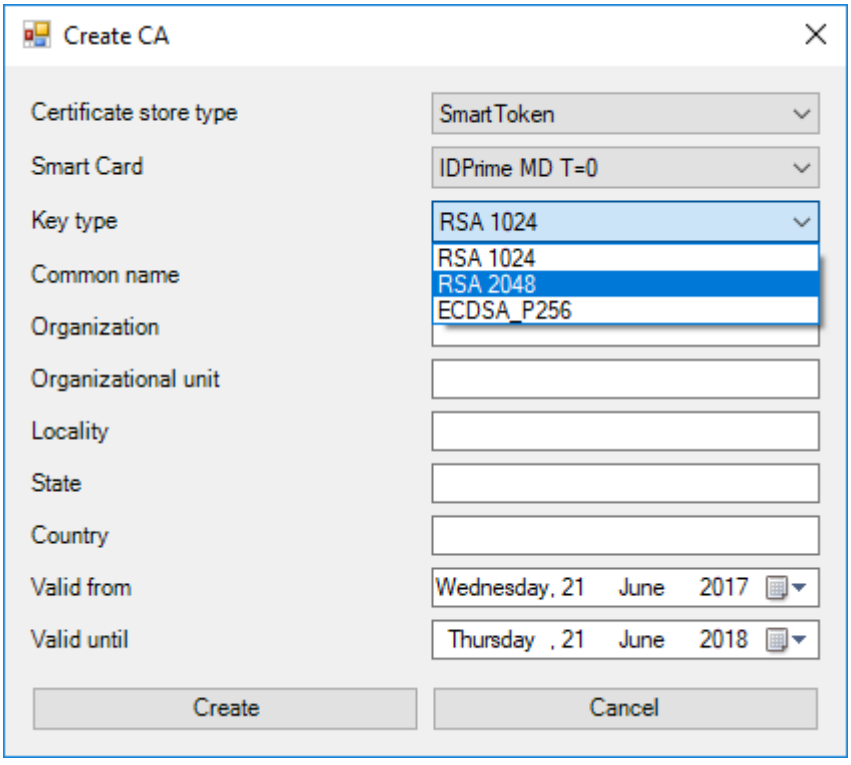


**Figure 3.3: Key type list**

9. Type the smart card PIN to be authorized using the private key including self-signing.

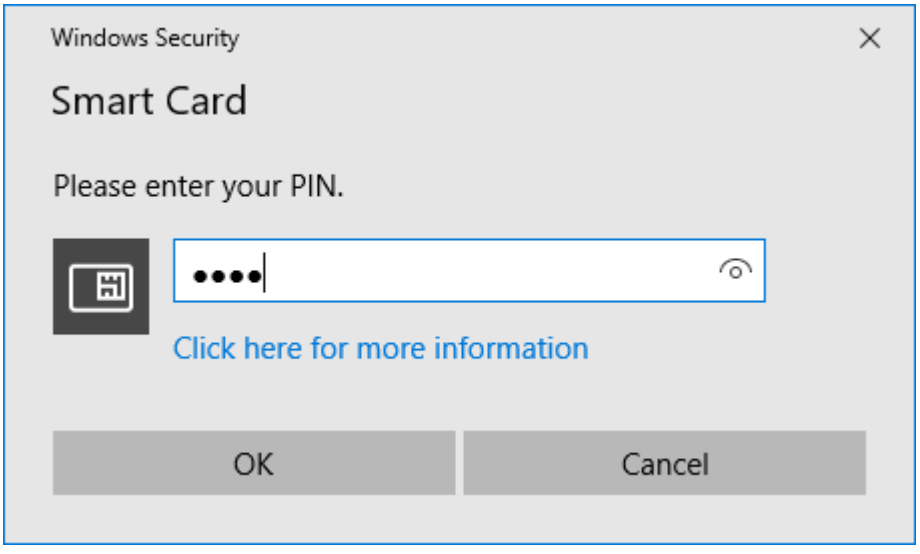


**Figure 3.4: Smart card PIN**

A new Certificate Authority is displayed in the MicroCA list.

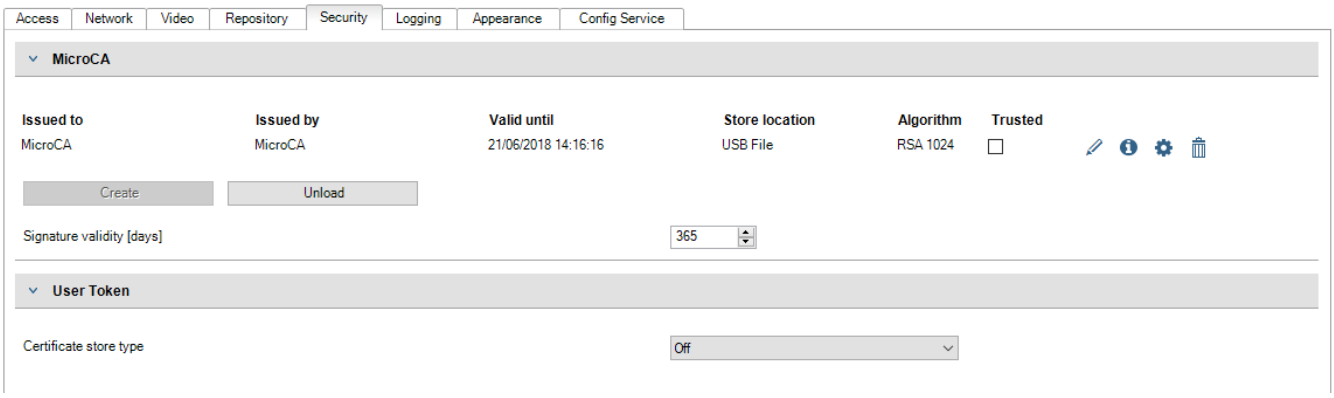


**Figure 3.5: MicroCA with configured Certificate Authority**

1. In the MicroCA list entry, click the **Trusted** check box. A security warning is displayed.
   **Note:** The **Trusted** check box facilitates to add MicroCA to the Windows **Trusted Certificates** list.
   Applications, for example, the Chrome browser, identifies the certificate as valid.
2. To confirm, click **Yes**.

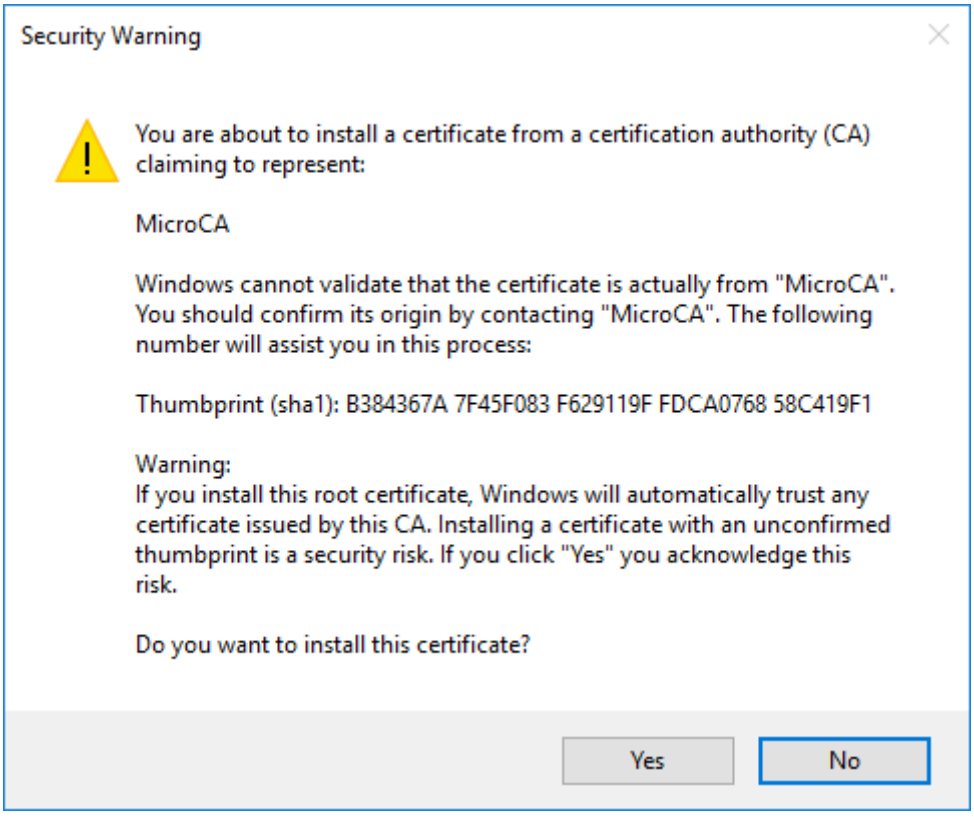


**Figure 3.6: Microsoft internal warning on changes of the trusted list**

## 3.2 USB file

1. In Configuration Manager program, navigate to **Preferences** > **Applications** > Configuration Manager.
2. Click the **Security** tab.

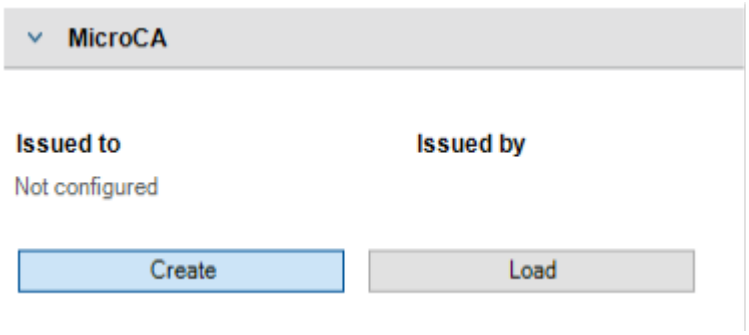


**Figure 3.7: Configuration page with no CA configured**

3. Click **Create**. The **Create CA** dialog box is displayed.

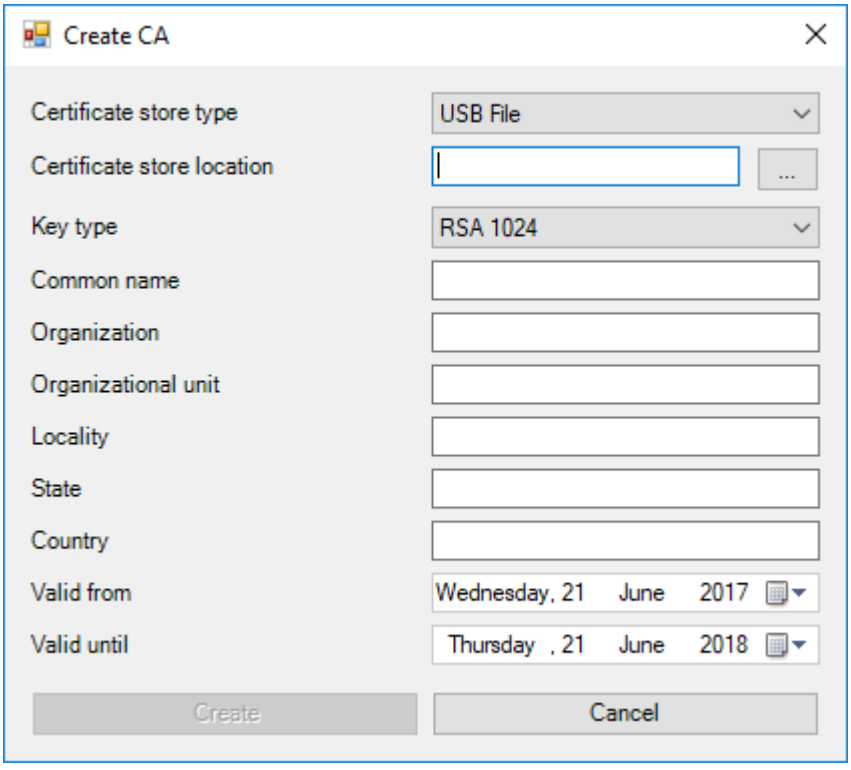


**Figure 3.8: Certificate details**

4. In the **Certificate store type** list, click **USB File**.
5. Insert a USB stick into your system and select a **Certificate store location** on it.
6. In the **Key type** list, select an entry.
   The list contains different key sizes and two different key types: the classical RSA type and the ECDSA type, a so-called Diffie-Hellman exchange type. While RSA is much more common, Diffie-Hellman has lower computational overhead. Although mixing both types on different tokens is possible, we recommend that you use the same type for all tokens.
   **Note:** Higher numbers reflect higher levels of security. For example, RSA 2048 is more secure than RSA 1024, but requires more computation time.

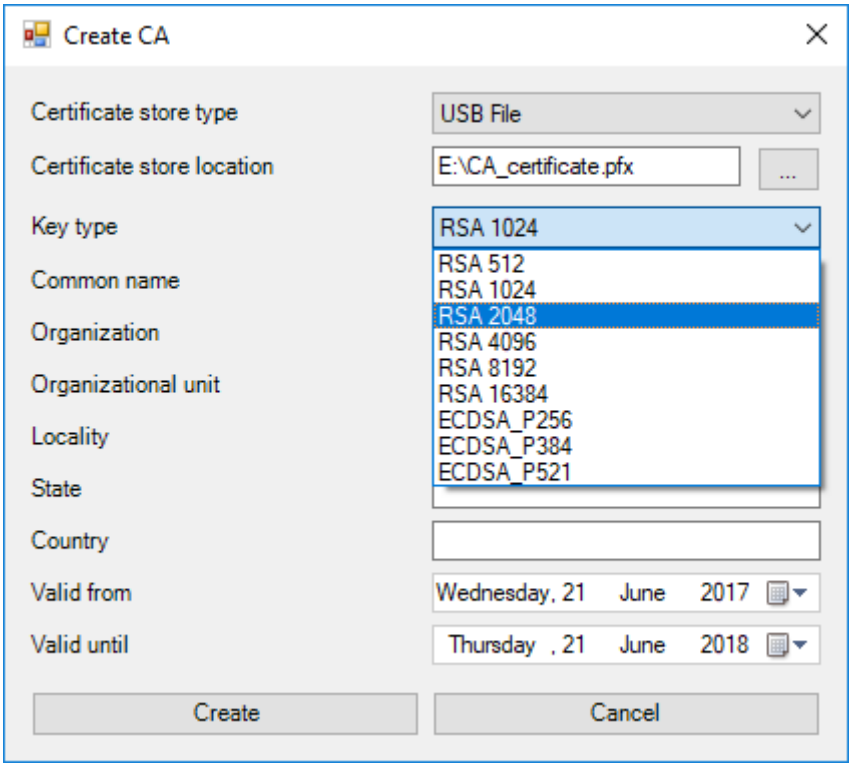


**Figure 3.9: Key type and length**

7. In the **Common name** box, enter a meaningful name for the new Certificate Authority.
8. In the **Valid from** and **Valid until** lists, click the desired start and end date.
   **Note:** Since the MicroCA functionality has no provisions to prolong validity, make sure that you select an appropriate period of time.
9. Fill out the remaining fields. In larger installations, this information will help you to identify the authority.
10. Click **Create** to open the **Generate Certificate** dialog box.
11. To confirm the creation of a new certificate, click **OK**.
12. A password dialog box opens. Type a new password. While you type, the **Password** box will change its color from red (very weak password) to yellow (weak password) to green (very strong password). Use a combination of characters, digits, and special characters to achieve a very strong password.
13. In the **Confirm** box, type the same password.
14. To create the certificate, click **OK**.

A new Certificate Authority is displayed in the MicroCA list.

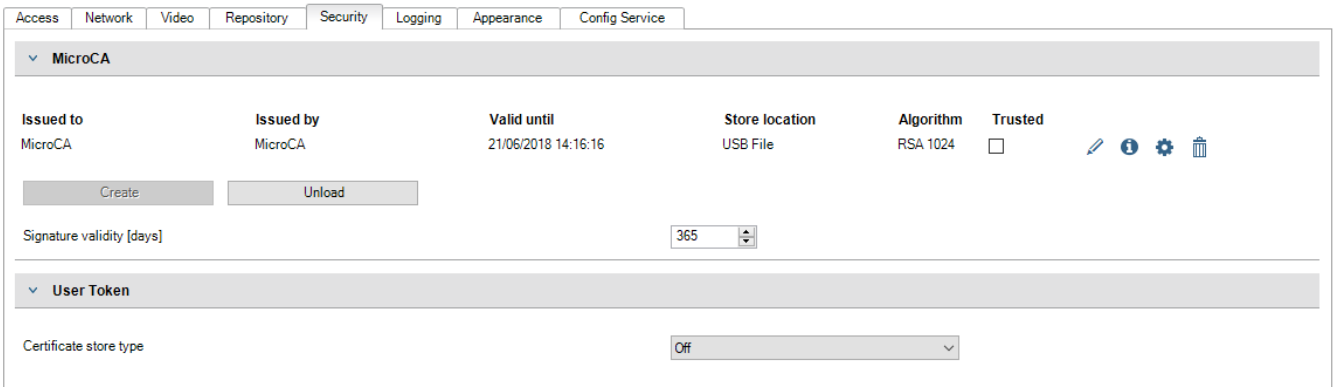


**Figure 3.10: MicroCA with configured Certificate Authority**

# 4 Signing device certificates

One of the main purposes of the MicroCA functionality is to deploy certificates to devices. To achieve this, you will replace a self-signed certificate by a MicroCA signed certificate. In order to secure device access by using certificates you need to change the devices authentication mode.

1. In the Configuration Manager program, click the **Devices** or **My Devices** tab, then click the desired device.
2. Click the **General** tab, then click the **Unit Access** tab.
3. Find the **Allowed authentication modes** section.
4. Click the upload icon .
   A message box will inform you that MicroCA certificate is active on your system and that you can upload the MicroCA certificate.
5. Click **Yes** to start certificate-based authentication on the device.
   After successfully uploading the MicroCA certificate, the device needs a restart in order to engage certificate handling.
6. Confirm the restart by clicking **Yes** when the message box appears.

Wait for the device to be online again. In order to verify the successful switching to certificate based authentication go to the **Service > Certificates** page of the device. You will find a MicroCA certificate similar to the one shown here:

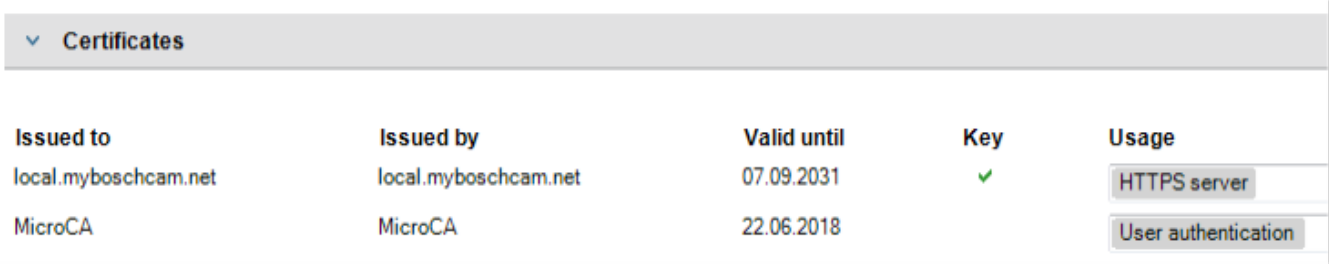


**Figure 4.1: Example device certificate listing**

1. To create a signing request, click **Generate signing request**. The **Generate signing request** dialog box is displayed.

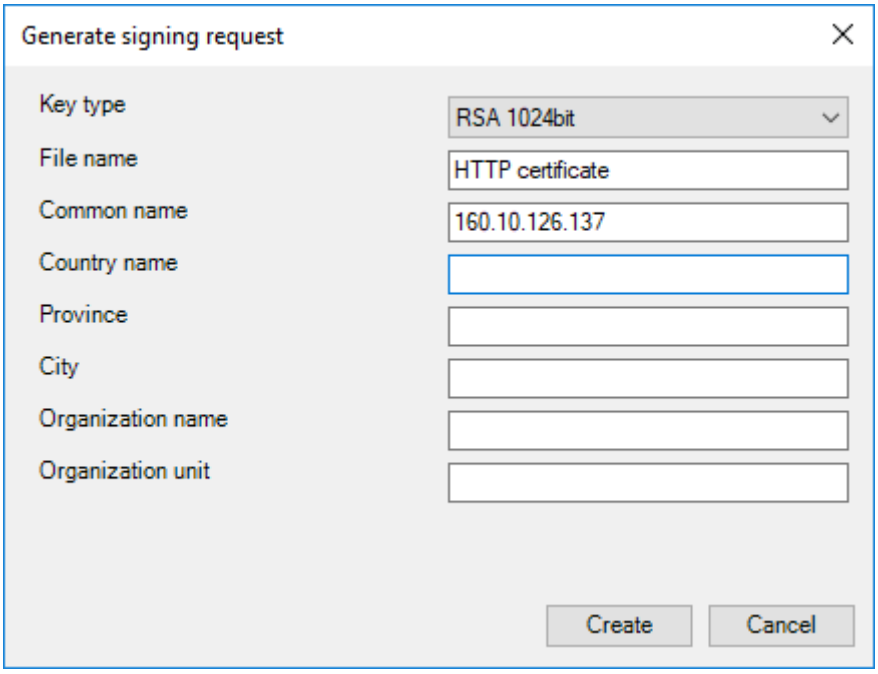


**Figure 4.2: Generate a signing request dialog box**

2. In the **Common name** box, the IP address of the device is displayed. Do not change this!
3. The remaining boxes are filled from the MicroCA certificate and can be adapted according to your needs.
4. Click **Create**.
   **Note:** The creation of the certificate request may take some time due to the key creation process.

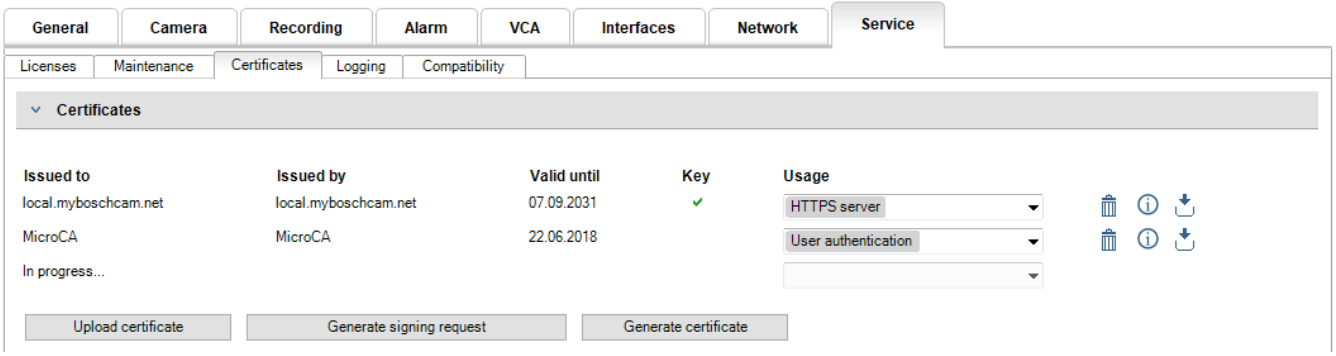


**Figure 4.3: Certificate request – creation in progress**

5. To sign and upload the certificate, click the reload icon or press **F5** to update until the line shows a valid signing request.
   **Note:** The sign icon is available after the MicroCA has been configured. The sign icon allows you to sign and upload the signed certificate in a single step.

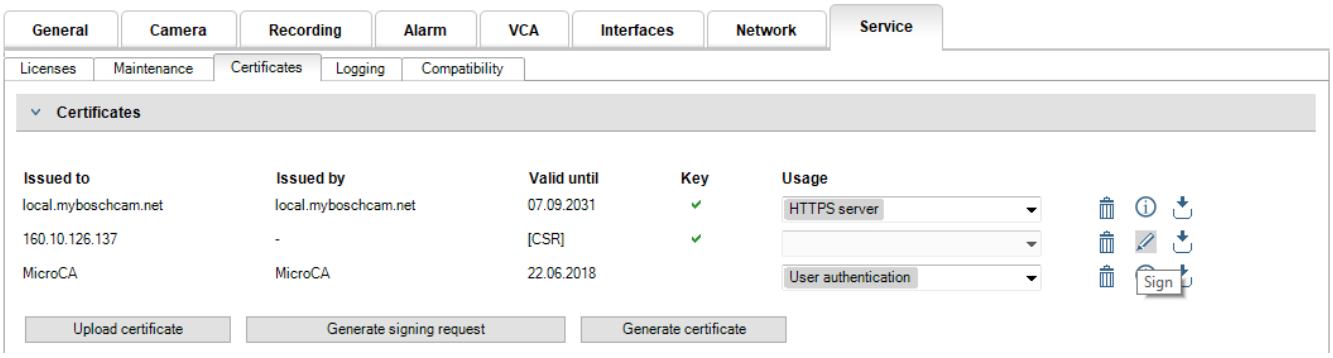


**Figure 4.4: Certificate request – ready for signing**

6. Click the sign icon to the right of the certificate's list entry. You may be asked to insert your smart card and/or to type your PIN to authorize the action.
7. After the certificate is signed you may switch its **Usage** to **HTTPS server**:

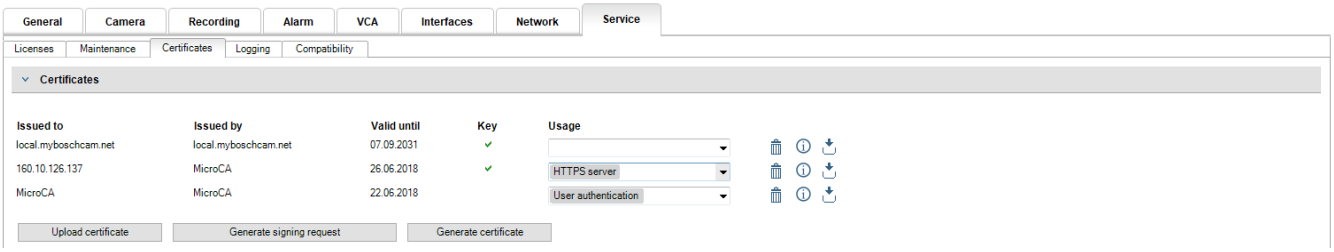


**Figure 4.5: Signed signing request shown as certificate with private key**

8. Restart the device. After the restart, the newly created signed certificate will apply as a TLS communication encryption certificate.

# 5 User Token

A user token - also known as security token - is a physical device which can be used to gain access to an electronically secured computer. A user token can be used as a replacement or in addition to a password. MicroCA certificate uses smart cards or (crypto-) USB sticks as the token hardware.
The user token contains a private key which will be tested against the public key of the MicroCA certificate. Only if this test is successful, access to the device or to the video software will be granted.

## 5.1 Managing tokens

To manage tokens:

1. In Configuration Manager program, navigate to **Preferences** > **Applications** > Configuration Manager.
   The **User Token** section of the **Security** page allows that you inspect existing tokens. Smart tokens and PKCS12 files on USB sticks are supported.
   **Note:** A list of existing tokens known to your system can be displayed if you click the **Certificate store type** list.

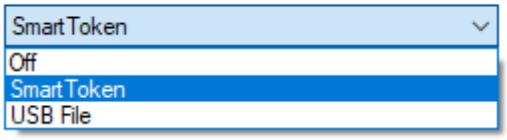


**Figure 5.1: Token type selection**

2. In the **Certificate store type** list, click the corresponding entry.
3. Select a certificate. For the following reasons, more than one certificate can be displayed in the list:
   – You have inserted multiple different tokens into your system.
   – A single token contains multiple certificates.

For each certificate two functions are available:
– Showing detailed certificate information
– Deleting the certificate from the token

**Note:** Use caution when deleting token information. You cannot recover the token information.

## 5.2 Creating tokens

User token creation is similar to certificate creation.
To create user token:

1. On the **Security** page proceed as follows:
   – Insert a smart card, click **Smart Token** and select the smart card.
   or
   – Select **USB File** and enter a path and a new file name.
2. Click **Create**. The **Generate and sign key pair** dialog box is displayed.
3. In the **Common name** box, enter a meaningful name for the new Certificate Authority.
4. In the **Valid from** and **Valid until** lists, click the desired start and end date.
   **Note:** Since the MicroCA functionality has no provisions to prolong validity, make sure that you select an appropriate period of time.
5. Fill out the remaining fields. In larger installations, this information will help you to identify the authority.

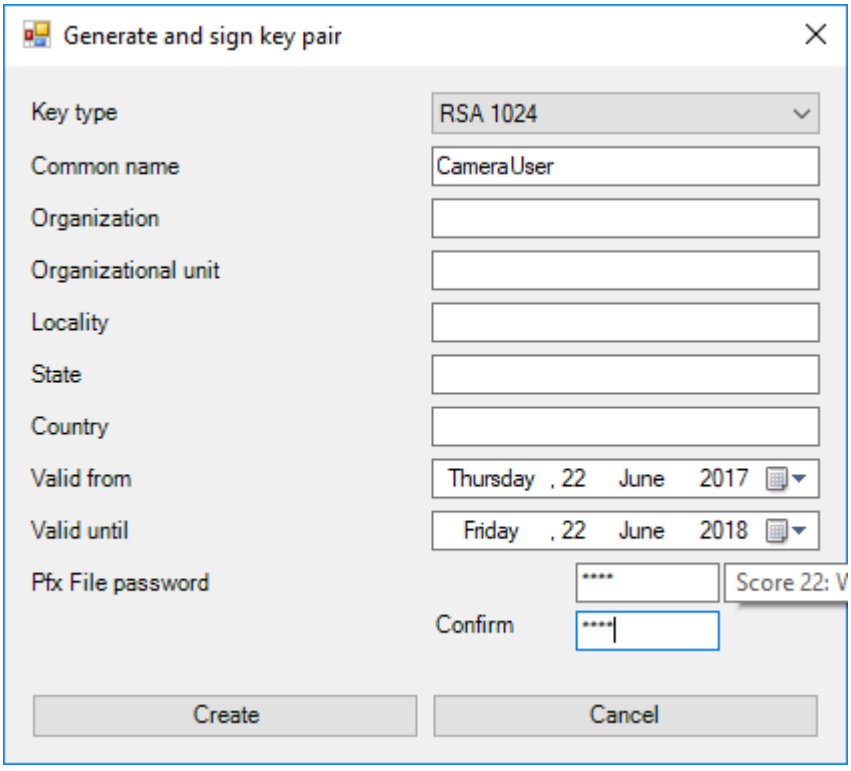


**Figure 5.2: Generate and sign key pair dialog box**

6. Click **Create** to submit your data.
   **Note:** To allow the creation of a valid user token, the system needs access to the CA certificate. Insert a smart card with a valid CA certificate and authorize its use by entering the CA PIN and the user token pin.

## 5.3 Configuring token-based device authentication

Add the user to the device's list of users:

1. Click the **General** tab, then click the **Unit Access** tab. Find the **Users** section.
2. Click **Add user** in the user panel. This will open the **Add User** dialog.
3. From **Type** select **Certificate**.
4. Choose a **Group** to specify the user's role.
5. Enter the **User name**. This must match the **Common name** that you used while creating the User token!
6. Click **Create** to finish the procedure.
7. Finally, you need to activate the new authentication mode. Go to the **Allowed authentication modes** section and click the **Certificate** checkbox. A green checkmark will appear to state that the new mode is active.

**Bosch Security Systems B.V.**
Torenallee 49
5617 BA Eindhoven
Netherlands
**www.boschsecurity.com**