

# Certificate Management

---

## DESCRIPTION

**Copyright**

© Ericsson AB 2014, 2015, 2016. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

**Disclaimer**

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

**Trademark List**

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



# Contents

|          |                                 |           |
|----------|---------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>             | <b>1</b>  |
| <b>2</b> | <b>Functions and Concepts</b>   | <b>3</b>  |
| 2.1      | Types of Operation              | 3         |
| <b>3</b> | <b>Managed Object Model</b>     | <b>9</b>  |
| <b>4</b> | <b>Configuration Management</b> | <b>11</b> |
| <b>5</b> | <b>Fault Management</b>         | <b>13</b> |
| <b>6</b> | <b>Security Management</b>      | <b>15</b> |





# 1 Introduction

This document provides an overview of the management model and concepts associated with the Certificate Management managed area.

A managed area is represented by a group of Managed Object Classes (MOCs) within the Managed Object Model (MOM).

It is assumed that the users of this document are familiar with the following:

- Basics of the terms and concepts of ITU-T X.509 certificates
- Basics of the Certificate Management Protocol (CMP) for online enrollment





## 2 Functions and Concepts

Certificate Management provides a management interface for certificate information, which is stored on the Managed Element (ME) and used for authentication and encryption purposes.

A certificate file is a proof of ownership of a public key for a subject. The subject can be a person, company, or other entity such as an ME. A certificate can be used in various ways, for example, for authentication. Certificate Authority (CA) is an entity that issues certificates.

Certificate Management handles the following two major certificate categories:

- Node Credential consists of a (node) certificate and associated private key.

The node certificate is used as a proof of identity of the ME to other network elements. The private key is known by the ME only, and can be used by the secured services in the ME for various cryptographic actions. An ME can have several Node Credentials that can be used for different purposes, for example, one for O&M and another for IP Security (IPsec).

- Trusted certificate is a certificate of a CA that the operator trusts. The ME uses the CA certificate to verify the CA signature on the peer certificate, to ensure that the peer certificate is valid, before using the peer certificate to establish a secure connection.

**Note:** If the peer node expects to receive the chain certificates from the ME (for example, in Transport Layer Security (TLS) handshake), chain certificates must be installed as trusted certificates.

For example, the CA trusted by the operator issues the certificate to a Lightweight Directory Access Protocol (LDAP) server. The certificate of this trusted CA is configured in the ME as a trusted certificate, and the ME verifies the LDAP server identity against the trusted certificate. A node credential containing valid node certificate must be configured in the ME to allow the LDAP server to accept LDAP/TLS connection from the ME.

### 2.1 Types of Operation

Certificate Management supports the operations described in this section.

Node certificates can be enrolled in the following ways:

- Certificate Signing Request (CSR) based offline enrollment
- PKCS#12 container-based offline enrollment
- CMP version 2 (CMPv2) based online enrollment

Trusted certificates are not subject to an enrollment operation triggered by the ME and are installed directly on the ME.

### 2.1.1 CSR-Based Offline Enrollment

The user can perform offline enrollment of a node certificate based on a CSR. The CSR is a PKCS#10 container file, prepared in the ME, and taken to a CA to enroll into a certificate that can be installed on the ME.

An overview of CSR-based offline enrollment is shown in Figure 1.

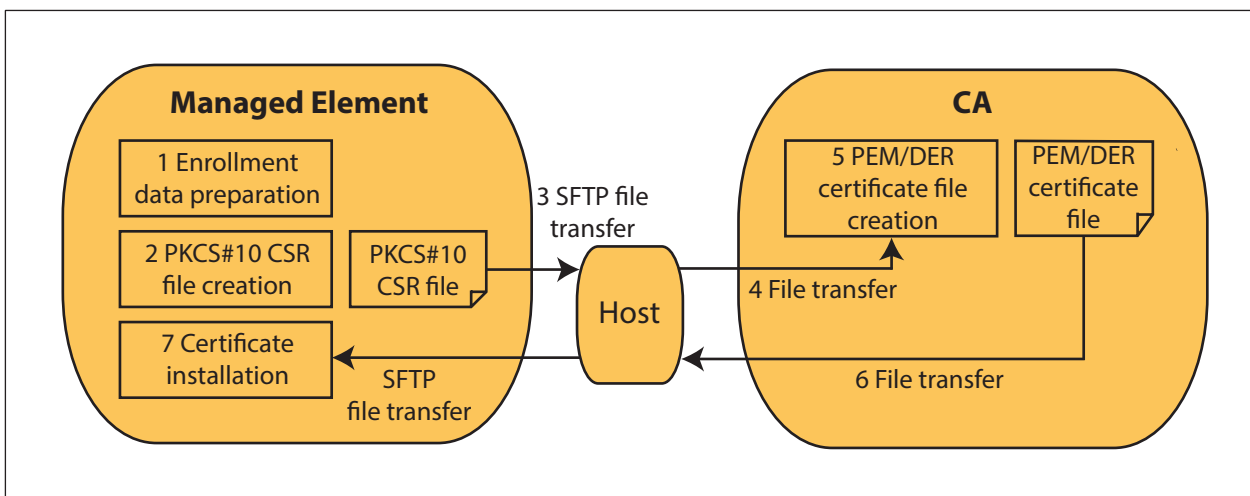


Figure 1 CSR-Based Offline Enrollment

The installation or renewal of a node credential by CSR consists of the following main phases:

1. In the ME, the user enters the data needed for the certificate generation.
2. The ME generates a private and public key pair for the certificate. The public key is written in the CSR file. The private key remains secret and stored in the ME.
3. The user downloads the CSR file from the ME to an external host with the SSH File Transfer Protocol (SFTP).
4. The user submits the CSR file further from the external host to the CA and requests the CA to generate a certificate from the file.
5. The CA generates the certificate file from the CSR file. The certificate is signed with the public key of the CA and written to a Privacy Enhanced Mail (PEM) or Distinguished Encoding Rules (DER) file.
6. The user receives the certificate file on the external host from the CA. With this file, the user receives information of the correct fingerprint calculated from the file.





7. The user installs the certificate file in the ME. During this phase, the ME downloads the file with the SFTP and calculates the fingerprint from the file. The ME compares the result with the fingerprint value that the user entered. If the fingerprint values are equal, the certificate file is installed on the ME. Otherwise the file is discarded as invalid or corrupt for the node certificate in question. The successful installation results in a Node Credential, which associates the installed node certificate and the private key generated in Step 2.

For further details on how to perform this operation, refer to *Install or Renew Node Credential by CSR*.

**Note:** The fingerprint, also known as digest, is used to control that a certificate file has not been compromised. The fingerprint value is an optional input parameter for certificate file installation. For further details on how to perform this operation, refer to *Generate Fingerprint for File*.

### 2.1.2

### PKCS#12 Container-Based Offline Enrollment

The user can perform offline enrollment of a node certificate by installing a PKCS#12 container file. The file is issued by a CA and consists of a private key and a certificate. This enrollment is essentially a one-step offline enrollment.

An overview of the installation or renewal of a node credential by PKCS#12 is shown in Figure 2.

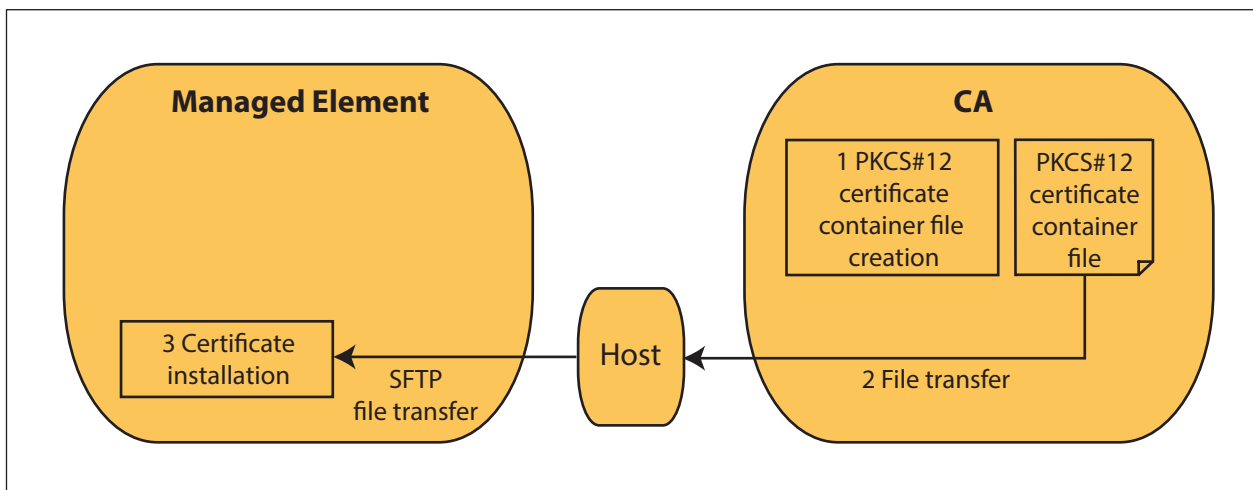


Figure 2 Installation or Renewal of a Node Credential by PKCS#12

The installation or renewal of a node credential by PKCS#12 consists of the following main phases:

1. The user requests a PKCS#12 certificate container file from the CA. The CA generates a private and public key pair for the certificate and writes both keys in the PKCS#12 file.



2. The user receives the PKCS#12 file from the CA into an external host. With the file, the user receives the password needed for decrypting the file and information of the correct fingerprint calculated from the file.
3. The user installs the PKCS#12 file in the ME. During this phase, the ME downloads the file from the external host with the SFTP. The ME decrypts the downloaded file with an encryption password, provided by the user. The ME also calculates the fingerprint from the downloaded file. The ME compares the calculated fingerprint with the fingerprint value provided by the user. If the fingerprint values are equal, the PKCS#12 file is installed on the ME. Otherwise the file is discarded as invalid or corrupt for the node certificate in question. The successful installation results in a Node Credential, which associates the installed node certificate and the private key generated in Step 1.

**Note:** In addition to the public key, the PKCS#12 certificate container file also contains the private key for the node certificate. Thus, the PKCS#12 file received from the CA is protected with encryption. The user must specify the decryption key (parameter `credentialPassword`) when installing the file. This enables the ME to decrypt the file.

For further details on how to perform this operation, refer to *Install or Renew Node Credential by PKCS 12*.

**Note:** The fingerprint value is an optional input parameter for certificate file installation. For further details on how to perform this operation, refer to *Generate Fingerprint for File*.

### 2.1.3 CMPv2-Based Online Enrollment

The CMPv2 protocol is used to provide online certificate enrollment and automatic certificate renewal. It is an alternative to offline enrollment methods and the main advantages are to reduce operating expenses and remove the risk of undesired certificate expiration.

Overviews of the installation and renewal of a node credential online are shown in Figure 3 and Figure 4. For further details on how to perform these operations, refer to *Install Node Credential Online* and *Renew Node Credential Online*.

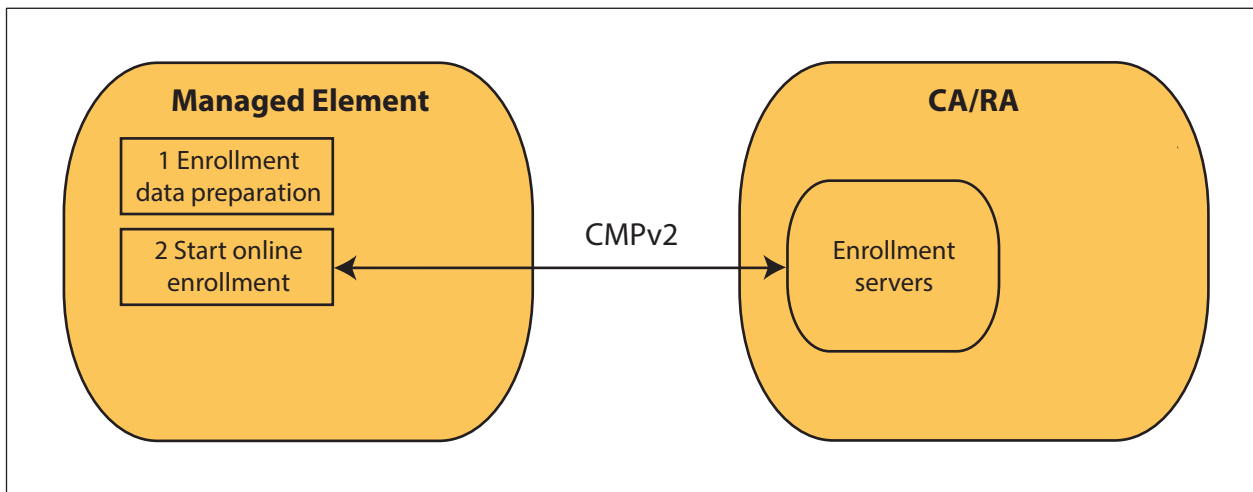


Figure 3 Installation of a Node Credential Online

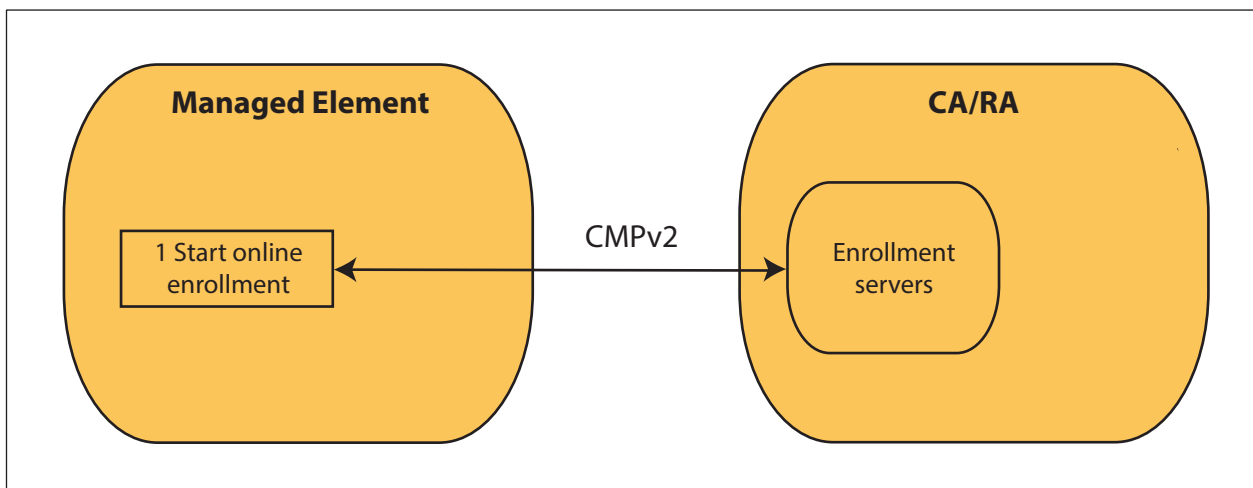


Figure 4 Renewal of a Node Credential Online

For online enrollments, an enrollment authority and an enrollment server must be preconfigured. An enrollment server is the front end to a CA or a Registration Authority (RA). To perform load balancing, servers can be grouped into server groups. Inside the server groups, an algorithm local to the ME can select any server to initiate the enrollment request. For further details on how to perform these operations, refer to *Configure Enrollment Authority*, *Configure Enrollment Server Group Together with Enrollment Servers*, and *Change Enrollment Server*.

The ME and the CA/RA provide a CMPv2 client and a CMPv2 server, respectively. The ME information, including enrollment password, must have been configured to a CA before starting initial online enrollment. Details vary depending on the CA solution used and are not described in this document.

**Note:** If automatic renewal is used, the ME automatically updates the certificate when it is about to expire.

## 2.1.4 Trusted Certificate Installation

The user can install trusted certificates on the ME. The certificates are used to authenticate the communication peers. The user can organize trusted certificates into trust categories.

An overview of the installation of a trusted certificate is shown in Figure 5.

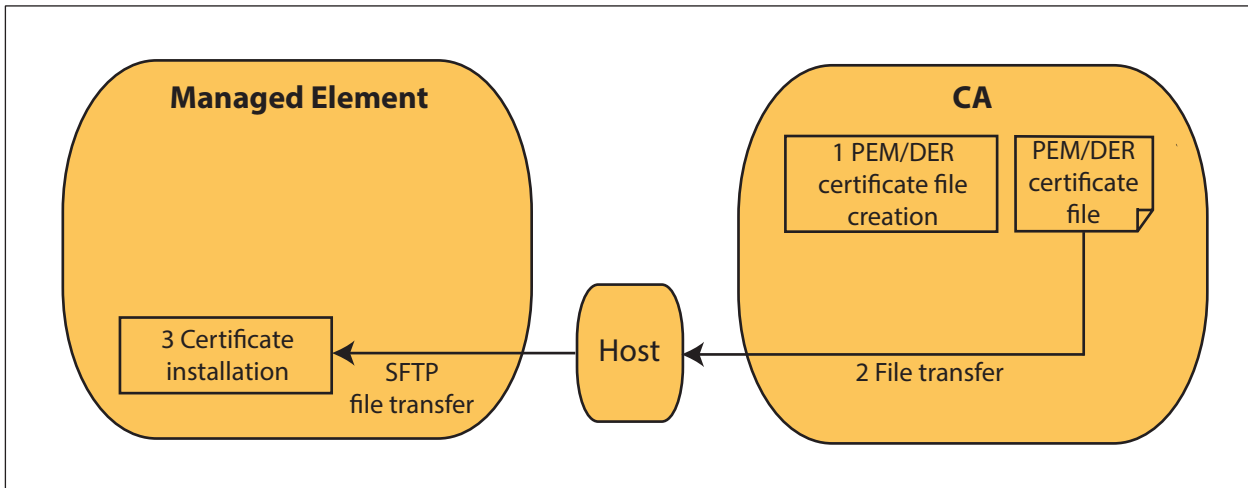


Figure 5 Installation of a Trusted Certificate

Trust categories can be referred by ME functions that intend to trust and use the corresponding certificates. Such ME functions play a credential user role.

The installation of a trusted certificate consists of the following main phases:

1. The CA prepares a PEM or DER certificate file for a trusted peer.
2. The user receives the CA certificate file in an external host. With the file, the user receives information of the fingerprint calculated from the file.
3. The user installs the certificate file on the ME as trusted certificate. During this phase, the ME downloads the file from the external host with the SFTP and calculates the fingerprint from the downloaded file. The ME compares the result with the fingerprint value provided by the user. If the fingerprint values are equal, the certificate file is installed on the ME. Otherwise the file is discarded as invalid or corrupt for the trusted certificate in question.

For further details on how to perform this operation, refer to *Install Trusted Certificate*.

For details on how to manage trusted certificates, refer to *Create Trust Category*, *Change Trust Category*, *Enable Trusted Certificate*, and *Disable Trusted Certificate*.

**Note:** The fingerprint value is an optional input parameter for certificate file installation. For further details on how to perform this operation, refer to *Generate Fingerprint for File*.

### 3 Managed Object Model

The Certificate Management managed area is represented in the *Managed Object Model (MOM)* as follows:

```
ManagedElement
+-SystemFunctions
+-SecM
+-CertM
+-CertMCapabilities
+-EnrollmentAuthority
+-EnrollmentServerGroup
+-EnrollmentServer
+-NodeCredential
+-TrustCategory
+-TrustedCertificate
```

For general information about the MOM, MOCs, Managed Objects (MOs), cardinality, and related concepts, refer to *Managed Object Model User Guide*.

The Certificate Management MOCs are described in Table 1.

**Table 1** Certificate Management Managed Object Class Descriptions

| Managed Object Class         | Description  |
|------------------------------|--|
| <i>CertM</i>                 | The root of the Certificate Management model, handles installation and deletion of trusted certificates.   |
| <i>CertMCapabilities</i>     | Describes the supported Certificate Management capabilities; enrollment and fingerprint support.   |
| <i>EnrollmentAuthority</i>   | Contains the configured certification and registration authorities.  |
| <i>EnrollmentServerGroup</i> | Contains the configured groups of enrollment servers used for load balancing purpose.  |
| <i>EnrollmentServer</i>      | Contains the configured CMP enrollment servers.  |
| <i>NodeCredential</i>        | Handles installation and deletion of node credentials. Contains the installed node credentials and information about the corresponding certificates. |
| <i>TrustCategory</i>         | Contains the configured trust categories, which are groups of trusted certificates that can be referenced by other MOs.                              |
| <i>TrustedCertificate</i>    | Contains the installed trusted certificates.   |





## 4 Configuration Management

Certificate Management is accessed using NETCONF or the Ericsson Command-Line Interface (ECLI) to manipulate the Management Information Base (MIB).

The following operations can be performed by the user and are described in Operating Instructions using the ECLI:

### **Manage Node Credentials Offline**

- *Install or Renew Node Credential by CSR*
- *Cancel Offline Enrollment*
- *Install or Renew Node Credential by PKCS 12*
- *Generate Fingerprint for File*

### **Manage Node Credentials Online**

- *Configure Enrollment Authority*
- *Configure Enrollment Server Group Together with Enrollment Servers*
- *Change Enrollment Server*
- *Install Node Credential Online*
- *Renew Node Credential Online*
- *Configure Renewal Mode of Node Credential*
- *Delete Enrollment Server*
- *Delete Enrollment Authority*
- *Delete Enrollment Server Group*

### **Delete Node Credentials Offline or Online**

- *Delete Node Credential*

### **Manage Trusted Certificates**

- *Install Trusted Certificate*
- *Create Trust Category*



- *Change Trust Category*
- *Delete Trust Category*
- *Disable Trusted Certificate*
- *Enable Trusted Certificate*
- *Delete Trusted Certificate*





## 5 Fault Management

The Certificate Management alarms are described in Table 2.

*Table 2 Certificate Management Alarm*

| <b>Alarm</b>  | <b>Description</b>  |
|---|---|
| <i>Certificate Management, a Valid Certificate is not Available</i> | Issued when the certificate is expired, revoked, or unavailable, which can result in failure of a secure service using this certificate.            |
| <i>Certificate Management, Automatic Enrollment Failed</i>          | Issued when automatic enrollment or renewal failed to execute because of local misconfiguration or remote enrollment service denial.                |
| <i>Certificate Management, the Certificate is to Expire</i>         | Issued when the threshold before certificate expiration has been crossed, and the certificate is to be renewed to prevent a secure service failure. |





## 6 Security Management

The System Security Administrator role is required for the operations described in this document.

With the System Administrator role, certificate-related information can only be read.

For details about this role, refer to *User Management*.