secunet eID PKI Suite

Certified CA Kernel

Version 1.0.0

Security Target

Version 1.07, October 8th, 2015

Certification-ID: BSI-DSZ-CC-0960

secunet
secunet Security Networks AG

Contents

Figures

Tables

History

Version	Date	Change(s)	Author(s)
0.9	09/09/14	First version of Security Target	Christian Koob Marcus Meier
0.91	26/09/14	Minor changes (History added, CLI of CA-Core manager deleted, Figure 4 changed, bootstrapping description)	Christian Koob
0.92	10/10/14	Certification-ID added and some changes according to Kick-Off	Christian Koob
0.93	31/10/14	Minor changes due to feedback from evaluation facility (mainly rationale) Audit functionality modified Role Operator removed HSM operates in FIPS mode Cryptographic operations modified	Christian Koob
0.94	10/11/14	Minor changes due to feedback from evaluation facility	Christian Koob
0.95	22/12/14	Changes due to feedback from certification body from 01/12/14. Replay detection added. TOE description modified.	Christian Koob
0.96	17/02/15	Changes due to feedback from certification body from 16/02/15.	Christian Koob
0.97	01/04/15	Replay detection added as bullet point in major security features	Christian Koob
0.98	22/04/15	A.HSM and OE.HSM added.	Christian Koob
0.99	21/05/15	New TOE name: secunet eID PKI Suite Certified CA Kernel Version numbers changed/added	Christian Koob
1.00	06/07/15	Final Version	Christian Koob
1.01	14/07/15	Utimaco HSM package version changed Version number of Manual and Security Target changed.	Christian Koob

1.02	18/08/2015	Versions of Guidance Documents changed	Christian Koob
1.03	21/08/2015	Table 1 edited. OE.Cryptographic functions in chapter 8.1.1.13 changed. P.Cryptography in Chapter 8.1.1.2 changed. Application Note for FAU_STG.4.1 added. Figure 1 changed.	Christian Koob
1.04	04/09/2015	Comments from certification body: - CXI-Library added in Figure 1 - Length of Challenge changed	Christian Koob
1.05	30/09/2015	Description of A.Physical Protection and OE. OE.Physical Protection changed.	Torsten Mänz
1.06	07/10/2015	Encryption key instead of encryption key pair changed in chapter 2.2.6, 2.7 and 10.1.1.2	Christian Koob
1.07	08/10/2015	In chapters 1.2, 1.2.1.2, 2.2.2, 2.2.6, 10.1.1.2, 10.1.3.1 and 10.1.4 added, that the TOE triggers cryptographic operations.	Christian Koob

1 INTRODUCTION

1.1 ST reference and TOE reference

Title:	secunet eID PKI Suite Certified CA Kernel, Version 1.0.0, Security Target
Sponsor:	secunet Security Networks AG
Editor:	Marcus Meier, Christian Koob, secunet Security Networks AG
Version:	1.07
Date:	2015/10/08
CC Version:	Version 3.1, Revision 4
Assurance Level:	EAL 4 augmented with ALC_FLR.2
General Status:	Draft
Certification-ID:	BSI-DSZ-CC-0960
Keywords:	Public Key Infrastructure, PKI, Certificate Issuing and Management Component
TOE name:	Certified CA Kernel
TOE version:	1.0.0

1.2 TOE overview

The Certified CA Kernel of secunet eID PKI Suite (abbr. Certified CA Kernel) consists of different components that are responsible for the request, issuance, revocation, and overall management of certificates and certificate status information. It supports Extended Access Control Certification Authorities (EAC CAs,) according Technical Guidline BSI TR-03110 [BSI TR-03110] and International Civil Aviation Organization CAs (ICAO CAs), which are X.509 CAs according ITU-T X.509 [ITU-T X.509]. The TOE triggers generation of X.509 certificates and CRLs.

For cryptographic operations the Certified CA Kernel rely on a FIPS 140-2 Level 3 [FIPS140-2] validated cryptographic module – a Hardware Security Module (HSM) – according to the Certificate Issuing and Management Components (CIMC) Protection Profile [CIMC PP]. All cryptographic operations (key generation, hashing, signing, verifying and key deletion) are performed within this validated cryptographic module. The HSM runs in FIPS mode. Here, FIPS mode means FIPS approved mode of operation according to [FIPS140-2].

The Certified CA Kernel provides Registration Authority (RA) functionality as well as CA functionality according the CIMC PP [CIMC PP].

The Certified CA Kernel is JAVA-based and uses the Bouncy Castle cryptography libraries.

1.2.1 TOE definition and operational usage

The Target of Evaluation (TOE) described in this Security Target is a Certificate Issuing and Management Component, called Certified CA Kernel. A Public Key Infrastructure (PKI) is a security infrastructure that creates and manages public key certificates to facilitate the use of public key cryptography. To achieve this goal, the Certified CA Kernel performs two basic tasks:

1) generates and distributes public key certificates to bind public keys to other information after validating the accuracy of the binding; and
2) maintains and distributes certificate status information for unexpired certificates.

By expanding passports through electronic and biometric data, new security relevant requirements have emerged: on the one hand, the data stored on the chip must be protected against manipulation and forgery. On the other hand,

it must be guaranteed that only authorized persons have access to the electronic data on the passport. These security requirements are implemented by different certification-instances within Certified CA Kernel: ICAO PKI and EAC PKI.

1.2.1.1 Cryptographic Modules

An external key storage is used to generate key material, to store these keys and to execute cryptographic operations with them.

CIMC PP [CIMP PP], Section 2, category (2) requires that "cryptographic operations must be performed within FIPS 140-2 validated cryptographic modules". Thus, the evaluated configuration only supports HSMs. These cryptographic modules are assumed to be part of the non-TOE environment; i. e. the HSMs are not part of the TOE (according to CIMC PP [CIMP PP], Section 2). Thus, only the interface to a HSM is part of the TOE. The supported validated FIPS 140-2 HSMs are listed in section 1.2.3. The key storage to be used (here: HSM) is determined in the appropriate CAProfile. During bootstrapping and during initial start-up of the TOE the Administrator is prompted to enter the access data of the HSM.

1.2.1.2 CRL and OCSP

As described above the CSCA (X.509-CA) of Certified CA Kernel triggers generation of complete CRLs according [RFC5280] for certificates which are no longer trustworthy and were revoked. CRL generation and distribution is part of the TOE. The CRL is provided via a distribution point that is determined in the appropriate CAProfile via user interface.

The secunet Certified CA Kernel does not act as Online Certificate Status Protocol (OCSP) responder according [RFC6960]. It only generates and transfers certificates and certificate status information to OCSP responders that are part of the TOE environment. According CIMC PP [CIMC PP], Section 2, "The features provided by repositories, OCSP servers, key recovery servers, and roaming credential servers are optional in a PKI implementation." And according [CIMC PP], Section 2: Only "Where the CA acts as an OCSP server, this service is within the boundary of the CIMC." I. e., an OCSP server is not part of the TOE.

1.2.2 TOE major security features for operational use

The Certified CA Kernel provides the following security features to cover the CIMC PP [CIMC PP] requirements:

- Security Audit (FAU) includes a chronological logging of events that occur in a system to act as a deterrent against security violations.
- Communication (FCO) involves the transport of information and enforces non-repudiation of origin and receipt as well as replay detection.
- Cryptographic Support (FCS) employs cryptographic functionality and addresses key management and the operational use of cryptographic keys. Please note that the TOE only triggers cryptographic functionality – but cryptographic functionality is always performed within FIPS 140-2 validated cryptographic modules.
- User Data Protection (FDP) relates to the protection of user data including certificate issuance, revocation, and profile management of certificates, as well as Certificate Revocation List (CRL).
- Identification and Authentication (FIA) supports the administration and enforcement of the TOE access control policies to unambiguously identify the person and/or entity performing functions in the TOE.
- Security Management (FMT) specifies several aspects of management of security functions including distinct roles to maintain the security of the TOE.
- Protection of the TOE Security Functions (FPT) supports functions that protect the integrity of TSF data from modification through the use of reliable time stamps and protected audit logs resp. audit trails.

1.2.3 Required non-TOE hardware /software/firmware

1.2.3.1 CA-Server

The CA-Server must fulfill the following minimal requirements:

- 1024 MB RAM
- 1,4 GHz CPU (64 Bit)
- 32 GB storage entity

The hardware must be compatible with the operating system (see Section 1.2.3.2). The physical connections are:

- Network Card
- Power Supply
- PS/2- or USB-attached keyboard
- VGA graphics adapter

1.2.3.2 Operating system

The certified Certified CA Kernel supports Windows Server 2012 R2 operating system. The operating system must be appropriately prepared for the operation of Certified CA Kernel. It is sufficient if the operating system was installed in the basic configuration. Particularly, the server does not need any additional services such as print servers or web servers. In addition to the base installation, the following packages must be installed:

- Oracle Java SE 8u45

1.2.3.3 Adapter

All data of Certified CA Kernel may be stored in or retrieved from a database via an Adapter. The Adapter provides the interfaces to the TOE and to the database. Certified CA Kernel and the Adapter run on the same machine (CA Server, see Figure 1).

1.2.3.4 Cryptographic Module

To be compliant with the CIMC PP [CIMP PP] the certified Certified CA Kernel supports Utimaco SafeGuard LAN CryptoServers, which at least are FIPS 140-2 Level 3 [FIPS140-2] validated cryptographic modules. TOE functional testing comprises the Utimaco HSM package version 3.11.0, which also includes the CXI library. The CXI library is not part of the TOE.

1.3 Conventions

With a few exceptions, the notation, formatting, and conventions used in this document are consistent with version 3.1 revision 4 of the CC. Specific style and clarifying information conventions were developed to aid the reader, as described below.

- Whenever an operation (assignment, selection, or refinement) has been applied to a security functional requirement, the corresponding text is underlined.
- Whenever a security functional requirement has been used more than once in the ST, the title of the security functional requirement is followed by an iteration number (e.g., iteration 1) to distinguish between the different iterations of the security functional requirement.
- Notes provide additional information about the requirement or provide clarification of the intent of the requirement (e.g., NOTE: One method of meeting the requirements of FAU_STG.1 is to write audit data directly to non-modifiable media).
- Wherever possible, the security functional requirements used in the ST were taken from CIMC PP [CIMP PP] and thus part 2 of the CC. Those functional security requirements that were not drawn from part 2 of the CC contain “CIMC” in their names in order to clearly identify them as requirements that are unique to the CIMC PP. Where a new requirement was closely related to one of the existing families of security requirements in part 2 of the CC, the new requirement name consists of that family’s name followed by CIMC (e.g., FCO_NRO_CIMC.3).

Where a new requirement was not closely related to any existing family of security requirements, the most closely related class was used as the basis for the requirement's name (e.g., FDP_CIMC_CER.1).

- Whenever a unique requirement has been specified in the document, the rationale for including this requirement is located immediately following the security functional requirement. This has been done as an alternative to including the rationale.
- The ST author added application notes in Section 6. In order to distinguish between CIMC PP [CIMP PP] application notes and the added application notes the suffix "_D" is added.

2 TOE DESCRIPTION

2.1 Physical scope of the TOE

The Certified CA Kernel consists of software and the related guidance documents; both delivered in a zip file, which is a common archive and compression standard file. The following table lists all deliverables contained in the zip file:

Table 1 Scope of TOE delivery

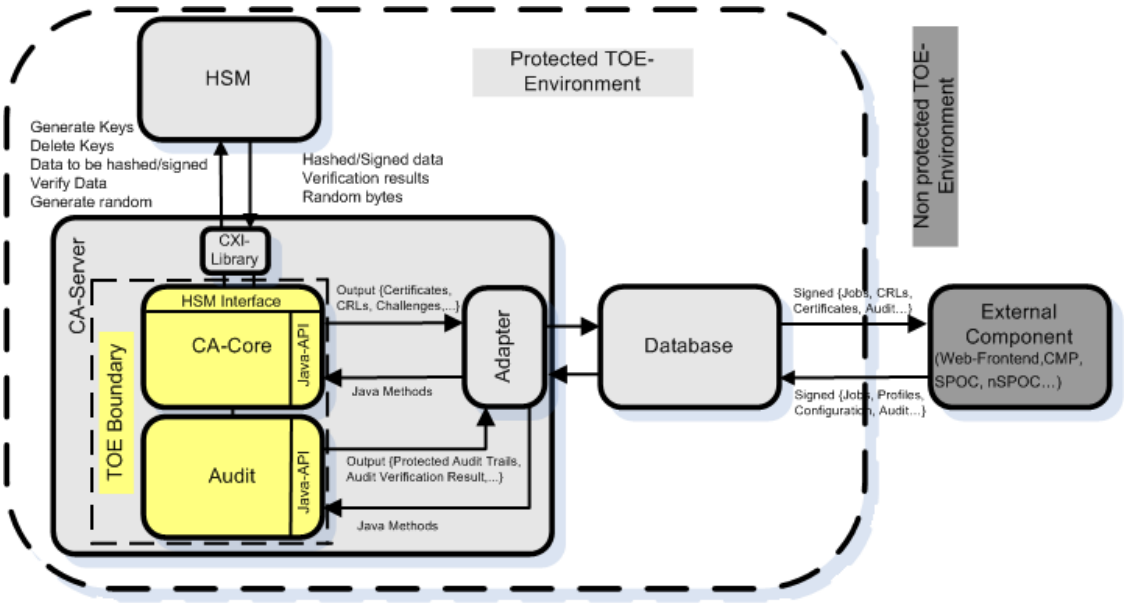
Category	Description	Version	Remarks
Software	Certified CA Kernel (zip file) that contains:	1.0.0	Delivered on CD/DVD or download via secunet Download-Portal
	CertifiedCAKernel.jar	1.0.0	JAR archive with the Certified CA Kernel functionality
	bootstrap.bat	1.0.0	Batch file with bootstrapping functionality
Guidance Documents	Manual	1.03	Certified CA Kernel Manual
	Security Target	1.07	Certified CA Kernel Security Target

The manual will describe all procedures that are necessary to maintain security when distributing versions of the TOE.

2.2 Logical scope of the TOE

From the logical point of view the Certified CA Kernel provides RA functionality to verify the information in the public key certificates and determine certificate status and CA functionality to generate certificates and certificate status information as well as audit data generation according example CIMC-3 (single component) of CIMC PP [CIMC PP]. The following figure shows the TOE boundary with the TOE components:

Figure 1: TOE boundary



2.2.1 CA-Core

The CA-Core is responsible for CA-internal tasks such as key generation and issuing certificates. The CA-Core can manage multiple Cas. The CA-Core is managed during bootstrapping with bootstrapping tool (see Section 2.7).

The CA-Core is realized as library and has a defined Java-API. The Java-API of the CA-Core has two interfaces: One is for controlling the CA-Core (Java methods) and the second is for storing CA-Core data (Output) via Adapter. The main Java methods are:

- Generate a CA
- Generate a CRL
- Generate a Certificate
- Change Profile Configuration
- Generate Challenge

By implementing an Adapter (not part of the TOE) the CA-Core and Audit can be integrated into any PKI solution. Such an Adapter may perform for example Workflow-based processes. The processes of a CA – by example certificate creation – are called Jobs.

In order to prevent replay, selected Java methods like certificate generation, CRL generation or change configuration require a challenge-response algorithm.

Thanks to the Java-API an Adapter can be implemented and adjusted to customer-specific requirements and systems.

Due to security reason the configuration to connect the HSM is predefined within the TOE and cannot be changed. Also the user roles are predefined within the TOE and cannot be changed. User roles are also encoded as X.509 extension into the user certificate.

If the CA-Core cannot reach the HSM over the network with the predefined configuration it does not startup.

2.2.1.1 CA-Core Interface to the HSM

The required HSM (see Section 1.2.3.4) is part of the protected TOE-Environment (see A.Physical Protection). The interface between CA-Core and HSM is Java Cryptography Extension (JCE). The CA-Core uses the HSM for:

- Random generation for challenge-response,

- cryptographic Key generation,
- cryptographic Key operation (hashing, signing, signature verification) and
- cryptographic Key deletion.

2.2.1.2 CA-Core Interface to the Adapter

The required Adapter (see Section 1.2.3.3) is part of the TOE-Environment (see A.Physical Protection). The interface between CA-Core and Adapter is a Java-API. The CA-Core uses the Adapter for:

- Fetching Jobs – dedicated for CA-Core – created and stored by external components (Web-Frontend, SPOC, nSPOC, CMP,…),
- Storing Job results for external components (Web-Frontend, SPOC, nSPOC, CMP,…),
- Storing and fetching the following data
    - CA configuration
    - CAProfile/CRLProfile
    - CRLs

2.2.2 Challenge Request and Response

In order to prevent replay the CA-Core triggers a challenge-response algorithm. In a first step the external component must request a challenge via Adapter from the CA-Core. The request is not cryptographic protected. The CA-Core then triggers generation of a challenge (10 Byte) within HSM. As the HSM operates in FIPS mode the HSMs Deterministic Random Number Generator (DRNG) is mandatory used to generate the challenge. The CA-Core then stores the challenge with the user identification given in the request and sends the challenge back to the external component via Adapter. Now the external component may request Job processing via Adapter in a second step. A Job must contain amongst others the requested challenge and must be signed with the user's private key.

2.2.3 User Authorization and Challenge Verification

Before CA-Core starts a particular process it performs the following checks to ensure the integrity of the consigned Java method data: The CA-Core

- performs user certificate validation and the appropriate certificate chain validation
- performs the signature verification with all consigned data
- checks whether the given challenge and the signature identity matches a stored challenge/identity and
- checks whether the role of the signature identity has the right to perform the requested process (for example creating a new certificate or a new certification revocation list).

If all checks succeed, the Audit generates an audit log record and starts request processing (see Section 2.2.4 and 2.2.5). If a check fails, the Audit generates an audit log record and the CA-Core does not start request processing.

2.2.4 Configuration Security

At the first startup the CA-Core has no configuration. Thus, the CA-Core must first be configured (e. g. CA configuration) via the Java-API. The CA-Core performs the same checks for Java configuration method as described in Section 2.2.3. That is certificate validation, signature verification, challenge/identity check and role check. If all checks succeed, the Audit generates an audit log and the CA-Core triggers the generation of a new symmetric key within HSM. Then the CA-Core triggers HMAC [RFC2104] protection of the configuration within HSM. Finally the CA-Core stores the HMAC protected configuration via Java-API to the Adapter.

In order to prevent replay every change of a configuration requires that the CA-Core triggers the generation a new symmetric key and the deletion of the formerly used symmetric key within HSM.

If a configuration is needed during processing the CA-Core loads all information via Java-API from the Adapter. Then the CA-Core triggers HMAC verification within HSM. If HMAC verification fails the Audit generates an audit log record and the CA-Core does not further continue processing. If HMAC verification succeeds the CA-Core Job processing is continued.

2.2.5 Profile Validation and Certificate/CRL generation

In case of a certificate request the CA-Core

- validates the certificate request against the loaded CAProfile,
- triggers signature verification of the certificate request within HSM,
- transforms the CAProfile and merge it with the certificate request into a certification template,
- triggers signing of certificate template to generate a certificate within HSM and
- returns the new certificate via Java-API to the Adapter.

In case of a certificate revocation list request the CA-Core

- merges the CRLProfile and the list of revoked certificates into the certificate revocation list template,
- triggers singing of the certificate revocation list template within HSM and
- returns the new certificate revocation list via Java-API to the Adapter.

2.2.6 Audit

The Audit protects audit messages against modification or deletion to ensure accountability of user actions.
The Audit logs of the security-relevant events that were performed by the TOE. These events are either triggered internally or by external components/users via Java methods. That is the CA-Core logs amongst others every event and the appropriate event state, in the case that this event triggers a process of the CA-Core.

For that the CA-Core sends the log messages to the Audit. The Audit unit of the TOE then generates uniquely identifiable audit messages, so called audit records. The TOE triggers that a set of these chronological ordered audit records (called audit trail) are periodically signed by means of a digital signature by the Hardware Security Module, resulting in a so called protected audit trail. This period is configurable. In order to protect audit messages against modification or deletion the Audit uses timestamps and audit trail sequence numbers.

The Audit also triggers some cryptographic operations with HSM to protect these messages. The Audit needs three different cryptographic keys to protect the audit trails. It needs two asymmetric key pairs, one signature key pair (ASK) and one encryption key (AEK) and also one current symmetric trail record key (TRK). All these keys are stored on the HSM. The asymmetric keys are generated during the bootstrap process of the TOE (see Section 2.7). Audit records and audit trails are stored via Java-API in the Adapter.

The CA-Core generates audit messages for the following auditable events:

- Star-up and shutdown of the audit functions and
- further security-relevant events

Each audit record contains:

- Sequence number of the trail to which the record belongs to
- Sequence number of the record inside this trail
- Creation date of this record
- Event type of the log message
- Identifier of the module who creates this entry
- Identifier of the user who issues this entry
- This element contains the log message
- MAC over this log record generated with current trail record key (TRK)

The TOE performs on every startup of the Audit an integrity check of the latest audit trail with HSM. Any error during this check shows manipulation or deletion of the audit trail storage. In this case the TOE does not startup.

2.3 CIMC Intended Environment

The Certified CA Kernel is operated in a protected environment (A.Physical Protection). The sensitivity of the information protected by the certificates issued by CIMCs will vary significantly. Users will be required to evaluate the environment and the associated threats and vulnerabilities and determine the level of risk they are willing to accept based on the sensitivity of the information.

Certified CA Kernel is appropriate for environments where risks and consequences of data disclosure and loss of data integrity are moderate to low. This ST requires integrity controls to ensure data is not modified and includes additional assurance requirements to ensure the CIMC is functioning securely.

Certified CA Kernel provides some protection against malicious authorized users by requiring, at a minimum, three distinct roles. One role – Administrator – will be responsible for account administration, key generation, and audit configuration; a second role – Officer – will be responsible for issuing and revoking certificates; and a third role – Auditor – responsible for maintaining the audit logs. Please note that the role Operator as defined in CIMC PP [CIMC] is not a distinct ST role as backup and recovery functionality is not part of the Certified CA Kernel in accordance with CIMC PP [CIMP PP], Section 5.3.

Cryptographic modules responsible for long-term private key protection or for signing certificates or certificate status information must be validated to FIPS 140-2 Level 3. Finally, there is public key protection and digital signatures are required on all messages.

Within this ST, the applicable CC assurance level is EAL 4 (methodically designed, tested and reviewed) augmented by ALC_FLR.2 (flaw reporting procedures) to further ensure that identified flaws in the product are addressed appropriately.

2.4 CIMC Keys

The TOE may manage the following keys:

- *Component keys*: Keys, other than CIMS personnel keys, which are used by the CIMC. Certified CA Kernel uses Component keys to sign certificates and certificate status information. Component keys are also used to sign audit logs and to ensure the integrity of changed Jobs by CA-Core. Component secret keys are neither used to encrypt CIMC stored or transmitted data nor to compute authentication codes.

Please note that Certified CA Kernel does not manage *Certificate subject private keys* as defined in CIMC PP [CIMC PP], Section “CIMC Keys” as the Certified CA Kernel does not provide key recovery. These keys are managed in TOE environment.

Please further note that Certified CA Kernel does not manage *CIMS personnel keys* as defined in CIMC PP [CIMC PP], Section “CIMC Keys”. These keys are managed in TOE environment. CIMS personnel keys are used to sign configuration files and Jobs created by users of external Components for authentication against CA-Core.

2.4.1 Cryptographic Functions Involving Private or Secret Keys

Private and secret keys within a CIMC are separated into different usage categories as described below. Listed in brackets next to each usage category are the associated key user categories defined in the CIMC Keys section.

- *Certificate and Status Signing Keys*: Private keys used to sign certificates, CRLs, or other statements about the status of certificates. [Component keys]
- *Integrity or Approval Authentication Keys*: Secret keys used to protect the integrity of all data stored by CIMC in the database. [Component keys]

Please note that Certified CA Kernel does not manage *General Authentication Keys* as defined in CIMC PP [CIMC PP], Section “Cryptographic Functions Involving Private or Secret Keys”. These keys are managed in TOE environment.

Please further note that Certified CA Kernel does not require any of the following keys defined in CIMC PP [CIMC PP], Section "Cryptographic Functions Involving Private or Secret Keys": *Long Term Private Key Protection Keys*, *Long Term Confidentiality Keys*, *Short Term Private Key Protection Keys*, or *Short Term Confidentiality Keys*.

2.5 Data Input

Certified CA Kernel receives information of the following category 3) as already defined in CIMC PP [CIMC PP], Section "Data Input":

1) *Unauthenticated Data Entry:* Certified CA Kernel never accepts unauthenticated data.

2) *Local Data Entry:* Certified CA Kernel does not support local data entry.

3) *Remote Data Entry:* Certified CA Kernel fetches via Adapter data from a database in such a way that it can be bound to the identity of the sender of the data (signed Jobs).

2.6 Trusted Public Key Entry, Deletion, and Storage

All "trust anchors" created during bootstrapping (see Section 2.7) are stored in the HSM. The TOE identifies trust anchors by a special prefix of the public key name. The TOE can simply use these public keys, to verify the issued certificates.

2.7 Bootstrapping

In order to protect audit data or to verify Java methods or configuration data the TOE requires a bootstrapping process to first generate the required cryptographic keys or certificates with help of HSM. The bootstrapping process is provided by the so called bootstrapping tool, installed on the CA-Server. After bootstrapping, the bootstrapping tool must be deleted from the CA-Server. During bootstrapping the following cryptographic keys or certificates are generated:

- signature key pair (ASK) for the Audit
- encryption key (AEK) for the Audit
- user key pair and user root certificate for user PKI

3 Conformance Claim

3.1 CC Conformance Claim

This Security Target claims conformance to

- Common Criteria for Information Technology Security Evaluation, Part 1:
  Introduction and general model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 2:
  Security functional components; CCMB-2012-09-002, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluation, Part 3:
  Security assurance components; CCMB-2012-09-003, Version 3.1, Revision 4, September 2012

as follows

- Part 2 extended,
- Part 3 conformant.

The

- Common Methodology for Information Technology Security Evaluation,
  Evaluation methodology; CCMB-2012-09-004, Version 3.1, Revision 4, September 2012

has to be taken into account.

3.2 PP Claim

This Security Target claims demonstrable conformance to the Protection Profile "Certificate Issuing and Management Component", Version 1.5, 11th August 2011 [CIMC PP].

3.3 Package Claim

The current Security Target is conformant to the following security requirements package: Assurance package EAL4 augmented by ALC_FLR.2

3.4 CC Conformance Claim Rationale

3.4.1 PP conformance

This Security Target is demonstrable conformant to the CIMC PP [CIMC PP]. That is, the Security Target contains all applicable statements that are in the CIMC PP [CIMC PP].

3.4.2 TOE type conformance

The TOE type in this Security Target is consistent with the TOE type in the CIMC PP [CIMC PP] as the TOE is a PKI software.

4 SECURITY PROBLEM DEFINITION

This chapter introduces the security problem definition of the TOE. This comprises:

- The assets which have to be protected by the TOE.
- The subjects/external entities which are interacting with the TOE.
- The assumptions which have to be made about the environment of the TOE.
- The threats which exist against the assets of the TOE
- The organizational security policies the TOE has to comply to.

This information provides the basis for the Security Objectives specified in Section 5, the security functional requirements for the TOE specified in Section 6, and the TOE Security Assurance Requirements specified in Section 7.

4.1 Assets

The following assets need to be protected by the TOE and its environment:

Audit records	The TOE generates uniquely identifiable audit messages (so called audit records) that need to be protected against modification.
CRLs	The TOE generates so called Certificate Revocation Lists. These lists contain information about revoked certificates. These lists need to be protected against modification.
TSF data	All configuration files (CA configuration, CAProfile and CRLProfile) are created/edited with external components and protected against modification. Only unmodified configuration files must be processed by the TOE.
Requests	E. g. certificate or certificate revocation list requests. All requests are created/edited with external components and protected against modification. Only unmodified requests must be processed by the TOE.
Cryptographic Keys	Cryptographic Keys are only generated within the HSM. The TOE triggers key generation, key usage and key deletion within the HSM. Private and symmetric keys are only stored within the HSM. These keys need to be protected by the HSM – hence the TOE environment – against modification spy out.

4.2 Subjects/External Entities

The following subjects/external entities may interact with the TOE:

Administrator	Role authorized to install, configure and maintain the TOE, establish and maintain user accounts, configure profiles, access rights and audit parameters and manage component keys.
Auditor	Role authorized to view and maintain audit logs.
Officer	Role authorized to request or approve certificates or certificate revocations.
Attacker	Any entity (human or IT) outside the TOE that interacts (or may inter-act) with the TOE – also called hacker. A goal of an attacker may be to change configuration files or send unauthorized requests. Attackers with an Enhanced-Basic attack potential are assumed.

4.3 Secure Usage Assumptions

The usage assumptions are organized in three categories: personnel (assumptions about administrators and users of the system as well as any threat agents), physical (assumptions about the physical location of the TOE or any attached peripheral devices), and connectivity (assumptions about other IT systems that are necessary for the secure operation of the TOE).

Personnel

A.Auditors Review Audit Logs	Audit logs are required for security-relevant events and must be reviewed by the Auditors.
A.Authentication Data Management	An authentication data management policy is enforced to ensure that users change their authentication data at appropriate intervals and to appropriate values (e.g., proper lengths, histories, variations, etc.) (Note: this assumption is not applicable to biometric authentication data.)
A.Competent Administrators, Officers and Auditors	Competent Administrators, Officers and Auditors will be assigned to manage the TOE and the security of the information it contains. It is further assumed that these people are non-hostile.
A.Cooperative Users	Users need to accomplish some task or group of tasks that require a secure IT environment. The users require access to at least some of the information managed by the TOE and are expected to act in a cooperative manner.
A.CPS	All Administrators, Officers, and Auditors are familiar with the certificate policy (CP) and certification practices statement (CPS) under which the TOE is operated.
A.Disposal of Authentication Data	Proper disposal of authentication data and associated privileges is performed after access has been removed (e.g., Job termination, change in responsibility).
A.Malicious Code Not Signed	Malicious code destined for the TOE is not signed by a trusted entity.
A.Notify Authorities of Security Issues	Administrators, Officers, Auditors, and other users notify proper authorities of any security issues that impact their systems to minimize the potential for the loss or compromise of data.
A.Social Engineering Training	General users, administrators, officers and auditors are trained in techniques to thwart social engineering attacks.

Connectivity

A.HSM	The HSM must only be used exclusively by the TOE. That is no other IT component is allowed to use the HSM.
A.Operating System	The operating system has been selected to provide the functions required by this CIMC to counter the perceived threats identified in this ST.

Although this ST does not specifically address the operating system, functions/requirements traditionally attributed to an operating system are distributed throughout this ST in appropriate sections. PKIs incorporating CIMC components that rely on operating systems to provide/enforce these functions/requirements must utilize operating systems with features that counter the perceived threats identified in this ST.

Physical

A.Communications	The system is adequately physically protected against loss of communications i. e.,

Protection	availability of communications.
A.Physical Protection	The TOE and non TOE hardware, software, and firmware critical to security policy enforcement will be protected from unauthorized physical modification.

4.4 Threats

The threats are organized in four categories: authorized users, system, cryptography, and external attacks.

Authorized Users
Threat agent for the following threats is an authorized user. Asset that can be compromised are the CIMC and/or the systems that rely on the PKI objects such as certificates or CRLs. The latter systems are termed relying party systems.

T.Administrative errors of omission	Administrators, Officers or Auditors fail to perform some function essential to security.
T.Administrators, Officers and Auditors commit errors or hostile actions	An Administrator, Officer or Auditor commits errors that change the intended security policy of the system or application or maliciously modify the system's configuration to allow security violations to occur.
T.User abuses authorization to collect and/or send data	User abuses granted authorizations to improperly collect and/or send sensitive or security-critical data.
T.User error makes data inaccessible	User accidentally deletes user data rendering user data inaccessible.

System

T.Critical system component fails	Failure of one or more system components results in the loss of system critical functionality. Threat agent in this case is the CIMC hardware. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Flawed code	A system or applications developer delivers code that does not perform according to specifications or contains security flaws. Threat agent in this case is the TOE developer. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Malicious code exploitation	An authorized user, IT system, or hacker downloads and executes malicious code, which causes abnormal processes that violate the integrity, availability, or confidentiality of the system assets. Threat agent could be an authorized user, TOE itself, or an unauthorized user. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Message content modification	A hacker modifies information that is intercepted from a communications link between two unsuspecting entities before passing it on to the intended recipient. Threat agent is an unauthorized user. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.

Cryptography

T.Disclosure of private and secret keys	A private or secret key is improperly disclosed. Threat agent is the authorized user or erroneous protocol. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.

T.Modification of private/secret keys	A secret/private key is modified. Threat agent is the authorized user or erroneous protocol. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Sender denies sending information	The sender of a message denies sending the message to avoid accountability for sending the message and for subsequent action or inaction. Threat agent is a subscriber to CIMC. Adverse action can be reduced trust in CIMC.

External Attacks

T.Hacker gains access	A hacker masquerades as an authorized user to perform operations that will be attributed to the authorized user or a system process or gains undetected access to a system due to missing, weak and/or incorrectly implemented access control causing potential violations of integrity, confidentiality, or availability. Threat agent is the unauthorized user. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Hacker physical access	A hacker physically interacts with the system to exploit vulnerabilities in the physical environment, resulting in arbitrary security compromises. Threat agent is the unauthorized user. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.
T.Social engineering	A hacker uses social engineering techniques to gain information about system entry, system use, system design, or system operation. Threat agent is the unauthorized user. Adverse action can be compromise of the security of the CIMC and/or relying party systems that rely on the PKI objects such as certificates or CRLs.

4.5 Organization Security Policies

P.Authorized use of information	Information shall be used only for its authorized purpose(s).
P.Cryptography	FIPS-approved or NIST-recommended cryptographic functions shall be used to perform all cryptographic operations.

5 SECURITY OBJECTIVES

This section includes the security objectives for the TOE and the security objectives for the environment as listed in CIMC PP [CIMC PP], Sections 4.1 and 4.2. Please note that the security objectives listed in CIMC PP [CIMC PP], Section 4.3 “Security Objectives for both the TOE and the Environment”, are either assigned to “Security Objectives for the TOE”, see Section 5.1 or to “Security Objectives for the Environment”, see Section 5.2. Please further note that all security objectives for the environment are renamed with the suffix “OE.” In order to distinguish between both types of security objectives.

The CIMC PP [CIMC PP] covers all kinds of architectures for a PKI and does not require a particular architecture. Thus, for Certified CA Kernel architecture all CIMC PP [CIMC PP] Security Functional Requirements are checked for application. Table 2 shows which CIMC PP [CIMC PP] Security Functional Requirements are stated in this Security Target. See also description below Table 2 for further details.

The following CIMC PP [CIMC PP] Security Objectives are not stated in the Security Target because the appropriate Security Functional Requirements are not applicable at all for the TOE:

- O.Data import/export (FDP_ETC_CIMC.5 Extended user private and secret key export, FMT_MTD_CIMC.7 Extended TSF private and secret key export, FPT_ITC.1 Inter-TSF confidentiality during transmission (iterations 1 and 2) and FDP_UCT.1 Basic data exchange confidentiality (iterations 1 and 2))
- O.Protect user and TSF data during internal transfer (FDP_ITT.1 Basic internal transfer protection (iterations 3 and 4) and FPT_ITT.1 Basic internal TSF data transfer protection (iterations 3 and 4))

5.1 Security Objectives for the TOE

This section includes the security objectives for the TOE, divided among six categories: authorized users, system, cryptography, external attacks, management and audits.

Authorized Users

O.Certificates	The TSF must ensure that certificates, certificate revocation lists, and certificate status information are valid.
O.Individual accountability and audit records	Provide individual accountability for audited events. Record in audit records: date and time of action and the entity responsible for the action.
O.Limitation of administrative access	Design administrative functions so that Administrators, Officers and Auditors do not automatically have access to user objects, except for necessary exceptions. Control access to the system by Administrators who troubleshoot the system and perform system updates.
O.Maintain user attributes	Maintain a set of security attributes (which may include role membership, access privileges, etc.) associated with individual users. This is in addition to user identity.
O.Restrict actions before authentication	Restrict the actions a user may perform before the TOE authenticates the identity of the user.
O.Security roles	Maintain security-relevant roles and the association of users with those roles.
O.User authorization management	Manage and update user authorization and privilege data to ensure they are consistent with organizational security and personnel policies.

System

O.React to detected	Implement automated notification (or other responses) to the TSF-discovered

attacks	attacks in an effort to identify attacks and to create an attack deterrent.

Cryptography

O.Non-repudiation	Prevent user from avoiding accountability for sending a message by providing evidence that the user sent the message.
O.Integrity protection of user data	Provide appropriate integrity protection for user data.

External Attacks

O.Control unknown source communication traffic	Control (e.g., reroute or discard) communication traffic from an unknown source to prevent potential damage.

Management

O.Manage behavior of security functions	Provide management functions to configure, operate, and maintain the security mechanisms.
O.Configuration Management	Implement a configuration management plan. Implement configuration management to assure identification of system connectivity (software, hardware, and firmware), and components (software, hardware, and firmware), auditing of configuration data, and controlling changes to configuration items.

Audit

O.Protect stored audit records	Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.
O.Respond to possible loss of stored audit records	Respond to possible loss of audit records when audit trail storage is full or nearly full by restricting auditable events.

5.2 Security Objectives for the Environment

This section specifies the security objectives for the environment.

OE.Administrators, Officers and Auditors guidance documentation	Deter Administrator, Officer or Auditor errors by providing adequate documentation on securely configuring and operating the CIMC.
OE.Auditors Review Audit Logs	Identify and monitor security-relevant events by requiring auditors to review audit logs on a frequency sufficient to address level of risk.
OE.Authentication Data Management	Ensure that users change their authentication data at appropriate intervals and to appropriate values (e.g., proper lengths, histories, variations, etc.) through enforced authentication data management (Note: this objective is not applicable to biometric authentication data.)
OE.Communications Protection	Protect the system against a physical attack on the communications capability by providing adequate physical security.

OE.Competent Administrators, Officers and Auditors	Provide capable management of the TOE by assigning competent Administrators, Officers and Auditors to manage the TOE and the security of the information it contains. Only non-hostile people are entrusted with administrative tasks.
OE.Cooperative Users	Ensure that users are cooperative so that they can accomplish some task or group of tasks that require a secure IT environment and information managed by the TOE.
OE.CPS	All Administrators, Officers and Auditors shall be familiar with the certificate policy (CP) and the certification practices statement (CPS) under which the TOE is operated.
OE.Detect modifications of firmware, software, and backup data	Provide integrity protection to detect modifications to firmware, software, and backup data.
OE.Disposal of Authentication Data	Provide proper disposal of authentication data and associated privileges after access has been removed (e.g., Job termination, change in responsibility).
OE.HSM	The HSM in FIPS mode enforces usage of smartcards. Thus all Administrators, Officers and Auditor must only use smartcards as authentication token between them and the HSM via CXI library.
OE.Installation	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which maintains IT security.
OE.Lifecycle security	Provide tools and techniques used during the development phase to ensure security is designed into the CIMC. Detect and resolve flaws during the operational phase.
OE.Malicious Code Not Signed	Protect the TOE from malicious code by ensuring all code is signed by a trusted entity prior to loading it into the system.
OE.Notify Authorities of Security Issues	Notify proper authorities of any security issues that impact their systems to minimize the potential for the loss or compromise of data.
OE.Object and data recovery free from malicious code	Recover to a viable state after malicious code is introduced and damage occurs. That state must be free from the original malicious code.
OE.Operating System	The operating system used is validated to provide adequate security, including domain separation and non-bypassability, in accordance with security requirements recommended by the National Institute of Standards and Technology.
OE.Periodically check integrity	Provide periodic integrity checks on both system and software.
OE.Physical Protection	Those responsible for the TOE must ensure that the security-relevant components of the TOE and non TOE are protected from physical attack that might compromise IT security.
OE.Preservation/trusted recovery of secure state	Preserve the secure state of the system in the event of a secure component failure and/or recover to a secure state.
OE.Procedures for preventing malicious code	Incorporate malicious code prevention procedures and mechanisms.
OE.Repair identified security flaws	The vendor repairs security flaws that have been identified by a user.
OE.Require inspection for downloads	Require inspection of downloads/transfers.
OE.Security-relevant configuration management	Manage and update system security policy data and enforcement functions, and other security-relevant configuration data, to ensure they are consistent with organizational security policies.
OE.Social Engineering Training	Provide training for general users, Administrators, Officers and Auditors in techniques to thwart social engineering attacks.
OE.Sufficient backup storage and effective restoration	Provide sufficient backup storage and effective restoration to ensure that the system can be recreated.
OE.Time stamps	Provide time stamps to ensure that the sequencing of events can be verified.

OE.Trusted Path	Provide a trusted path between the user and the system. Provide a trusted path to security-relevant (TSF) data in which both end points have assured identities.
OE.Validation of security function	Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.
OE.Cryptographic functions	Provide approved cryptographic algorithms for authentication and signature generation/verification; approved key generation techniques and use validated cryptographic modules in the TOE environment. (Validated is defined as FIPS 140-2 validated.). The cryptographic module is required to run in FIPS mode.

6 TOE SECURITY FUNCTIONAL REQUIREMENTS

This section specifies the security requirements that are applicable to CIMC functionality, such as key management, certificate registration, and CIMC configuration and management functions.

Table 2 lists all the functional security requirements for the TOE that are included in this ST. They are listed in alphabetical order in Table 2 for ease of reference. Also included are the applicable CIMC ST sections to which each requirement applies.

Please note that the functional security requirement suffix "iteration" is taken exactly from the CIMC PP [CIMC PP], Section 6.

Table 2 CIMC TOE Functional Security Requirements

Security Functional Components	**CIMC ST Section**
FAU_GEN.1 Audit data generation (iteration 2)	6.1 Security Audit
FAU_GEN.2 User identity association (iteration 2)	6.1 Security Audit
FAU_SEL.1 Selective audit (iteration 2)	6.1 Security Audit
FAU_STG.1 Protected audit trail storage (iteration 2)	6.1 Security Audit
FAU_STG.4 Prevention of audit data loss (iteration 2)	6.1 Security Audit
FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	6.5 Remote Data Entry and Export
FCO_NRO_CIMC.4 Advanced verification of origin	6.5 Remote Data Entry and Export
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	6.4.4 Private and Secret Key Destruction
FDP_ACC.1 Subset access control (iteration 2)	6.3 Access Control
FDP_ACF.1 Security attribute based access control (iteration 2)	6.3 Access Control
FDP_CIMC_CER.1 Certificate Generation	6.10 Certificate Registration
FDP_CIMC_CRL.1 Certificate revocation list validation	6.11.1 Certificate Revocation List Validation
FDP_CIMC_CSE.1 Certificate status export	6.5.1 Certificate Status Export
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	6.6.2 Public Key Storage
FIA_ATD.1 User attribute definition	6.2 Roles
FIA_SOS.1 Verification of secrets (iteration 2)	6.4 Identification and Authentication
FIA_UAU.1 Timing of authentication (iteration 2)	6.4 Identification and Authentication
FIA_UID.1 Timing of identification (iteration 2)	6.4 Identification and Authentication
FIA_USB.1 User-subject binding (iteration 2)	6.4 Identification and Authentication
FMT_MOF.1 Management of security functions behavior (iteration 2)	6.2 Roles
FMT_MOF_CIMC.3 Extended certificate profile management	6.7 Certificate Profile Management
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	6.8 Certificate Revocation List Profile Management
FMT_MTD.1 Management of TSF data	6.2 Roles
FMT_MSA.1 Management of security attributes	6.2 Roles

FMT_SMR.1 Security roles	6.2 Roles
FPT_CIMC_TSP.1 Audit log signing event	6.1 Security Audit

Due to the chosen TOE design The following CIMC PP [CIMC PP] functional security requirements are not applicable and thus not included in this ST:

- Since the TOE provides CRL functionality but not OCSP functionality – consistently with CIMC PP [CIMC PP] – FDP_CIMC_OCSP.1 is not applicable.
- Since the TOE does not provide Basic Response Validation according FDP_CIMC_OCSP.1 the functional requirement FMT_MOF_CIMC.6 is not applicable.
- Since there are no physically-separated parts of the TOE, there is no need to employ specific security mechanisms according FDP_ITT.1 (iterations 3 and 4).
- Since the TOE does not transmit confidential objects, there is no need to employ specific security mechanisms according FDP_UCT.1 (iteration 2).
- Since the TOE does not transmit confidential TSF data, there is no need to employ specific security mechanisms according FPT_ITC.1 (iteration 2).
- Since the TOE does neither manage CIMS personal keys nor certificate subject private keys FDP_ACF_CIMC.2 is not applicable.
- Since the Component private keys are stored only on the HSM FMT_MTD_CIMC.4 is not applicable.
- Since the TOE does not manage user secret keys FDP_ACF_CIMC.3 is not applicable.
- Since Component private keys are only stored on the HSM FMT_MTD_CIMC.5 is not applicable.
- Since the TOE does neither export user private keys nor TSF private keys FDP_ETC_CIMC.5 and FMT_MTD_CIMC.7 are not applicable.
- Since there are no physically-separated parts of the TOE, there is no need to employ specific security mechanisms according FPT_ITT.1 (iterations 3 and 4).
- Since reliable time stamps are provided by the TOE environment (see OE.Time stamps) there is no need to employ specific security mechanism according FPT_STM.1 Reliable time stamps (iterations 2).

6.1 Security Audit

Audit includes a chronological recording of events that occur in a system. The objective is to track what occurs to enable the reconstruction and examination of a sequence of events and/or changes in an event. This is useful in ensuring that the system is operated securely and in providing evidence when a suspected or actual security compromise has occurred. Audit also provides for reconstructing a specific state of a system. The objective in a PKI system is to enable an appropriate authority to determine whether a signature should have been accepted as valid.

The audit will be used to reconstruct important events that were performed by the TOE, such as issuance of a CA certificate, and the user or event (e.g., a signed certificate request) that caused them. The audit will be used to arbitrate future disputes by establishing the validity of a signature at a particular time. The audit log records the security-relevant events that were performed by the TOE and the users or events (e.g., a signed certificate request) that caused them. This subsection specifies the security requirements for maintaining and protecting the integrity of the audit logs.

The TOE maintains a single audit log in the database.

FAU_GEN.1 Audit data generation (iteration 2)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:
a) Start-up and shutdown of the audit functions;
b) All auditable events for the minimum level of audit; and
c) The events listed in Table 3 below.

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, the information specified in the Additional Details column in Table 3 below.

Additionally, the audit shall not include plaintext private or secret keys or other critical security parameters.

Table 3 Auditable Events and Audit Data

Section/Function	Component	Event	Additional Details
6.1; Security Audit	FAU_GEN.1 Audit data generation (iteration 2)	Any changes to the audit parameters, e.g., audit frequency, type of event audited Any attempt to delete the audit log	
	FPT_CIMC_TSP.1 Audit log signing event	Audit log signing event	Digital signature, keyed hash, or authentication code shall be included in the audit log.
Local Data Entry		All security-relevant data that is entered in the system	Application Note_D: Not applicable, since the TOE does not support local data entry. The identity of the data entry individual if the entered data is linked to any other data (e.g., clicking an "accept" button). This shall be included with the accepted data.
Remote Data Entry		All security-relevant messages that are received by the system	
Data Export and Output		All successful and unsuccessful requests for confidential and security-relevant information	Application Note_D: Not applicable, since the TOE does not provide confidential or security-relevant information
Key Generation	FCS_CKM.1 Cryptographic Key Generation	Whenever the TSF requests generation of a cryptographic key. (Not mandatory for single session or one-time use symmetric keys.)	The public component of any asymmetric key pair generated
Private Key Load		The loading of Component private keys	Application Note_D: Not applicable, since component private keys are generated and stored on the HSM.

6.6.1: Private Key Storage		All access to certificate subject private keys retained within the TOE for key recovery purposes	Application Note_D: Not applicable, since the TOE does not manage certificate subject private keys
Trusted Public Key Entry, Deletion and Storage		All changes to the trusted public keys, including additions and deletions	The public key and all information associated with the key
6.6.3: Secret Key Storage		The manual entry of secret keys used for authentication	Application Note_D: Not applicable, since the TOE does use Secret Key Storage
6.6.5: Private and Secret Key Export	FDP_ETC_CIMC.5 Extended user private and secret key export;	The export of private and secret keys (keys used for a single session or message are excluded)	Application Note_D: Not applicable, since the TOE does not export any private or secret keys.
	FMT_MTD_CIMC.7 Extended TSF private and secret key export		Application Note_D: Not applicable, since the TOE does not export any private or secret keys.
6.10: Certificate Registration	FDP_CIMC_CER.1 Certificate Generation	All certificate requests	If accepted, a copy of the certificate. If rejected, the reason for rejection (e.g., invalid data, request rejected by Officer, etc.).
Certificate Status Change Approval		All requests to change the status of a certificate	Whether the request was Accepted or rejected.
CIMC Configuration		Any security-relevant changes to the configuration of the TSF.	
6.7: Certificate Profile Management	FMT_MOF_CIMC.3 Extended certificate profile management	All changes to the certificate Profile	The changes made to the Profile
Revocation Profile Management		All changes to the revocation profile	The changes made to the Profile
6.8: Certificate Revocation List Profile Management	FMT_MOF_CIMC.5 Extended certificate revocation list profile management	All changes to the certificate revocation list profile	The changes made to the profile
6.9: Online Certificate Status Protocol (OCSP) Profile Management	FMT_MOF_CIMC.6 OCSP Profile Management	All changes to the OCSP profile	The changes made to the Profile Application Note_D: Not applicable, since the TOE does not provide OCSP functionality.

FAU_GEN.2 User identity association (iteration 2)

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SEL.1 Selective audit (iteration 2)

FAU_SEL.1.1 The TSF shall be able to select the set of events to be audited from the set of all auditable events based on the following attributes:
a) [*object identity, user identity, event type*]
b) [*no list of additional attributes that audit selectivity is based upon*].

Application Note: For FAU_SEL.1.1a, the ST author should select whether the security attributes upon which audit selectivity is based, is related to object identity, user identity, subject identity, host identity, or event type. For FAU_SEL.1.1b, the ST author should specify any additional attributes upon which audit selectivity is based.
Application Note_D: Done

FAU_STG.1 Protected audit trail storage (iteration 2)

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2 The TSF shall be able to detect unauthorized modifications to the stored audit records in the audit trail.

NOTE: One method of meeting the requirements of FAU_STG.1 is to write audit data directly to non-modifiable media.

Application Note_D: The audit trail is protected by cryptographic means.

FAU_STG.4 Prevention of audit data loss (iteration 2)

FAU_STG.4.1 The TSF shall prevent auditable events, except those taken by the Auditor if the audit trail is full.

Application Note_D: If the audit trail is full the TOE shutdowns. In that case no role (even the Auditor) is able to perform any actions apart from starting the TOE again. Thus neither role (even the Auditor) may further trigger auditable events. Thus, the administrator first has to free up memory on the trail storage (not part of the TOE).

Otherwise the TOE will not start up again.

FPT_CIMC_TSP.1 Audit log signing event

FPT_CIMC_TSP.1.1 The TSF shall periodically create an audit log signing event in which it computes a digital signature, keyed hash, or authentication code over the entries in the audit log.

FPT_CIMC_TSP.1.2 The digital signature, keyed hash, or authentication code shall be computed over, at least, every entry that has been added to the audit log since the previous audit log signing event and the digital signature, keyed hash, or authentication code from the previous audit log signed event.

FPT_CIMC_TSP.1.3 The specified frequency at which the audit log signing event occurs shall be configurable.

FPT_CIMC_TSP.1.4 The digital signature, keyed hash, or authentication code from the audit log signing event shall be included in the audit log.

Dependencies: FAU_GEN.1 Audit data generation
FMT_MOF.1 Management of security functions behavior

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by existing CC requirements. It supports the security objective O.Protect stored audit records, by providing additional protection for stored audit records.

6.2 Roles

The ability to perform many of the functions specified in this ST will be allocated to distinct roles to maintain the security of the TOE.

FIA_ATD.1 User attribute definition

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [*role*].

FMT_MOF.1 Management of security functions behavior (iteration 2)

FMT_MOF.1.1 The TSF shall restrict the ability to modify the behavior of the functions listed in Table 4 to the authorized roles as specified in Table 4.

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the [*the CIMC TOE Access Control Policy specified in section 9.1*] to restrict the ability to [modify, [*no other operations*]] the security attributes [*role*] to [*Administrators*].

FMT_MTD.1 Management of TSF data

FMT_MTD.1.1 The TSF shall restrict the ability to [modify, [*no other operations*]] the [*Configurations and Profiles*] to [*Administrators*].

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles [*Administrators, Officers and Auditors*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Table 4 Authorized Roles for Management of Security Functions Behavior

Section/Function	Component Function	Authorized Role
6.1: Security Audit		The capability to configure the audit parameters shall be restricted to Administrators. The capability to change the frequency of the audit log signing event shall be restricted to Administrators.
6.11: Certificate Registration		The capability to approve fields or extensions to be included in a certificate shall be restricted to Officers.

		If an automated process is used to approve fields or extensions to be included in a certificate, the capability to configure that process shall be restricted to Officers.
Data Export and Output		The export of CIMC private keys shall require the authorization of at least two Administrators or one Administrator and one Officer or Auditor. Application Note_D: Not applicable, since the TOE does not export CIMC private keys.
Certificate Status Change Approval		Only Officers shall configure the automated process used to approve the revocation of a certificate or information about the revocation of a certificate. Only Officers shall configure the automated process used to approve the placing of a certificate on hold or information about the on hold status of a certificate.
CIMC Configuration		The capability to configure any TSF functionality shall be restricted to Administrators. (This requirement applies to all configuration parameters unless the ability to configure that aspect of the TSF functionality has been assigned to a different role elsewhere in this document.)
6.8: Certificate Profile Management	FMT_MOF_CIMC.3 Extended certificate profile management	The capability to modify the certificate profile shall be restricted to Administrators.
Revocation Profile Management		The capability to modify the revocation profile shall be restricted to Administrators.
6.9: Certificate Revocation List Profile Management	FMT_MOF_CIMC.5 Extended certificate revocation list profile management	The capability to modify the certificate revocation list profile shall be restricted to Administrators.
Online Certificate Status Protocol (OCSP) Profile	FMT_MOF_CIMC.6 OCSP profile management	The capability to modify the OCSP profile shall be restricted to Administrators. Application Note_D: Not applicable, since the TOE does not provide OCSP functionality.

6.3 Access Control

FDP_ACC.1 Subset access control (iteration 2)

FDP_ACC.1.1 The TSF shall enforce the CIMC TOE Access Control Policy specified in section 9.1 on [*subjects: Certificate Request, Certificate Revocation List Request; objects: certificates, CRLs and operations: generate; among subjects and objects covered by the SFP*].

Application Note: The terms object and subject refer to generic elements in the TSF. For a policy to be implemented, these entities must be clearly identified. For most systems there is only one type of subject, usually called a process or task, which needs to be specified in the ST. For a PP, the objects and operations might be expressed as types such as: named objects, data repositories, observe accesses, etc. The ST author should specify the list of subjects, objects, and operations among subjects and objects covered by the SFP.

Application Note_D: Done

FDP_ACF.1 Security attribute based access control (iteration 2)

FDP_ACF.1.1 The TSF shall enforce the CIMC TOE Access Control Policy specified in section 9.1 to objects based on the following: the identity of the subject and the role that the subject is authorized to assume.

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: specified in Table 5.

FDP_ACF.1.3 The TSF shall explicitly 36uthorize access of subjects to objects based on the following additional rules: [*no additional rules*].

Application Note: The rules that govern the CIMC TOE Access Control Policy may vary between TOEs; those rules need to be specified in the ST. The ST must list the attributes that are used for access decisions. These attributes may include permission bits, access control lists, and object ownership. The ST author should specify the rules, based on security attributes, that explicitly **authorize** access of subjects to objects. These rules are in addition to those specified in FDP_ACF.1.1. They are included in FDP_ACF.1.3 as they are intended to contain exceptions to the rules in FDP_ACF.1.1.
Application Note_D: No additional rules defined.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [*no additional rules*].

Application Note: The rules that govern the CIMC TOE Access Control Policy may vary between TOEs; those rules need to be specified in the ST. The ST must list the attributes that are used for access decisions. These attributes may include permission bits, access control lists, and object ownership. The ST author should specify the rules, based on security attributes, that explicitly **deny** access of subjects to objects. These rules are in addition to those specified in FDP_ACF.1.1. They are included in FDP_ACF.1.4 as they are intended to contain exceptions to the rules in FDP_ACF.1.1.
Application Note_D: No additional rules defined.

Table 5 Access Controls

Section/Function	Event
Certificate Request Remote and Local Data Entry	The entry of certificate request data shall be restricted to Officers and the subject of the requested certificate.
Certificate Revocation Request Remote and Local Data Entry	The entry of certificate revocation request data shall be restricted to Officers and the subject of the certificate to be revoked.
Data Export and Output	The export or output of confidential and security-relevant data shall only be at the request of authorized users. Application Note_D: Not applicable, since the TOE does not export or output confidential or security-relevant data.
Key Generation	The capability to request the generation of Component keys (used to protect data in more than a single session or message) shall be restricted to Administrators. Application Note_D: Not applicable, since the TOE generates and stores Components keys on the HSM.

Private Key Load	The capability to request the loading of Component private keys into cryptographic modules shall be restricted to Administrators. Application Note_D: Not applicable, since the TOE does not load Component private keys.
6.7.1: Private Key Storage	The capability to request the decryption of certificate subject private keys shall be restricted to Officers. The TSF shall not provide a capability to decrypt certificate subject private keys that may be used to generate digital signatures. At least two Officers or one Officer and an Administrator or Auditor shall be required to request the decryption of a certificate subject private key. Application Note_D: Not applicable, since the TOE does not manage certificate subject private keys.
Trusted Public Key Entry, Deletion, and Storage	The capability to change (add, revise, delete) the trusted public keys shall be restricted to Administrators.
6.7.3: Secret Key Storage	The capability to request the loading of CIMC secret keys into cryptographic modules shall be restricted to Administrators. Application Note_D: Not applicable, since the TOE generates and stores CIMC secret keys in a HSM.
6.7.4: Private and Secret Key Destruction	The capability to zeroize CIMC plaintext private and secret keys shall be restricted to Administrators, Auditors and Officers.
6.7.5: Private and Secret Key Export	The capability to export a component private key shall be restricted to Administrators. The capability to export certificate subject private keys shall be restricted to Officers. The export of a certificate subject private key shall require the authorization of at least two Officers or one Officer and an Administrator or Auditor. Application Note_D: Not applicable, since the TOE does neither export component private keys nor nor subject private keys.

Certificate Status Change Approval[1]	Only Officers and the subject of the certificate shall be capable of requesting that a certificate be placed on hold. Only Officers shall be capable of removing a certificate from on hold status. Only Officers shall be capable of approving the placing of a certificate on hold. Only Officers and the subject of the certificate shall be capable of requesting the revocation of a certificate. Only Officers shall be capable of approving the revocation of a certificate and all information about the revocation of a certificate.

6.4 Identification and authentication

Identification and authentication includes recognizing an entity (e.g., user, device, or system) and verifying the identity of that entity.

FIA_SOS.1 Verification of secrets (iteration 2)

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [

1) For each attempt to use the authentication mechanism, the probability shall be less than one in 1,000,000 that a random attempt will succeed or a false acceptance will occur (e.g., guessing a password or PIN, false acceptance error rate of a biometric device, or some combination of authentication methods.) and
2) For multiple attempts to use the authentication mechanism during a one-minute period, the probability shall be less than one in 100,000 that a random attempt will succeed or a false acceptance will occur].

FIA_UAU.1 Timing of authentication (iteration 2)

FIA_UAU.1.1 The TSF shall allow [*requesting TOE state and requesting a challenge*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UID.1 Timing of identification (iteration 2)

FIA_UID.1.1 The TSF shall allow [*requesting TOE state and requesting a challenge*] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

[1] Every request to change certificate status, for example, revoke a certificate, place a certificate on hold, or remove a certificate from hold must be accepted or rejected. If a request is accepted, any information about the request that may be exported from the TOE must be approved. Approval may be manual or automated.

Application Note: FIA_UAU.1 and FIA_UID.1 allow the ST author to specify TSF-mediated actions that may be performed on behalf of a user before that user is identified and/or authenticated. However, the TSF shall not perform any security-relevant functions or export/output any confidential information on behalf of a user before that user has been identified or authenticated. Examples of TSF-mediated actions that may be performed on behalf of a user before that user is identified and/or authenticated include:

16) Responding to a request for public information (e.g., responding to an Online Certificate Status Protocol (OCSP) request).

b) Accepting data from a user that will not be processed until an (identified and authenticated) authorized user has accepted the data (e.g., a unauthenticated user may submit a certificate request message so long as the certificate is not generated until after an Officer has approved the request).

Application Note_D: Before a user is identified and authenticated no TSF-mediated actions that are security relevant are allowed.

FIA_USB.1 User-subject binding (iteration 2)

FIA_USB.1.1 The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [*the role that a user is authorized to assume*].

FIA_USB.1.2 The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [*A subject shall not have more rights than the associated role*].

FIA_USB.1.3 The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [*none*].

6.5 Remote Data Entry and Export

This section covers cases in which data is to be associated with a user who is not acting locally. In most cases, this will involve data that has been received in a message that has been signed or that contains an authentication code or keyed hash allowing the source of the message to be determined (in which case the data may be associated with the source of the message). Data received over a secure communication channel (e.g., SSL) could be treated similarly.

The security requirements of remote data entry apply whenever data has been received from a remote source that is considered reliable (i.e., the source of the information can be determined).

This section also specifies security requirements associated with the export of data from TOEs. The data may be distributed to a device that is outside the boundary of a TOE (either locally or remotely). The remote device or computer may not be directly connected to the TOE. Data export also applies when data is sent between physically distributed subcomponents of a TOE (e.g., data sent between a CA and RA) and the data is transmitted over an untrusted network.

FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin

FCO_NRO_CIMC.3.1 The TSF shall enforce the generation of evidence of origin for certificate status information and all other security-relevant information at all times.

FCO_NRO_CIMC.3.2 The TSF shall be able to relate the identity and [*no other attributes*] of the originator of the information, and the security-relevant portions of the information to which the evidence applies.

FCO_NRO_CIMC.3.3 The TSF shall verify the evidence of origin of information for all security-relevant information.

Dependencies: FIA_UID.1 Timing of identification

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by existing CC requirements. It supports the security objective O.Non-repudiation and O.Control unknown source communication traffic.

NOTE: Based on FCO_NRO_CIMC.3, the TSF shall reject any information whose origin cannot be verified unless:

16) Acceptance of the information will not cause the TSF to perform any security relevant functions; and

b) Acceptance of the data will not cause the TSF to output or export any confidential information.

The TSF may, for example, accept information whose origin cannot be verified in the following cases:

16) The received information is a request for public information (e.g., an Online Certificate Status Protocol (OCSP) request).

b) The received information will not be processed until an authorized user has accepted its contents (e.g., a certificate request). In this case, the received information may be processed as if it had originated from the authorized user who approved it.

Application Note_D: In order to prevent replay FCO_NRO_CIMC.3.3 also comprises the verification if a given challenge/identity matches or not.

FCO_NRO_CIMC.4 Advanced verification of origin

FCO_NRO_CIMC.4.1 The TSF shall, for initial certificate registration messages sent by the certificate subject, only accept messages protected using an authentication code, keyed hash, or digital signature algorithm.

FCO_NRO_CIMC.4.2 The TSF shall, for all other security-relevant information, only accept the information if it was signed using a digital signature algorithm.

Dependencies: FCO_NRO_CIMC.3

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by existing CC requirements. It supports the security objective O.Non-repudiation.

6.5.1 Certificate Status Export

All CIMCs must be capable of exporting certificate status information. Any message sent by a CIMC containing certificate status information must meet the requirements for Certificate Status Export in addition to the requirements for Data Export specified in Section 6.6.

The following requirements apply to Certificate Status Export.

FDP_CIMC_CSE.1 Certificate status export

FDP_CIMC_CSE.1.1 Certificate status information shall be exported from the TOE in messages whose format complies with [*the X.509 [ITU-T X.509] standard for CRLs*].

Application Note: The ST should specify the format used to supply certificate status information. If a standard format is not used, then the ST shall include a description of the format.
Application Note_D: A standard format is used.

Dependencies: No dependencies

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC.

NOTE: As certificate status information is exported using the X.509 CRL format, the functional security requirements FDP_CIMC_CRL.1 and FMT_MOF_CIMC.5 apply.

6.6 Key Management

Cryptographic keys are used by CIMCs for many different reasons: to ensure the integrity of messages sent over untrusted networks, to authenticate users, to protect the confidentiality of private information, and to protect the confidentiality of stored information such as audit logs. As such, the unauthorized modification, disclosure, or substitution of any of these cryptographic keys could result in a loss of security.

Keys have a life cycle that begins with their generation. After generation, keys are stored, activated, deactivated, and destroyed. In many cases, keys are backed up and audited. Typically, public keys are distributed.

6.6.1 Public Key Storage

This subsection specifies security requirements that are designed to detect the unauthorized modification of public keys stored in a CIMC.

FDP_SDI_CIMC.3 Stored public key integrity monitoring and action

FDP_SDI_CIMC.3.1 Public keys stored within the CIMC, but not within a FIPS 140-2 validated cryptographic module, shall be protected against undetected modification through the use of digital signatures, keyed hashes, or authentication codes.

FDP_SDI_CIMC.3.2 The digital signature, keyed hash, or authentication code used to protect a public key shall be verified upon each access to the key. If verification fails, the TSF shall [*generate an audit log and not use this public key*].

Application Note: The ST should specify the actions to be taken in case the verification fails.

Application Note_D: Done

Dependencies: No dependencies

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC.

6.6.2 Private and Secret Key Destruction

This section specifies requirements for the zeroization/destruction of plaintext private and secret keys stored within CIMCs.

FCS_CKM_CIMC.5 CIMC private and secret key zeroization

FCS_CKM_CIMC.5.1 The TSF shall provide the capability to zeroize plaintext secret and private keys within the FIPS 140-2 validated cryptographic module.

Dependencies: FCS_CKM.4 Cryptographic key destruction
FDP_ACF.1 Security attribute based access control

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC.

6.7 Certificate Profile Management

A certificate profile defines the set of acceptable values for fields and extensions in a certificate. Examples of information that may be specified in a certificate profile include:

- constraints on the key owner's identifier (e.g., subject and/or subjectAltName in X.509);
- the set of allowable algorithms for the subject's public/private key pair;
- the certificate issuer's identifier (e.g., issuer and/or issuerAltName in X.509);
- the limitations on the length of time for which the certificate is valid;
- additional information that may/must be included in a certificate (e.g., which extensions may/must be included in an X.509 certificate);
- whether the subject of the certificate may be a CA;
- the types of operations that may be performed using the private key corresponding to the public key in the certificate (e.g., possible values for keyUsage and/or extKeyUsage in X.509);
- the policy (policies) under which the certificate may/must be issued.

FMT_MOF_CIMC.3 Extended certificate profile management

FMT_MOF_CIMC.3.1 The TSF shall implement a certificate profile and shall ensure that issued certificates are consistent with that profile.

FMT_MOF_CIMC.3.2 The TSF shall require the Administrator to specify the set of acceptable values for the following fields and extensions:

- the key owner's identifier;
- the algorithm identifier for the subject's public/private key pair;
- the identifier of the certificate issuer;
- the length of time for which the certificate is valid;

FMT_MOF_CIMC.3.3 If the certificates generated are X.509 public key certificates, the TSF shall require the Administrator to specify the set of acceptable values for the following fields and extensions:

- keyUsage;
- basicConstraints;
- certificatePolicies

FMT_MOF_CIMC.3.4 The Administrator shall specify the acceptable set of certificate extensions.

Dependencies: FMT_MOF.1 Management of security functions behavior
FMT_SMR.1 Security roles

Rationale: This component is necessary to specify a unique requirement of certificate issuing and management components that is not addressed by the CC. It supports the security objective O.Configuration Management.

6.8 Certificate Revocation List Profile Management

A certificate revocation list profile is used to define the set of acceptable values for fields and extensions in Examples of values that may be covered by a certificate revocation list profile include:

- **extensions** – the set of extensions that may/must be included in a CRL and the value of each extension's criticality bit.
- **issuer**, **issuerAltName** – the name of the CRL issuer.
- **nextUpdate** – a promise of next CRL in specified time.

FMT_MOF_CIMC.5 Extended certificate revocation list profile management

FMT_MOF_CIMC.5.1 If the TSF issues CRLs, the TSF must implement a certificate revocation list profile and ensure that issued CRLs are consistent with the certificate revocation list profile.

FMT_MOF_CIMC.5.2 If the TSF issues CRLs, the TSF shall require the Administrator to specify the set of acceptable values for the following fields and extensions:

- issuer;
- issuerAltName (NOTE: If a CIMC does not issue CRLs with this extension, then it is not required within the certificate revocation list profile.)
- nextUpdate (i.e., a promise of next CRL in specified time).

FMT_MOF_CIMC.5.3 If the TSF issues CRLs, the Administrator shall specify the acceptable set of CRL and CRL entry extensions.

Dependencies: FMT_MOF.1 Management of security functions behavior
FMT_SMR.1 Security roles

Rationale: This component is necessary to specify a unique requirement of certificate issuing and management components that is not addressed by the CC. It supports the security objective O.Configuration Management.

6.9 Certificate Registration

The functions in this section address the validation, approval, and signing of public key certificates. X.509 public key certificates issued by CIMCs must be compliant with the X.509 standard. Any fields or extensions to be included in an X.509 certificate will either be created by the CIMC according to the rules of the X.509 standard or validated by the CIMC to ensure compliance.

The data entered in each field and extension to be included in a certificate must be approved. Generally, a certificate field or extension value may be approved in one of four ways:

1) The data may be approved manually by an Officer.
2) An automated process may be used to review and approve the data.
3) The value for a field or extension may be automatically generated by the CIMC.
4) The value for a field or extension may be taken from the certificate profile.

FDP_CIMC_CER.1 Certificate Generation

FDP_CIMC_CER.1.1 The TSF shall only generate certificates whose format complies with [*the X.509 [ITU-T X.509] standard for public key certificates and BSI TR-03110 [BSI TR-03110] standard*].

Application Note: The ST should specify the format (or formats) used to generate certificates. If a standard format is not used, then the ST shall include a description of the format.
Application Note_D: Only standard formats for certificate generation are used.

FDP_CIMC_CER.1.2 The TSF shall only generate certificates that are consistent with the currently defined certificate profile.

FDP_CIMC_CER.1.3 The TSF shall verify that the prospective certificate subject possesses the private key that corresponds to the public key in the certificate request before issuing a certificate, unless the public/private key pair was generated by the TSF, whenever the private key may be used to generate digital signatures.

FDP_CIMC_CER.1.4 If the TSF generates X.509 public key certificates, it shall only generate certificates that comply with requirements for certificates as specified in ITU-T Recommendation X.509. At a minimum, the TSF shall ensure that:
a) The **version** field shall contain the integer **0**, **1**, or **2**.
b) If the certificate contains an **issuerUniqueID** or **subjectUniqueID** then the **version** field shall contain the integer **1** or **2**.
c) If the certificate contains **extensions** then the **version** field shall contain the integer **2**.
D) The **serialNumber** shall be unique with respect to the issuing Certification Authority.
e) The **validity** field shall specify a **notBefore** value that does not precede the current time and a **notAfter** value that does not precede the value specified in **notBefore**.
f) If the **issuer** field contains a null **Name** (e.g., a sequence of zero relative distinguished names), then the certificate shall contain a critical **issuerAltName** extension.
g) If the **subject** field contains a null **Name** (e.g., a sequence of zero relative distinguished names), then the certificate shall contain a critical **subjectAltName** extension.
h) The **signature** field and the **algorithm** in the **subjectPublicKeyInfo** field shall contain the OID for a FIPS-approved or recommended algorithm.

Dependencies: No dependencies

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC.

6.10 Certificate Revocation

The functions in this section address the validation and approval of certificate revocation information.

6.10.1 Certificate Revocation List Validation

Certificate revocation lists (CRLs) issued by CIMCs shall be compliant with the X.509 standard. Any fields or extensions to be included in a CRL shall be created by the CIMC according to the X.509 standard.

FDP_CIMC_CRL.1 Certificate revocation list validation

FDP_CIMC_CRL.1.1 A TSF that issues CRLs shall verify that all mandatory fields in any CRL issued contain values in accordance with ITU-T Recommendation X.509. At a minimum, the following items shall be validated:
1. If the **version** field is present, then it shall contain a **1**.
2. If the CRL contains any critical extensions, then the **version** field shall be present and contain the integer **1**.
3. If the **issuer** field contains a null **Name** (e.g., a sequence of zero relative distinguished names), then the CRL shall contain a critical **issuerAltName** extension.
4. The **signature** and **signatureAlgorithm** fields shall contain the OID for a FIPS-approved digital signature algorithm.
5. The **thisUpdate** field shall indicate the issue date of the CRL.
6. The time specified in the **nextUpdate** field (if populated) shall not precede the time specified in the **thisUpdate** field.

Dependencies: No dependencies

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC.

6.11 Strength of Function Requirements

Application Note_D: Please note that the functional security requirement FCS_SOF_CIMC.1 according CIMC PP [CIMC PP] is not fulfilled by the TOE but by the TOE environment. FIPS 140-2 level 3 validated cryptographic modules (HSMs) perform all cryptographic functions. The HSMs are required to operate in FIPS Mode. OE.Cryptographic Functions provides the appropriate security requirement for the TOE environment. Thus all [CIMC PP] requirements regarding cryptography according [CIMC PP], section Strength of Function Requirements, are implicitly fulfilled by the HSMs.

7 TOE SECURITY ASSURANCE REQUIREMENTS

The assurance requirements for the TOE are EAL 4 augmented with ALC_FLR.2 (flaw reporting procedures) as listed in Table 6. Please refer to the CIMC PP [CIMC PP], Section 7, for a detailed description of these requirements.

These requirements are designed to provide evidence that the CIMC has been methodically designed, tested and reviewed, and that it provides useful protection suitable for an environment requiring moderate to high confidence in security of commercial products at a reasonable development and evaluation cost.

Table 6 Assurance Requirements

Requirement Class	Requirement Component
ADV: Development	ADV_ARC.1: Security architecture description
	ADV_FSP.4: Complete functional specification
	ADV_IMP.1 Implementation representation of the TSF
	ADV_TDS.3: Basic modular design
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.4: Production support, acceptance procedures and automation
	ALC_CMS.4: Problem tracking CM coverage
	ALC_DEL.1: Delivery procedures
	ALC_DVS.1 Identification of security measures\
	ALC_FLR.2: Flaw reporting procedures
	ALC_LCD.1 Developer defined life-cycle model
	ALC_TAT.1 Well-defined development tools
ATE: Tests	ATE_COV.2: Analysis of coverage
	ATE_DPT.1 Testing: basic design
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing – sample
AVA: Vulnerability assessment	AVA_VAN.3: Focused vulnerability analysis

8 RATIONALE

This section includes the rationale for the functional and assurance requirements specified for the TOE. The rationale is based on specified objectives, threats, assumptions, and policies.

8.1 Security Objectives Rationale

This section demonstrates that the stated security objectives counter all identified threats, policies, or assumptions.

The following tables provide a mapping of security objectives to the environment defined by the threats, policies, and assumptions, illustrating that each security objective covers at least one threat, policy or assumption and that each threat, policy or assumption is covered by at least one security objective. Table 7 maps security objectives for the TOE to threats and Table 8 maps security objectives for the environment to threats. Table 9 maps the organizational security policies to security objectives. Table 10 maps assumptions to IT security objectives, listing which objectives each assumption helps to cover. The items in the tables are ordered alphabetically, sorted on the first column.

Table 7 Relationship of Security Objectives for the TOE to Threats

IT Security Objective	Threat
Authorized Users	
O.Certificates	T.Administrators, Officers and Auditors commit errors or hostile actions
O.Individual accountability and audit records	T.Administrative errors of omission, T.Hacker gains access, T.Administrators, Officers and Auditors commit errors or hostile actions, T.User abuses authorization to collect and/or send data
O.Limitation of administrative access	T.Disclosure of secret and private keys, T.Administrators, Officers and Auditors commit errors or
O.Maintain user attributes	T.Administrators, Officers and Auditors commit errors or hostile actions
O.Restrict actions before authentication	T.Hacker gains access, T.Administrators, Officers and Auditors commit errors or
O.Security roles	T.Administrators, Officers and Auditors commit errors or hostile actions
System	
O.React to detected attacks	T.Hacker gains access
Cryptography	
O.Non-repudiation	T.Sender denies sending information
O.Integrity protection of user data	T.Modification of private/secret keys, T.Malicious code exploitation
External Attacks	
O.Control unknown source communication traffic	T.Hacker gains access
Management	

O.Manage behavior of security functions	T.Critical system component fails, T.Administrators, Officers and Auditors commit errors or hostile actions
Audit	
O.Protect stored audit records	T.Modification of secret/private keys, T.Administrators, Officers and Auditors commit errors or hostile actions
O.Respond to possible loss of stored audit records	T.Administrators, Officers and Auditors commit errors or hostile actions
O.Configuration Management	T.Critical system component fails T.Malicious code exploitation

Table 8 Relationship of Security Objectives for the Environment to Threats

Non-IT Security Objective	Threat
OE.Administrators, Officers and Auditors guidance documentation	T.Disclosure of private and secret keys, T.Administrators, Officers and Auditors commit errors or hostile actions, T.Social engineering
OE.Competent Administrators, Officers and Auditors	T.Administrators, Officers and Auditors commit errors or hostile actions
OE.CPS	T.Administrative errors of omission
OE.Detect modifications of firmware, software, and backup data	T.User error makes data inaccessible, T.Administrators, Officers and Auditors commit errors or hostile actions
OE.HSM	T.Disclosure of private and secret keys
OE.Installation	T.Critical system component fails
OE.Lifecycle security	T.Critical system component fails, T.Malicious code exploitation
OE.Notify Authorities of Security Issues	T.Hacker gains access
OE.Object and data recovery free from malicious code	T.Modification of secret/private keys, T.Malicious code exploitation
OE.Periodically check integrity	T.Malicious code exploitation
OE.Physical Protection	T.Hacker physical access
OE.Procedures for preventing malicious code	T.Malicious code exploitation, T.Social engineering
OE.Repair identified security flaws	T.Flawed code, T.Critical system component fails
OE.Require inspection for downloads	T.Malicious code exploitation
OE.Security-relevant configuration management	T.Administrative errors of omission
OE.Social Engineering Training	T.Social Engineering
OE.Sufficient backup storage and effective restoration	T.Critical system component fails, T.User error makes data inaccessible
OE.Trusted Path	T.Hacker gains access, T.Message content modification
OE.Validation of security function	T.Malicious code exploitation, T.Administrators, Officers and Auditors
OE.Time stamps	T.Critical system component fails, T.Administrators, Officers and Auditors commit errors or hostile actions

OE.Preservation/trusted recovery of secure state	T.Critical system component fails
OE.Cryptographic functions	T.Disclosure of private and secret keys, T.Modification of secret/private keys

Table 9 Relationship of Organizational Security Policies to Security Objectives

Security Policy	Objective
P.Authorized use of information	OE.Auditors Review Audit Logs O.Maintain user attributes O.Restrict actions before authentication O.Security roles O.User authorization management
P.Cryptography	OE.Cryptographic functions

Table 10 Relationship of Assumptions to IT Security Objectives

Assumption	IT Security Objective
A.Auditors Review Audit Logs	OE.Auditors Review Audit Logs
A.Authentication Data Management	OE.Authentication Data Management
A.Communications Protection	OE.Communications Protection
A.Competent Administrators, Officers and Auditors	OE.Competent Administrators, Officers and Auditors, OE.Installation, OE.Security-relevant configuration management
A.Cooperative Users	OE.Cooperative Users
A.CPS	OE.CPS, OE.Security-relevant configuration management
A.HSM	OE.HSM
A.Disposal of Authentication Data	OE.Disposal of Authentication Data
A.Malicious Code Not Signed	OE.Procedures for preventing malicious code, OE.Require inspection for downloads, OE.Malicious Code Not Signed
A.Notify Authorities of Security Issues	OE.Notify Authorities of Security Issues
A.Operating System	OE.Operating System
A.Physical Protection	OE.Physical Protection
A.Social Engineering Training	OE.Social Engineering Training

8.1.1 Security Objectives Sufficiency

The following discussions provide information regarding:

Why the identified security objectives provide for effective countermeasures to the threats;

Why the identified security objectives provide complete coverage of each organizational security policy;

Why the identified security objectives uphold each assumption.

8.1.1.1 Threats and Objectives Sufficiency

8.1.1.1.1 Authorized users

T.Administrative errors of omission addresses errors that directly compromise organizational security objectives or change the technical security policy enforced by the system or application. It is countered by:

OE.CPS provides Administrators, Officers, and Auditors with information regarding the policies and practices used by the system. Providing this information ensures that these authorized users of the system are aware of their responsibilities, thus reducing the likelihood that they will fail to perform a security-critical operation.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. These audit records will expose administrators that fail to perform security-critical operations so they can be held accountable.

OE.Security-relevant configuration management ensures that system security policy data and enforcement functions, and other security-relevant configuration data are managed and updated. This ensures that they are consistent with organizational security policies and that all changes are properly tracked and implemented.

T.User abuses authorization to collect and/or send data addresses the situation where an authorized user abuses granted authorizations by browsing files in order to collect data and/or violates export control policy by sending data to a recipient who is not authorized to receive the data.

It is countered by:

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. This audit records will expose users who abuse their authorized to collect and/or send data.

T.User error makes data inaccessible addresses a user accidentally deleting user data. Consequently, the user data is inaccessible. Examples include the following:

User accidentally deletes data by striking the wrong key on the keyboard or by striking the enter key as an automatic response.

User does not understand the implications of the prompt at hand and inadvertently gives a response that deletes user data.

User misunderstands a system command and issues a command that unintentionally deletes user data.

It is countered by:

OE.Sufficient backup storage and effective restoration ensures that there is sufficient backup storage and effective restoration to recreate the system, when required. This ensures that user data is available from backup, even if the current copy is accidentally deleted.

OE.Detect modifications of firmware, software, and backup data ensures that if the backup components have been modified, that it is detected. If modifications of backup data cannot be detected, the backup copy is not a reliable source for restoration of user data.

T.Administrators, Officers and Auditors commit errors or hostile actions addresses:

Errors committed by administrative personnel that directly compromise organizational security objectives, change the technical security policy enforced by the system or application, or Malicious obstruction by administrative

personnel of organizational security objectives or modification of the system's configuration to allow security violations to occur.

It is countered by:

OE.Competent Administrators, Officers and Auditors ensures that users are capable of maintaining effective security practices. This reduces the likelihood that they will commit errors.

OE.Administrators, Officers and Auditors guidance documentation which deters administrative personnel errors by providing adequate guidance.

O.Certificates ensures that certificates, certificate revocation lists, and certificate status information are valid. The validation of information provided by Officers that is to be included in certificates helps to prevent improperly entered information from appearing in certificates.

OE.Detect modifications of firmware, software, and backup data ensures that if the backup components have been modified, that it is detected.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past user behavior to an authorized individual through system mechanisms. These audit records will expose administrators that perform inappropriate operations so they can be held accountable.

O.Limitation of administrative access. The administrative functions are designed in such a way that administrative personnel do not automatically have access to user objects, except for necessary exceptions. In general, the exceptions tend to be role specific. Limiting the set of operations that a user may perform limits the damage that a user may cause.

O.Maintain user attributes. Maintains a set of security attributes (which may include group membership, access rights, etc.) associated with individual users in addition to user identity. This prevents users from performing operations that they are not authorized to perform.

O.Manage behavior of security functions provides management controls/functions for security mechanisms. This ensures that security mechanisms which protect against hostile users are properly configured.

O.Protect stored audit records ensures that audit records are protected against unauthorized access, modification, or deletion to provide for traceability of user actions.

O.Respond to possible loss of stored audit records ensures that only auditable events executed by the Auditor shall be audited if the audit trail is full. This ensures that operations that are performed by users other than the Auditor are audited and so can be detected.

O.Restrict actions before authentication ensures that only a limited set of actions may be performed before a user is authenticated.

O.Security roles ensures that security-relevant roles are specified and that users are assigned to one (or more) of the defined roles. This prevents users from performing operations that they are not authorized to perform.

OE.Time stamps ensures that time stamps are provided to verify a sequence of events. This allows the reconstruction of a timeline of events when performing an audit review.

OE.Validation of security function. Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures such as underlying machine testing and integrity checks.

8.1.1.1.2 System

T.Critical system component fails addresses the failure of one or more system components that results in the loss of system-critical functionality. This threat is relevant when there are components that may fail due to hardware and/or software imperfections and the availability of system functionality is important.

It is countered by:

O.Configuration Management assures that a configuration management program is implemented. The configuration management program includes configuration identification and change control. This ensures that critical system components do not fail as a result of improper configuration.

OE.Installation ensures that the TOE is delivered, installed, managed, and operated in a manner which maintains IT
security. This ensures that critical system components do not fail as a result of improper installation.

O.Manage behavior of security functions provides management controls/functions for security mechanisms. This ensures that critical system components do not fail as a result of improper configuration of security mechanisms.

OE.Preservation/trusted recovery of secure state ensures that the system remains in a secure state throughout operation in the presence of failures and subsequent system recovery. This objective is relevant when system failures could result in insecure states that, when the system returns to operational mode (or continues to operate), could lead to security compromises.

OE.Sufficient backup storage and effective restoration ensures that there is sufficient backup storage and effective restoration to recreate the system, when required. This ensures that data is available from backup, even if the current copy is lost through failure of a system component (e.g., a disk drive).

OE.Time stamps provides time stamps to ensure that the sequencing of events can be verified. If the system must be reconstructed, it may be necessary to establish the order in which transactions were performed to return the system to a state consistent with the state when a critical component failed.

OE.Lifecycle security provides tools and techniques that are used throughout the development phase reducing the likelihood of hardware or software imperfections. **OE.Lifecycle security** also addresses the detection and resolution of flaws discovered during the operational phase that may result in failure of a critical system component.

OE.Repair identified security flaws. The vendor repairs security flaws that have been identified by a user. Such security flaws may result in critical system component failures if not repaired.

T.Flawed code addresses accidental or deliberate flaws in code made by the developer. Examples of accidental flaws are lack of engineering detail or bad design. An example of a deliberate flaw would be the inclusion of a trapdoor for later entry into the TOE.

It is countered by:

OE.Repair identified security flaws ensures that identified security flaws are repaired.

T.Malicious code exploitation addresses the threat where an authorized user, IT system, or hacker downloads and executes malicious code, which causes abnormal processes that violate the integrity, availability, or confidentiality of the system assets. The execution of malicious code is done through a triggering event.

It is countered by:

O.Configuration Management assures that a configuration management program is implemented. The configuration management program includes configuration identification and change control. This ensures that malicious code is not introduced during the configuration process.

O.Integrity protection of user data ensures that appropriate integrity protection is provided for user data. This prevents malicious code from attaching itself to user data.

OE.Object and data recovery free from malicious code ensures that the system recovers to a viable state after malicious code has been introduced and damage has occurred. The malicious code, e.g., virus or worm, is removed as part of the process.

OE.Periodically check integrity ensures that periodic integrity checks are performed on both system and software. If these checks fail, malicious code may have been introduced into the system.

OE.Procedures for preventing malicious code provides a set of procedures and mechanisms that work to prevent incorporation of malicious code into the system.

OE.Require inspection for downloads ensures that software that is downloaded/transferred is inspected prior to being made operational.

OE.Validation of security function. Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures such as underlying machine testing and integrity checks.

OE.Lifecycle security provides tools and techniques that are used throughout the development phase, reducing the likelihood that malicious code was included in the product by the developer. **OE.Lifecycle security** also addresses the detection and resolution of flaws discovered during the operational phase, such as modifications of components by malicious code.

OE.Malicious Code Not Signed protects the TOE from malicious code by ensuring all code is signed by a trusted entity prior to loading it into the system.

T.Message content modification addresses the situation where a hacker modifies information that is intercepted from a communications link between two unsuspecting entities before passing it on to the intended recipient. Several kinds of modification are possible: modification of a single message, deletion or reordering of selected messages, insertion of bogus messages, replay of previous messages, and modification of accompanying message security attributes.

It is countered by:

OE.Trusted Path ensures that a trusted path is established between the user and the system. The trusted path protects messages from interception or modification by a hacker.

8.1.1.1.3 Cryptography

T.Disclosure of private and secret keys addresses the unauthorized disclosure of secret and/or private keys.

It is countered by:

OE.Administrators, Officers and Auditors guidance documentation ensures that adequate documentation on securely configuring and operating the CIMC is available to Administrators, Officers and Auditors. This documentation will minimize errors committed by those users.

OE.Cryptographic functions ensures that TOE environment provides approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification; approved key generation techniques and uses validated cryptographic modules. Use of validated cryptographic modules ensures that cryptographic keys are adequately protected when they are stored within cryptographic modules.

O.Limitation of administrative access. The administrative functions are designed in such a way that administrative personnel do not automatically have access to user objects, except for necessary exceptions. In general, the exceptions tend to be role specific. Limiting the number of users who have access to cryptographic keys reducing the likelihood of unauthorized disclosure.

OE.HSM ensures that all Administrators, Officers and Auditor must only use smartcards as authentication token between them and the HSM via CXI library. As the smartcards are only used with the TOE that is – apart from CXI library – also present on the CA-Server it is ensured that the HSM is only used by the TOE. This prevents the unauthorized disclosure of secret and/or private keys.

T.Modification of private/secret keys addresses the unauthorized revision of a secret and/or private key.

It is countered by:

OE.Cryptographic functions ensures that TOE environment provides approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification; approved key generation techniques and uses validated cryptographic modules. Use of validated cryptographic modules ensures that cryptographic keys are adequately protected when they are stored within cryptographic modules.

O.Integrity protection of user data that ensures that appropriate integrity protection is provided for ensures that appropriate integrity protection is provided for user data.

OE.Object and data recovery free from malicious code ensures that the system recovers to a viable state after malicious code has been introduced and damage has occurred. If the malicious code cause private or secret keys to be revised in an unauthorized manner, this objective ensures that they are recovered to their correct values.

O.Protect stored audit records ensures that audit records are protected against unauthorized access, modification, or deletion to provide for traceability of user actions. This objective ensures that modifications to private and secret keys can be detected through the audit trail.

T.Sender denies sending information addresses the situation where the sender of a message denies sending the message to avoid accountability for sending the message and for subsequent action or inaction.

It is countered by:

O.Non-repudiation which ensures that the sender/originator of a message cannot successfully deny sending the message to the recipient.

8.1.1.1.4 External Attacks

T.Hacker gains access addresses:

Weak system access control mechanisms or user attributes

Weak implementation methods of the system access control

Vulnerabilities found in system or application code that allow a hacker to break into a system undetected.

It is countered by:

O.Restrict actions before authentication ensures that only a limited set of actions may be performed before a user is authenticated. This prevents a hacker who is unable to circumvent the access control mechanisms from performing security-relevant operations.

O.Control unknown source communication traffic ensures that communication traffic from an unknown source is controlled (e.g., rerouted or discarded) to prevent potential damage. Various kinds of hacker attacks can be detected or prevented by rerouting or discarding suspected hacker traffic.

O.Individual accountability and audit records provides individual accountability for audited events. Each user is uniquely identified so that auditable actions can be traced to a user. Audit records provide information about past

user behavior to an authorized individual through system mechanisms. This allows for the detection of unauthorized activity. Once detected, the damage resulting from such activity can be eliminated or mitigated.

OE.Notify Authorities of Security Issues ensures that proper authorities are notified regarding any security issues that impact their systems. This minimizes the potential for the loss or compromise of data.

O.React to detected attacks ensures that automated notification or other reactions to the TSFdiscovered attacks is implemented in an effort to identify attacks and to create an attack deterrent. This objective is relevant if actions that the organization deems essential also pose a potential attack that could be exploited.

OE.Trusted Path ensures that a trusted path is established between the user and the system. The trusted path is used to protect authentication data, thus reducing the likelihood that a hacker can masquerade as an authorized user.

T.Hacker physical access addresses the threat where an individual exploits physical security weaknesses to gain physical control of system components.

It is countered by:

OE.Physical Protection ensures that physical access controls are sufficient to thwart a physical attack on system components.

T.Social Engineering addresses the situation where a hacker uses social engineering techniques to gain information about system entry, system use, system design, or system operation.

It is countered by:

OE.Administrators, Officers and Auditors guidance documentation which deters administrative personnel errors by providing adequate guidance.

OE.Procedures for preventing malicious code provides a set of procedures and mechanisms that work to prevent incorporation of malicious code into the system. The introduction of malicious code into the system may be a goal of the social engineering attack.

OE.Social Engineering Training which ensures that general users, Administrators, Officers, and Auditors are trained in techniques to thwart social engineering attacks.

8.1.1.2 Policies and Objectives Sufficiency

P.Authorized use of information establishes that information is used only for its authorized purpose(s). This is addressed by the following objectives: **O.Maintain user attributes**, **O.Restrict actions before authentication**, **O.Security roles**, and **O.User authorization management**. **O.Restrict actions before authentication** ensures that the capability to perform security-relevant operations is limited to those who have been authorized to perform those operations. **O.Maintain user attributes**, **O.Security roles**, and **O.User authorization management** ensure that users are only authorized to perform those operations that are necessary to perform their Jobs. Finally, **OE.Auditors Review Audit Logs** deters users from misusing the authorizations they have been provided.

P.Cryptography establishes that accepted cryptographic standards and operations shall be used in the design of the TOE environment. This is addressed by **OE.Cryptographic functions** which ensures that such standards are used.

8.1.1.3 Assumptions and Objectives Sufficiency

8.1.1.3.1 Personnel

A.Auditors Review Audit Logs establishes that audit logs are necessary for security-relevant events and that they must be reviewed by auditors. This is addressed by **OE.Auditors Review Audit Logs**, which ensures that security-relevant events recorded in audit logs are reviewed by auditors.

A.Authentication Data Management establishes that management of user authentication data is external to the TOE. This is addressed by **OE.Authentication Data Management**, which ensures that users modify their authentication data in accordance with appropriate security policy.

A.Competent Administrators, Officers and Auditors establishes that security of the TOE is dependent upon those that manage it. This is addressed by **OE.Competent Administrators, Officers and Auditors**, which ensures that the system managers will be competent in its administration. **OE.Installation** further specifies that those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which maintains IT security. **OE.Security-relevant configuration management** further specifies that system security policy data and enforcement functions, and other security-relevant configuration data must be managed and updated, to ensure they are consistent with organizational security policies.

A.CPS establishes that Administrators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated. This is addressed by **OE.CPS,** which ensures that Administrators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated. **OE.Security-relevant configuration management** further specifies that system security policy data and enforcement functions, and other security-relevant configuration data shall be managed and updated, to ensure they are consistent with organizational security policies.

A.Disposal of Authentication Data establishes that users shall not retain access to the system after their authorization has been removed. This is addressed by **OE.Disposal of Authentication Data**, which ensures that access to the system will be denied after a user's privileges have been removed.

A.HSM requires that the HSM must only be used exclusively by the TOE. That is no other IT component is allowed to use the HSM. This is addressed by **OE.HSM**, which ensures that all Administrators, Officers and Auditor must only use smartcards as authentication token between them and the HSM via CXI library. As the smartcards are only used with the TOE that is – apart from CXI library – also present on the CA-Server it is ensured that the HSM is only used by the TOE.

A.Malicious Code Not Signed establishes that code not designed for the TOE will not be signed by a trusted party. This is addressed by **OE.Malicious Code Not Signed,** which ensures that code must be signed by a trusted party or it will not be loaded onto the system. **OE.Procedures for preventing malicious code** and **OE.Require inspection for downloads** further specifies that malicious code prevention procedures and mechanisms shall be incorporated, for example during donwloads/transfers.

A.Notify Authorities of Security Issues establishes that users notify proper authorities of any security issues that impact their systems to minimize the potential for the loss of compromise of data. This is addressed by **OE.Notify Authorities of Security Issues** which ensures that user notify proper authorities of any security issues that impact their systems.

A.Social Engineering Training establishes that individuals will attempt to gain access to the system using social engineering practices. This is addressed by **OE.Social Engineering Training**, which ensures that all users will be training to thwart social engineering attacks.

A.Cooperative Users establishes that a secure IT environment is required to securely operate the TOE, and that users must work within the constraints of that environment. This is addressed by **OE.Cooperative Users**, which ensures that users will cooperate with the constraints established.

8.1.1.3.2 Connectivity

A.Operating System establishes that an insecure operating system will compromise system security. This is addressed by **OE.Operating System**, which ensures that an operating system that meets security requirements recommended by the National Institute of Standards and Technology will be used.

8.1.1.3.3 Physical

A.Communications Protection establishes that the communications infrastructure is outside the TOE. This is addressed by **OE.Communications Protection**, which ensures that adequate physical protections are afforded the necessary communications infrastructure.

A.Physical Protection establishes that physical modification of the TOE and non TOE hardware, software, and firmware will compromise system security. This is addressed by **OE.Physical Protection**, which ensures that adequate physical protection will be provided.

8.2 Security Requirements Rationale

This section provides the rationale for necessity and sufficiency of security requirements, demonstrating that each of the security objectives is addressed by at least one security requirement, and that every security functional requirement is directed toward solving at least one objective.

8.2.1 Security Requirements Coverage

The following tables provide a mapping of the relationships of security requirements to objectives, illustrating that each security requirement covers at least one objective and that each objective is covered by at least one security requirement. The first table in this section, Table 11, addresses the mapping of security functional requirements to security objectives.

Table 11 Security Functional Requirements Related to Security Objectives

Functional Requirement	Objective
FAU_GEN.1 Audit data generation (iteration 2)	O.Individual accountability and audit records OE.Time stamps
FAU_GEN.2 User identity association (iteration 2)	O.Individual accountability and audit records
FAU_SEL.1 Selective audit (iteration 2)	O.Individual accountability and audit records
FAU_STG.1 Protected audit trail storage (iteration 2)	O.Protect stored audit records
FAU_STG.4 Prevention of audit data loss (iteration 2)	O.Respond to possible loss of stored audit records
FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	O.Non-repudiation, O.Control unknown source communication traffic
FCO_NRO_CIMC.4 Advanced verification of origin	O.Non-repudiation
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	OE.Procedures for preventing malicious code, O.React to detected attacks
FDP_ACC.1 Subset access control (iteration 2)	O.Limitation of administrative access
FDP_ACF.1 Security attribute based access control (iteration 2)	O.Limitation of administrative access
FDP_CIMC_CER.1 Certificate Generation	O.Certificates
FDP_CIMC_CRL.1 Certificate revocation list validation	O.Certificates
FDP_CIMC_CSE.1 Certificate status export	O.Certificates
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	O.Integrity protection of user data
FIA_ATD.1 User attribute definition	O.Maintain user attributes
FIA_SOS.1 Verification of secrets (iteration 2)	O.Limitation of administrative access O.React to detected attacks

FIA_UAU.1 Timing of authentication (iteration 2)	O.Limitation of administrative access, O.Restrict actions
FIA_UID.1 Timing of identification (iteration 2)	O.Individual accountability and audit records, O.Limitation of administrative access
FIA_USB.1 User-subject binding (iterations 2)	O.Maintain user attributes
FMT_MOF.1 Management of security functions behavior (iteration 2)	O.Configuration Management, O.Manage behavior of Security functions, O.Security-relevant configuration management
FMT_MOF_CIMC.3 Extended certificate profile management	O.Configuration Management
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	O.Configuration Management
FMT_MSA.1 Management of security attributes	O.Maintain user attributes, O.User authorization management
FMT_MTD.1 Management of TSF data	O.Individual accountability and audit records, O.Protect stored audit records
FMT_SMR.1 Security roles	O.Security roles
FPT_CIMC_TSP.1 Audit log signing event	O.Protect stored audit records

8.2.2 Security Requirements Sufficiency

8.2.2.1 Security Objectives for the TOE

8.2.2.1.1 Authorized Users

O.Certificates is provided by **FDP_CIMC_CER.1 (Certificate Generation)** which ensures that certificates are valid, and **FDP_CIMC_CRL.1 (Certificate revocation list validation)** and **FDP_CIMC_CSE.1 (Certificate status export)** which ensure that certificate revocation lists and certificate status information are valid.

O.Individual accountability and audit records is provided by a combination of requirements. **FIA_UID.1 (Timing of identification) (iteration 2)** covers the requirement that users be identified before performing any security-relevant operations. **FAU_GEN.1 (Audit data generation) (iterations 2)** and **FAU_SEL.1 (Selective audit) (iterations 2)** cover the requirement that security-relevant events be audited with date and time in the audit records while **FAU_GEN.2 (User identity association) (iterations 2)** cover the requirement that audit records contain identities of the entities responsible for the actions. **FMT_MTD.1 (Management of TSF data)** covers the requirement that audit data be available for review by ensuring that users, other than Auditors, cannot delete audit logs.

O.Limitation of administrative access is provided by **FDP_ACC.1 (Subset access control) (iteration 2)**, **FDP_ACF.1 (Security attribute based access control) (iteration 2)**, **FIA_SOS.1 (Verification of secrets) (iteration 2)**, **FIA_UAU.1 (Timing of authentication) (iteration 2)**, and **FIA_UID.1 (Timing of identification) (iteration 2)**. **FIA_UAU.1 (Timing of authentication) (iteration 2)**, **FIA_SOS.1 (Verification of secrets) (iteration 2)**, and **FIA_UID.1 (Timing of identification) (iteration 2)** ensure that Administrators, Officers, and Auditors cannot perform any security-relevant operations until they have been identified and authenticated and **FDP_ACC.1 (Subset access control) (iteration 2)** and **FDP_ACF.1 (Security attribute based access control) (iteration 2)** ensure that Administrators, Officers, and Auditors can only perform those operations necessary to perform their Jobs.

O.Maintain user attributes is provided by **FIA_ATD.1 (User attribute definition)** and **FIA_USB.1 (User- subject binding) (iteration 2)** which cover the requirement to maintain a set of security attributes associated with individual users and/or subjects acting on users' behalves. **FMT_MSA.1 (Management of security attributes)** ensures that only authorized users can modify security attributes.

O.Restrict actions before authentication is provided by **FIA_UAU.1 (Timing of authentication) (iteration 2)** which covers the requirement that no security-relevant actions are performed on behalf of a user until that user has been authenticated.

O.Security roles is provided by **FMT_SMR.1 (Security roles)** which covers the requirement that a set of security roles be maintained and that users be associated with those roles.

O.User authorization management is provided by **FMT_MSA.1 (Management of security attributes)** which covers the requirement that Administrators manage and update user's security attributes.

8.2.2.1.2 System

O.React to detected attacks is provided by **FCS_CKM_CIMC.5 (CIMC private and secret key zeroization)** which cover the requirement that the user who detected the attack be able to destroy any plaintext keys within the TOE in order to prevent the attacker from obtaining copies of these keys. **FIA_SOS.1 (Verification of secrets) (iteration 2)** covers the requirement that the TSF respond to detected attacks (in the form of repeated authentication attempts) by taking actions to prevent the attacker from successfully authenticating him/herself. In the case that an attack is detected by an Administrator, Auditor or Officer.

8.2.2.1.3 External Attacks

O.Control unknown source communication traffic is provided by **FCO_NRO_CIMC.3 (Enforced proof of origin and verification of origin)** which covers the requirement that the TOE discard messages from an unknown source that contain security-relevant information.

8.2.2.1.4 Cryptography

O.Non-repudiation is provided by **FCO_NRO_CIMC.3 (Enforced proof of origin and verification of origin)** which covers the requirement that messages containing security-relevant data are not accepted by the TOE unless they contain evidence of origin and **FCO_NRO_CIMC.4 (Advanced verification of origin)** which covers the requirement that digital signatures be used so that the evidence of origin for a message may be verified by a third-party.

O.Integrity protection of user data is provided by **FDP_SDI_CIMC.3 (Stored public key integrity monitoring and action)** which cover the requirement that user data be protected.

8.2.2.1.5 Management

O.Manage behavior of security functions is provided by **FMT_MOF.1 (Management of security functions behavior) (iteration 2)** which covers the requirement that authorized users be able to configure, operate, and maintain the security mechanisms.

O.Configuration Management is provided by **FMT_MOF.1 (Management of security functions behavior) (iteration 2)** which covers the requirement that only authorized users can change the configuration of the system. **FMT_MOF_CIMC.3 (Extended certificate profile management)** covers the requirement that Administrators be able to control the types of information that are included in generated certificates. **FMT_MOF_CIMC.5 (Extended certificate revocation list profile management)** covers the requirement that Administrators be able to control to the types of information that are included in generated certificate revocation lists.

8.2.2.1.6 Audit

O.Protect stored audit records is provided by **FAU_STG.1 (Protected audit trail storage) (iteration 2)** which covers the requirement that audit records be protected against modification or unauthorized deletion and **FMT_MTD.1 (Management of TSF data)** which covers the requirement that audit records be protected from unauthorized access. Where the threat of malicious activity is greater, **FPT_CIMC_TSP.1 (Audit log signing event)** is required so that modifications to the audit logs can be detected.

O.Respond to possible loss of stored audit records is provided by **FAU_STG.4 (Prevention of audit data loss) (iteration 2)** which covers the requirement that no auditable events, except those taken by the Auditor, can be performed when audit trail storage is full.

8.3 Requirement Dependency Rationale

This section demonstrates that the stated security requirements together form a mutually supportive and internally consistent whole. Internal consistency is demonstrated in an analysis of dependencies. Mutual support is shown through consideration of the interactions between and among the SFRs.

8.3.1 Rationale that Dependencies are satisfied

The selected security requirements include related dependencies, both direct and indirect. The indirect dependencies are those required by the direct dependencies. All of these dependencies must be met or their exclusion justified.

8.3.1.1 Security Functional Requirements Dependencies

The following table provides a summary of the security functional requirements dependency analysis.

Table 12 Summary of Security Functional Requirements Dependencies

Component	Dependencies	Which is:
FAU_GEN.1 Audit data generation (iteration 2)	FPT_STM.1 Reliable time stamps	Not included
FAU_GEN.2 User identity association (iteration 2)	FAU_GEN.1 Audit data generation	FAU_GEN.1 Audit data generation (iteration 2)
	FIA_UID.1 Timing of identification	FIA_UID.1 Timing of identification (iteration 2)
FAU_SEL.1 Selective audit (iteration 2)	FAU_GEN.1 Audit data generation	FAU_GEN.1 Audit data generation (iteration 2)
	FMT_MTD.1 Management of TSF data	FMT_MTD.1 Management of TSF data
FAU_STG.1 Protected audit trail storage (iteration 2)	FAU_GEN.1 Audit data generation	FAU_GEN.1 Audit data generation (iteration 2)
FAU_STG.4 Prevention of audit data loss (iteration 2)	FAU_STG.1 Protected audit trail storage	FAU_STG.1 Protected audit trail storage (iteration 2)
FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	FIA_UID.1 Timing of identification	FIA_UID.1 Timing of identification (iteration 2)
FCO_NRO_CIMC.4 Advanced verification of origin	FCO_NRO_CIMC.3	FCO_NRO_CIMC.3 Enforced proof of origin and verification of
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	FCS_CKM.4 Cryptographic key destruction	Not included
	FDP_ACF.1 Security attribute based access control	FDP_ACF.1 Security attribute based
FDP_ACC.1 Subset access control (iteration 2)	FDP_ACF.1 Security attribute based access control	FDP_ACF.1 Security attribute based
FDP_ACF.1 Security attribute based access control (iteration 2)	FDP_ACC.1 Subset access control	FDP_ACC.1 Subset access control (iteration 2)
	FMT_MSA.3 Static attribute initialization	Not included

FDP_CIMC_CER.1 Certificate Generation	None	
FDP_CIMC_CRL.1 Certificate revocation list validation	None	
FDP_CIMC_CSE.1 Certificate status export	None	
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	None	
FIA_ATD.1 User attribute definition	None	
FIA_SOS.1 Verification of secrets (iteration 2)	None	
FIA_UAU.1Timing of Authentication (iteration 2)	FIA_UID.1 Timing of identification	FIA_UID.1 Timing of identification (iteration 2)
FIA_UID.1 Timing of identification (iteration 2)	None	
FIA_USB.1 User-subject binding	FIA_ATD.1 User attribute definition	FIA_ATD.1 User attribute
FMT_MOF.1 Management of security functions behavior (iteration 2)	FMT_SMR.1 Security roles	FMT_SMR.1 Security roles
	FMT_SMF.1 Specification of Management Functions	Not included
FMT_MOF_CIMC.3 Extended certificate profile management	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1 Management of security functions behavior
	FMT_SMR.1 Security roles	FMT_SMR.1 Security roles
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1 Management of security functions behavior
	FMT_SMR.1 Security roles	FMT_SMR.1 Security roles
FMT_MSA.1 Management of security attributes	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Subset access control (iteration 2)
	FMT_SMR.1 Security roles	FMT_SMR.1 Security roles
	FMT_SMF.1 Specification of Management Functions	Not included
FMT_MTD.1 Management of TSF data	FMT_SMR.1 Security roles	FMT_SMR.1 Security roles
	FMT_SMF.1 Specification of Management Functions	Not included
FMT_SMR.1 Security roles	FIA_UID.1 Timing of identification	FIA_UID.1 Timing of identification (iteration 2)
FPT_CIMC_TSP.1 Audit log signing event	FAU_GEN.1 Audit data generation	FAU_GEN.1 Audit data generation (iteration 2)
	FMT_MOF.1 Management of security functions behavior	Included

8.3.1.1.1 Justification of Unsupported Dependencies Regarding FMT_SMF.1

The following components depend on FMT_SMF.1 Specification of Management Functions:

- FMT_MOF.1 Management of security functions behavior
- FMT_MSA.1 Management of security attributes
- FMT_MTD.1 Management of TSF data

This requirement need not be explicitly covered by the product since requirements in Table 4 meet or exceed the requirement for FMT_SMF.1 Specification of Management Functions.

8.3.1.1.2 Justification of Unsupported Dependencies Regarding FCS_CKM.4

Not applicable, as the TOE triggers cryptographic key destruction but the HSM destroys cryptographic keys (see OE.Cryptographic).

8.3.1.1.3 Justification of Unsupported Dependencies Regarding FMT_MSA.3

Not applicable, since the default value for the security attribute “role” of a new user is “none”. Thus always restrictive values are enforced. This can’t be changed.

8.3.1.1.4 Justification of Unsupported Dependencies Regarding FPT_STM.1 Reliable time stamps

Not applicable, since the reliable time stamps are provided by the TOE environment (see OE.Time stamps).

8.4 Extended Requirements Rationale

This ST includes a number of extended requirements. Each of the extended requirements is defined in the CIMC PP and rationale immediately follows the statement of each such requirement. The extended requirements can be identified by the use of the keyword “CIMC” in the requirement component and element identifiers.

9 ACCESS CONTROL POLICIES

9.1 CIMC TOE Access Control Policy

The TOE shall support the administration and enforcement of a CIMC TOE access control policy that provides the capabilities described below.

Subjects (Certificate Request, Certificate Revocation List Request) will be granted access to objects (certificates and CRLs) based upon the:

- Identity of the subject requesting access,
- Role (or roles) the subject is authorized to assume,
- Type of access requested,

Subject identification includes roles with different access authorizations.

Access to objects is defined by the simple access types used on access rules:

- accept;
- decline;
- unused (default).

The default access decision for an unused access rule is to deny access.

The TOE will support the following predefined roles:

- Administrator: role authorized to install, configure and maintain the TOE, establish and maintain user accounts, configure profiles, access rights and audit parameters and manage Component keys;
- Auditor: role authorized to view and maintain audit logs;
- Officer: role authorized to request or approve certificates or certificate revocations.

10 TOE Summary Specification

10.1 TOE security functionality

10.1.1 SF1 Security Audit

10.1.1.1 SF1.1 Audit message generation

The Audit (also called Audit system or Audit unit) logs the security-relevant events that were performed by the TOE. These events are either triggered internally or by external components/users via Java methods. That is the CA-Core logs amongst others every event and the appropriate event state, in the case that this event triggers a process of the CA-Core.

The CA-Core generates audit messages for the following auditable events (FAU_GEN.1 Audit data generation (iteration 2)):

- Star-up and shutdown of the audit functions and
- further security-relevant events according to Table 3

These audit messages are send to the Audit.

If the audit trail is full the TOE shutdowns (FAU_STG.4).

10.1.1.2 SF1.2 Audit trail protection

After audit message generation the Audit unit of the TOE generates uniquely identifiable audit messages, so called audit records.

The Audit is able to associate each auditable event with the identity of the user that caused the event as the identity (UserIdentity) is contained in the audit record (see Section 12.2; FAU_GEN.2 User identity association (iteration 2)).

The Audit is able to select the set of events to be audited from the set of all auditable events based on the following attributes contained in the audit record (see Section 12.2; FAU_SEL.1 Selective audit (iteration 2): object identity (Module), user identity (UserIdentity) and event type (EventType).

The TOE triggers that a set of these chronological ordered audit records (called audit trail) are periodically signed by means of a digital signature by the Hardware Security Module, resulting in a so called protected audit trail (see Sections 12.3 and 12.4; FPT_CIMC_TSP.1 Audit log signing event). This period is configurable. In order to protect audit messages against modification or deletion the Audit uses timestamps (OE.Time stamps) and sequence numbers.

The Audit also triggers further cryptographic operations with HSM to protect the audit messages (FAU_STG.1 Protected audit trail storage (iteration 2)). The Audit needs three different cryptographic keys to protect the audit trails. It needs two asymmetric key pairs, one signature key pair (ASK) and one encryption key (AEK) and also one current symmetric trail record key (TRK). All these keys are generated (See SF6) within and stored on the HSM. The asymmetric keys are generated during the bootstrap process of the TOE (see Section 2.7). Audit records and audit trails are stored via Java-API in the Adapter.

Table 13 Audit trail protection during operation

Step	Operation
1	The audit system generates a symmetric key TRK using the security module
2	The symmetric key TRK will encrypted with the public key AEK_{pub} resulting in ENC_{AEKpub} (TRK) using the security module
3	The current audit trail sequence number $TRAIL_{SEQ}$ is determined The audit system creates trail header data object and generates an following signature using the security module $SIG_{ASKpriv}(TRAIL_{SEQ} \mid Timestamp \mid ENC_{AEKpub}(TRK))$
4	The new $TRAIL\_REC_{SEQ}$ will be set to zero.
5	If a preceding trail is available, then the following step will performed using the security nodule. The audit system generates a MAC over the new trail key $ENC_{AEKpub}(TRK)$ with the last trail key TRK of the preceding trail MAC_{TRK} ($ENC_{AEKpub}(TRK)$) The audit system generates a signature over the preceding trail footer $SIG_{ASKpriv}(TRAIL\_SEQ \mid TRAIL\_REC_{SEQ} \mid MAC_{TRKn}(TRAIL\_REC_{SEQ}) \mid MAC_{TRK}$ ($ENC_{AEKpub}(TRK)$)) The audit stores the footer information of the preceding audit trail. The trail key TRK of the preceding trail will deleted. The preceding audit trail is now closed.
6	The audit system stores the new audit trail header including the signature $SIG_{ASKpriv}(TRAIL_{SEQ} \mid Timestamp \mid ENC_{AEKpub}(TRK))$ The process of creating the new audit trail is now finished.
7	CA-Core sends a new log entry. The audit system creates a new audit record REC and calculates the $MAC_{TRK}(REC)$ using the security module
8	The audit system stores the REC and the $MAC_{TRK}(REC)$
9	The $TRAIL\_REC_{SEQ}$ incremented by one
10	The process continues at step 7 if the trail capacity not reached yet. After reaching the maximum audit trail capacity the process will continue with step 1.

Additional the audit system performs the following steps on a regular shutdown of the audit system:

Table 14 Audit trail protection on a regular shutdown

Step	Operation
1	The audit system logs the shutdown process.
2	The audit system creates the trail footer with the current information $TRAIL_{SEQ} \mid TRAIL\_REC_{SEQ} \mid MAC_{TRK}(TRAIL\_REC_{SEQ}) \mid SIG_{ASKpriv}(TRAIL\_SEQ \mid TRAIL\_REC_{SEQ} + 1 \mid MAC_{TRK}(TRAIL\_REC_{SEQ} + 1) \mid MAC_{TRK}$ ($ENC_{AEKpub}(TRK)$)) The field of the MAC over the next exported TRK for the next audit trail will not filled by the audit system

10.1.2 SF2 Management of the TSF

At the first startup the CA-Core has no configuration. Thus, the CA-Core must first be configured (CA configuration, CAProfile (FMT_MOF_CIMC.3.2 and FMT_MOF_CIMC.3.3) and CRLProfile (FMT_MOF_CIMC.5.2 and FMT_MOF_CIMC.5.3)) via the Java-API.

The CA-Core performs the same checks for Java configuration method as described in SF3.2. That is certificate validation, signature verification, challenge/identity check and role check. If all checks succeed, the CAProfile

fulfills the requirements FMT_MOF_CIMC.3.2 and FMT_MOF_CIMC.3.3 and the CRLProfile fulfills the requirements FMT_MOF_CIMC.5.2 and FMT_MOF_CIMC.5.3, the Audit generates an audit log (see SF1.1) and the CA-Core triggers the generation of a new symmetric key within HSM (see SF6). Then the CA-Core triggers HMAC [RFC2104] protection (see SF6) of the configuration within HSM. Finally the CA-Core stores the HMAC protected configuration via Java-API to the Adapter.

In order to prevent replay every change of a configuration requires that the CA-Core triggers the generation a new symmetric key (see SF6) and the deletion of the formerly used symmetric key within HSM (see SF6, FCS_CKM_CIMC.5).

If a configuration is needed during processing the CA-Core loads all information via Java-API from the Adapter. Then the CA-Core triggers HMAC verification within HSM (see SF6). If HMAC verification fails the Audit generates an audit log record (see SF1.1) and the CA-Core does not further continue processing. If HMAC verification succeeds the CA-Core Job processing is continued.

10.1.3 SF3 Data Authenticity and Authorization

10.1.3.1 SF3.1 Challenge Request and Response

In order to prevent replay the CA-Core triggers a challenge-response algorithm (FCO_NRO_CIMC.3.3). In a first step the external component must request a challenge via Adapter from the CA-Core. The request is not cryptographic protected according FIA_UAU.1 (iteration 2) and FIA_UID.1 (iteration 2). The CA-Core then triggers generation of a challenge (10 Byte) within HSM (see SF6). As the HSM operates in FIPS mode the HSMs Deterministic Random Number Generator (DRNG) is mandatory used to generate the challenge. The CA-Core then stores the challenge with the user identification given in the request and sends the challenge back to the external component via Adapter. Now the external component may request Job processing via Adapter in a second step. A Job must contain amongst others the requested challenge and must be signed with the user's private key.

10.1.3.2 SF3.2 Remote Data entry Verification, Authorization and Challenge Verification

Before CA-Core starts a particular process it performs the following checks to ensure the integrity of the consigned Java method data: The CA-Core

- performs user certificate validation and the appropriate certificate chain validation (FCO_NRO_CIMC.3.3)
- performs the signature verification with all consigned data (FCO_NRO_CIMC.3.3, FIA_UAU.1 (iteration 2) and FIA_UID.1 (iteration 2)
- checks whether the given challenge and the signature identity matches a stored challenge/identity (FCO_NRO_CIMC.3.3) and
- checks whether the role (FIA_USB.1) of the signature identity has the right to perform the requested process according to FMT_MOF.1.1 (for example creating a new certificate or a new certification revocation list). The security attribute role belongs to individual users (FIA_ATD.1). The allowed roles are: Administrator, Auditor and Officer (FMT_SMR.1). The right to modify configuration files and profiles (FMT_MTD.1) and modify the security attribute role (FMT_MSA.1) is limited to Administrators.

If all checks succeed, the Audit generates an audit log record (see SF1.1) and starts request processing. If a check fails, the Audit generates an audit log record (see SF1.1) and the CA-Core does not start request processing.

10.1.4 SF4 Certificate and Certificate Status management

The TOE triggers generation of X.509 certificates and CRLs according to the standards:

• X.509v3 [ITU-T X.509] and

• RFC 5280 [RFC 5280].

In addition to this, the TOE also generates CVC for EAC e-Passport infrastructure according to the BSI TR-03110 [BSI TR-03110] standard.

The TOE maintains via Adapter all issued certificates and their current state in a database, in order to serve status information. Status information of certificates is made available through CRLs and delta CRLs (RFC 5280 [RFC5280]).

10.1.4.1 SF4.1 Certificate Generation

In case of a certificate request the CA-Core

- validates the certificate request against the loaded CAProfile (FDP_CIMC_CER.1.2 and FMT_MOF_CIMC.3.1),
- triggers signature verification of the certificate request within HSM (FDP_CIMC_CER.1.3 and FCO_NRO_CIMC.4),
- transforms the CAProfile and merge it with the certificate request into a certification template (FDP_CIMC_CER.1.1 and FDP_CIMC_CER1.4),
- triggers signing of certificate template to generate a certificate within HSM (see SF6) and
- returns the new certificate via Java-API to the Adapter.

10.1.4.2 SF4.2 Certificate Revocation

In case of a certificate revocation list request the CA-Core

- merges the CRLProfile and the list of revoked certificates into the certificate revocation list template (FDP_CIMC_CRL.1.1 and FMT_MOF_CIMC.5.1),
- triggers singing of the certificate revocation list template within HSM (see SF6) and
- returns the new certificate revocation list via Java-API to the Adapter.

10.1.4.3 SF4.3 Certificate Status Export

Issued CRLs are stored via Java-API in the Adapter (FDP_CIMC_CSE.1 Certificate status export).

10.1.5 SF5 Access Control

The TOE enforces the CIMC TOE Access Control Policy specified in Section 9.1. The access to resources in the TOE is controlled using access control lists according to FDP_ACC.1 (iteration 2), based on:

- access rule – accept or decline access to a resource,
- resource – a resource to which access is controlled,
- user – an entity that have access rights to a resource,
- role – a role that a user is allowed to take on. Since access rules are defined on a role, so for a user to have access rights he must be assigned roles.

When a controlled resource is accessed, the CA-Core verifies that the caller meets the appropriate access rules for the resource and, if not, denies access and generates an error. If there are no access rules associated to the resource, access is denied. The TOE access control system maps authentication information to a user entity. The entity is then associated to a role in order to acquire privileges according FDP_ACF.1 (iteration 2).

10.1.6 SF6 Cryptographic Key Management

For cryptographic operations the TOE relies on a FIPS 140-2 Level 3 [FIPS140-2] validated cryptographic module – a Hardware Security Module (HSM) – according to the Certificate Issuing and Management Components (CIMC) Protection Profile [CIMC PP]. All cryptographic operations (key generation, hashing, signing, verifying and key zeroizing) are performed within this validated cryptographic module. Of course, the TOE triggers all cryptographic operations of the HSM. The HSM runs in FIPS mode. Here, FIPS mode means FIPS approved mode of operation according to [FIPS140-2].

The TOE only manages Component keys. Component keys are used to sign certificates and certificate status information. Component keys are also used to sign audit logs and to ensure the integrity of changed Jobs by CA-Core. Component private keys are only stored on the HSM.

The integrity and authenticity of public keys stored by the TOE on the database – outside the HSM – is protected by the usage of a digital signature, namely of the digital certificate structure in which it has been included (FDP_SDI_CIMC.3.1). Every time a public key needs to be used to perform any cryptographic operation, its protective digital signature will be verified and, in case of failure, an audit log entry (see SF1.1) will be generated and the key will be marked as tampered with, becoming unusable for all types of operations (FDP_SDI_CIMC.3.2).

The TOE triggers zeroizing plaintext Component private keys within the HSM, if required (FCS_CKM_CIMC.5).

The TOE may trigger the following cryptographic operations within the HSM in FIPS mode, which includes amongst others all cryptographic operations required for FCO_NRO_CIMC.3.3:

Generate Key

This function creates a new key (or key pair) for DES, AES, RSA, ECDSA, ECDH:

- DES: 112 and 168 bits (16 or 24 bytes)
- AES: 128, 192 or 256 bits
- RSA: no modulus sizes less than 1024 bits
- ECDSA, ECDH: In FIPS mode, key generation relies on the HSM's deterministic random number generator that is compliant with FIPS 186-2, Appendix 3.1 [FIPS 186-2]. In FIPS mode only elliptic curves P-192, P-224, P-256, P-384, P-521, K-163, K-233, K-283, K-409, K-571, B-163, B-233, B-283, B-409 or B-571 as specified in FIPS 186-2 Appendix 6 are allowed (see also Appendix for further details).

Crypt Data

This function encrypts or decrypts data with a DES or AES key in ECB or CBC mode. In FIPS mode RSA keys are not allowed for data de- or encryption.

AES (128, 192 or 256 bits)

DES (112 and 168 bits (16 or 24 bytes))

Sign Data

This function calculates

- for asymmetric algorithms: a signature for a given hash value, or
- for symmetric algorithms: a message authentication code (MAC).

In FIPS mode, only the SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 hashing algorithms can be used.

The PKCS #1 v1.5 [PKCS#1] encoding is used.

Verify Signature

This function verifies a signature or MAC.

In FIPS mode, only the SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 hashing algorithms can be used.

The PKCS #1 v1.5 [PKCS#1] encoding is used.

Compute Hash

This function computes a hash or HMAC value over given data or key components with a chosen algorithm. In FIPS mode, only the SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 hashing algorithms or HMAC can be used.

Agree Secret

This function derives a shared secret from a public and a private EC key agreement key according to [TR-03111] chapter 4.3.1.

Generate Random Number

This function generates a random number of arbitrary lengths. In FIPS mode, always the Deterministic Random Number Generator (DRNG) is used. This DRNG is based on the SHA-512 algorithm as transition function and compliant to [NIST SP 800-90].

10.2 Fulfilling the security functional requirements

See Section 10.1 and:

- Since the TOE performs identification and authentication only via signature verification algorithms FIA_SOS.1 (iteration 2) is implicitly fulfilled.

11 GLOSSARY OF TERMS

The following definitions are used throughout this standard:

Authentication code: a cryptographic checksum, based on a FIPS-approved or recommended security method; also known as a Message Authentication Code (MAC) in ANSI standards.

CIMC: the set of hardware, software, firmware, or some combination thereof, that issues, revokes, and manages public key certificates and certificate status information, and is contained within the CIMC boundary.

CIMC boundary: an explicitly defined contiguous perimeter that establishes the physical bounds of a CIMC.

Complete CRL: a complete CRL lists all unexpired certificates, within its scope, that have been revoked for one of the revocation reasons covered by the CRL scope.

Compromise: the unauthorized disclosure, modification, substitution or use of sensitive data (including plaintext cryptographic keys and other CSPs).

Confidentiality: the property that sensitive information is not disclosed to unauthorized individuals, entities or processes.

CMP: a component of the secunet eID PKI Suite that supports the Certificate Management Protocol arrording RFC4210 and RFC4211.

Critical security parameter (CSP): security-related information (e.g., secret and private cryptographic keys, authentication data such as passwords and PINs) appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a CIMC or the security of the information protected by the CIMC.

Cryptographic key (key): a parameter used in conjunction with a cryptographic algorithm that determines:

- the transformation of plaintext data into ciphertext data,
- the transformation of ciphertext data into plaintext data,
- a digital signature computed from data,
- a keyed hash computed from data,
- the verification of a digital signature computed from data,
- an authentication code computed from data, or
- an exchange agreement of a shared secret.

Cryptographic key component (key component): a parameter used in conjunction with other key components in a FIPS-approved or recommended security method to form a plaintext cryptographic key or perform a cryptographic function.

Delta CRL: a delta CRL only lists those certificates, within its scope, whose revocation status has changed since the issuance of a referenced complete CRL. The referenced complete CRL is referred to as a base CRL.

Digital signature: a non-forgeable transformation of data that allows proof of the source (with non-repudiation) and verification of the integrity of that data.

Encrypted key: a cryptographic key that has been encrypted with a key encrypting key, a PIN or a password in order to disguise the value of the underlying plaintext key.

Error detection code (EDC): a code computed from data and comprised of redundant bits of information designed to detect, but not correct, unintentional changes in the data.

FIPS-Approved or recommended mode of operation: a mode that employs only the operation of FIPS-approved or recommended security methods.

FIPS-approved or recommended security method: a security method (e.g., cryptographic algorithm, cryptographic key generation algorithm or key distribution technique, authentication technique, or evaluation criteria) that is either a) specified in a FIPS or b) adopted in a FIPS and specified either in an appendix to the FIPS or in a document referenced by the FIPS.

Firmware: the programs and data stored in hardware (e.g., ROM, PROM, or EPROM) such that the programs and data cannot be dynamically written or modified during execution. *Hardware*: the physical equipment used to process programs and data in a CIMC.

HMAC: Keyed-Hash Message Authentication Code, RFC2104

Integrity: the property that sensitive data has not been modified or deleted in an unauthorized and undetected manner.

Key encrypting key: a cryptographic key that is used for the encryption or decryption of other keys.

Key management: the activities involving the handling of cryptographic keys and other related security parameters (e.g., Ivs, passwords) during the entire life cycle of the keys, including their generation, storage, distribution, entry and use, deletion or destruction, and archiving.

nSPOC: see SPOC

Password: a string of characters (letters, numbers, and other symbols) used to authenticate an identity or to verify access authorization.

Personal Identification Number (PIN): a 4 or more character alphanumeric code or password used to authenticate an identity, commonly used in banking applications.

Physical protection: the safeguarding of a CIMC, cryptographic keys, or other CSPs using physical means.

Plaintext key: an unencrypted cryptographic key.

Private key: a cryptographic key used with a public key cryptographic algorithm, uniquely associated with an entity, and not made public.

Protection Profile: an implementation-independent set of security requirements for a category of Targets of Evaluation (TOEs) that meet specific consumer needs.

Public key: a cryptographic key used with a public key cryptographic algorithm, uniquely associated with an entity, and which may be made public. (Public keys are not considered CSPs.)

Public key certificate: a set of data that unambiguously identifies an entity, contains the entity's public key, is digitally signed by a trusted party, and binds the public key to the entity.

Public key (asymmetric) cryptographic algorithm: a cryptographic algorithm that uses two related keys, a public key and a private key. The two keys have the property that, given the public key, it is computationally infeasible to derive the private key.

Secret key: a cryptographic key used with a secret key cryptographic algorithm, uniquely associated with one or more entities, and which shall not be made public. The use of the term "secret" in this context does not imply a classification level rather the term implies the need to protect the key from disclosure or substitution.

Secret key (symmetric) cryptographic algorithm: a cryptographic algorithm that uses a single, secret key for both encryption and decryption.

Security policy: a precise specification of the security rules under which a CIMC shall operate, including the rules derived from the requirements of this document and additional rules imposed by the vendor.

Software: the programs and associated data that can be dynamically written and modified.

Split knowledge: a condition under which two or more entities separately have key components that individually convey no knowledge of the plaintext key that will be produced when the key components are combined in the cryptographic module.

SPOC: a component of the secunet eID PKI Suite that supports the communication protocol CSN 36 9791 (SPOC) and the Technical Guideline 03129 of the German Federal Office for Information Security. nSPOC is a national SPOC.

Target of Evaluation (TOE) – An information technology product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.

TOE Security Functions (TSF) – A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TOE Security Policy (TSP) – A set of rules that regulate how assets are managed, protected and distributed within a TOE.

Trusted path: a means by which a user and a TSF can communicate with the necessary confidence to support the TSP.

User: an individual, or a process (subject) operating on behalf of the individual, accessing CIMC.

Zeroization: a method of erasing electronically stored data by altering or deleting the contents of the data storage so as to prevent the recovery of the data.

12 Appendix

12.1 Built-in Elliptic Curves

The following table lists all built-in elliptic curves domain parameters that are available in FIPS mode.

Name(s)	Size	Defined in:
NIST-P192 / secp192r1	192	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-P224 / secp224r1	224	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-P256 / secp256r1	256	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-P384 / secp384r1	384	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-P521 / secp521r1	521	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-K163 / sect163k1	163	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-B163 / sect163r2	163	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-K233 / sect233k1	233	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-B233 / sect223r1	233	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-K283 / sect283k1	283	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-B283 / sect283r1	283	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-K409 / sect409k1	409	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-B409 / sect409r1	409	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-K571 / sect571k1	571	[FIPS 186-2], [ANSI-X9.62], [SEC2]
NIST-B571 / sect571r1	571	[FIPS 186-2], [ANSI-X9.62], [SEC2]

12.2 Audit record

The audit record REC stores the log message of the CA-Core.

Elements	Description
$TRAIL_{SEQ}$	Sequence number of the trail to which the record belongs to
$TRAIL\_REC_{SEQ}$	Sequence number of the record inside this trail.
Timestamp	Creation date of this record
EventType	Event type of the log message
Module	Identifier of the module who creates this entry
UserIdentity	Identifier of the user who issues this entry
Message	This element stores the log message
$MAC_{TRK\_n}(TRAIL\_RECORD)$	MAC over this log record generated with current trail record key (TRK)

12.3 Trail header

Elements	Description
$TRAIL_{SEQ}$	Sequence number of the trail
Timestamp	Creation date of the trail
$ENC_{AEKpub}(TRK)$	Encrypted initial trail record key. The initial trail record key will generated when the audit system generates a new trail. This key will be encrypted by the audit encryption key
$SIG_{ASKpriv}(TRAIL_{SEQ} \mid Timestamp \mid ENC_{AEKpub}(TRK))$	Signature over the complete trail header information.

12.4 Trail footer

Elements	Description
$TRAIL_{SEQ}$	Sequence number of the trail
$TRAIL\_REC_{SEQ}$	The sequence number of the last trail record inside the trail
$SIG_{ASKpriv}(TRAIL\_SEQ \mid TRAIL\_REC_{SEQ} + 1 \mid MAC_{TRK}(TRAIL\_REC_{SEQ} + 1) \mid MAC_{TRK}(ENC_{AEKpub}(TRK)))$	Signature over the complete audit trail footer. This signature will generated when a new trail will created by the audit system
$MAC_{TRK}(TRAIL\_REC_{SEQ})$	MAC over the next serial number of the trail record, generated with the next generate trail record key (TRK)
$MAC_{TRK}(ENC_{AEKpub}(TRK))$	This Element stores a MAC over the exported initial trail record key of the next audit trail. This field will be filled when the audit system generates the next audit trail

13 ACRONYMS

ANSI	American National Standards Institute
API	Application Programming Interface
BAT	Batch File
CA	Certification Authority
CC	Evaluation Criteria for Information Technology Security (Common Criteria)
CIMC	Certificate Issuing and Management Component
CIMS	Certificate Issuing and Management System
CMS	Certificate Management System
CP	Certificate Policy
CPS	Certification Practices Statement
CRL	Certificate Revocation List
EAL	Evaluation Assurance Level
I&A	Identification and Authentication
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
IT	Information Technology
ITU	International Telecommunication Union
ITU-T	ITU Telecommunication Standardization Sector
JAR	Java Archive
KRA	Key Archival and Retrieval Authority
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PKI	Public Key Infrastructure
POP	Proof of Possession
PP	Protection Profile
RA	Registration Authority
SFP	Security Function Policy
SSL	Secure Socket Layer
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functions
TSP	TOE Security Policy

14 References

[BSI TR-03110]	BSI, Technical Guideline Advanced Security Mechanisms for Machine Readable Travel Documents, Part 1-3, Version 2.10, 20. March 2012, www.bsi.de
[CIMC PP]	Certificate Issuing and Management Component", Version 1.5, 11th August 2011, www.commoncriteriaportal.org
[FIPS140-2]	FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, FIPS PUB 140-2, SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES, NIST, 12.03.2002, www.nist.gov
[FIPS 186-2]	FIPS PUB 186-2: Digital Signature Standard (DSS) / National Institute of Standards and Technology (NIST), January 2000
[ITU-T X.509]	ITU, X.509 : Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks, www.itu.int
[RFC2104]	RFC2104, HMAC: Keyed-Hashing for Message Authentication, February 1997, http://tools.ietf.org/html/rfc2104
[RFC5280]	RFC5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008, http://tools.ietf.org/html/rfc5280
[RFC6960]	RFC6960, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP, June 2013, http://tools.ietf.org/html/rfc6960
[TR-03111]	TR-03111 Technical Guideline TR-03111 – Elliptic Curve Cryptography,; Version 1.11, April 2009 / Bundesamt für Sicherheit in der Informationstechnik (BSI)