Certificate Issuing and Management Components Protection Profile

Version 1.5

11 August, 2011

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

The Certificate Issuing and Management Components (CIMC) Protection Profile (PP) defines requirements for components that issue, revoke, and manage public key certificates, such as X.509 public key certificates.

This document was derived from the *Certificate Issuing and Management Components Family of Protection Profiles, Version 1.0, October 31, 2001* (CIMC PP). More specifically, this PP has essentially adopted the security functional requirements for Security Level 3 in that PP, while revising the assurance requirements to conform with Evaluation Assurance Level (EAL) 4 augmented with ALC_FLR.2, and also to update the content from Common Criteria (CC) version 2.1 to CC version 3.1 revision 2 (i.e., the current version).

The original CIMC PP was developed through a collaborative effort between the United States National Institute of Standards and Technology (NIST) and the United states National Security Agency (NSA) with the assistance and input of vendors.

1.1 Identification

Title:	Certificate Issuing and Management Components Protection Profile
PP Version:	Version 1.5, August 11, 2011
CC:	Part 2 extended, Part 3 conformant, EAL 4 augmented with ALC_FLR.2
CC Version:	CC version 3.1 revision 3 (CCv3.1r3)
Keywords:	Public Key Infrastructure, PKI, Certificate Issuing and Management Component, CIMC

1.2 Conformance Claims

A PP or Security Target (ST) claiming conformance to this PP is required to only meet "demonstrable" conformance.

1.3 Overview

Certificate Issuing and Management Components (CIMCs) may consist of one of more devices that are responsible for the issuance, revocation, and overall management of certificates and certificate status information. This CIMC Protection Profile (PP) defines the minimum security requirements for CIMCs. The requirements for FIPS 140-2 validated cryptographic modules and specific FIPS 140-2 levels are intended to provide additional assurance beyond those offered via Common Criteria (CC) evaluations.

1.4 Document Organization

The document is organized into the following sections:

- Section 1 includes the introductory material for the PP.
- Section 2 includes an overview of the components and operation of the CIMC.
- Section 3 includes a definition of the CIMC security problem. This section defines the threats that must be countered by the CIMC or through environmental controls.
- Section 4 defines the security objectives for the TOE and the environment.
- Section 5 defines the functional security requirements for the IT environment.

- Section 6 defines the functional security requirements for the TOE. Section 6 also contains the rationale for using functional security requirements that were not drawn from part 2 of the CC.
- Section 7 defines the assurance requirements for the TOE.
- Section 8 includes the rationale.
- Section 9 contains the Access Control Policies.
- Section 10 contains a Glossary of Terms.
- Section 11 contains a list of acronyms.

1.5 Conventions

With a few exceptions, the notation, formatting, and conventions used in this document are consistent with version 3.1 revision 3 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 a PP, 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.
- This PP contains some security functional requirements in which one or more operations, e.g., assignment and selection, have been left to the Security Target (ST) author to complete. Operations to be completed by the ST author are annotated between brackets by the words [ST assignment: …] or [ST selection: …]. In the case of an assignment, an explanation of how the operations may be completed is included in Italics within the brackets (e.g., [ST assignment: *other attributes*]). In the case of selection, a list of two or more elements from which the ST author may choose is included in Italics within the brackets (e.g., [ST selection: *the TSF, local users, remote users*]).
- Whenever a security functional requirement contains an operation that is to be completed by the ST author, an *Application Note* is provided immediately after the security requirement to clarify the assignment or selection (e.g., Application Note: The ST should specify the actions to be taken in case the verification fails).
- *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 CIMC PP were taken from 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_BKP.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 in section 9 of the document.

2 TOE DESCRIPTION

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, a PKI must perform two basic tasks:

1) generate and distribute public key certificates to bind public keys to other information *after* validating the accuracy of the binding; and

2) maintain and distribute certificate status information for unexpired certificates.

Some aspects of these tasks are relevant to the trustworthiness of the PKI. Other aspects affect the availability and performance of the PKI. The core tasks of the PKI are binding public keys to accurate information in a digitally signed certificate, and maintaining accurate certificate status information. If the components that implement these core tasks are implemented poorly, the PKI itself may be compromised. The distribution of certificates and status information affects the utility and performance of a PKI. If the components that handle distribution are compromised, denial of service may result, but the trustworthiness of the PKI is unaffected.

A PKI may also maintain user private keys for backup and recovery. This function is needed to meet the requirement for access to encrypted data even if private keys are lost. This function is orthogonal to the main goals of a PKI (distribution of public keys), but may undermine the trustworthiness of a PKI if implemented insecurely.

A monolithic PKI component could be designed to satisfy all of these requirements, but this is not a requirement. For scalability, PKIs are usually implemented with a set of complementary components, each focused on specific aspects of the PKI process. The PKI tasks are often assigned to the following logical components:

- *certification authorities* (CAs) to generate certificates and certificate status information;
- *registration authorities* (RAs) to verify the information in the public key certificates and determine certificate status;
- *repositories* to distribute certificates and certificate revocation lists (CRLs);
- *Online Certificate Status Protocol* (OCSP) *servers* to distribute certificate status information in the form of OCSP responses; and
- *key recovery servers* and *roaming credential servers* to backup or distribute private key material.

A particular PKI implementation must include the functionality of CAs and RAs, but the requirements may be assigned to any number of components. The features provided by repositories, OCSP servers, key recovery servers, and roaming credential servers are optional in a PKI implementation.

Certificate Issuing Management System (CIMS)

The basic building block of a PKI is the CA. PKIs are constructed by establishing trust relationships between CAs. However, the trustworthiness of the PKI is not a function of the CAs alone. The trustworthiness of a PKI depends on how the core tasks of the PKI are performed. This depends upon additional components: the RAs that validate the information that CAs place in certificates; the personnel and procedures involved in the operation of the CAs and RAs; and the physical controls provided by the environment in which the CAs and RAs are located. The aggregation of a CA, additional components performing core tasks, and the personnel and procedures in their operation are defined for this document as a Certificate Issuing and Management System (CIMS).

As noted above, the CIMS may be implemented as a single component, or a set of components. The central component of every CIMS is a CA. The CA issues certificates; in most cases, the CA also issues CRLs to distribute certificate status information. CIMS also validates the information to be placed in certificates and keeps track of changes in certificate status. These tasks are not generally included in a CA; most CIMS include RAs to validate information and inform the CA of changes in certificate status.

OCSP servers, key recovery servers, and roaming credential servers *may* be included in a CIMS in special cases. When a CA does not issue CRLs, the CIMS must include some other mechanism to distribute certificate status information. OCSP is the most popular status mechanism other than CRLs. In that case, an OCSP server may be part of the CIMS as well. If the CA stores private key material associated with users of the PKI, the security of that storage must be maintained, or the CA has compromised the binding represented in the corresponding certificate. In that case, the storage and protection of the private key material is also considered within the CIMS.

Note, however, that these services may be offered independently of a CA. OCSP servers are often implemented to retrieve and process CRLs from a repository. Users of a PKI may disclose their private keys to another server to facilitate roaming or recovery of encrypted data after private keys are lost. If private keys are stored by a component outside the CIMS, then the operation of this service will have no impact on the security of the CIMS. So, this PP does not make any assumptions nor do they impose any requirements on key recovery servers or roaming credential servers that may be implemented outside of the CIMS.

It should also be noted that repositories, which distribute certificates and CRLs, are not part of a CIMS. As noted above, the distribution of certificates and CRLs affects the availability and performance of a PKI, but does not affect its trustworthiness. In fact, in special cases, PKIs may omit the repository entirely and rely on users to distribute certificates. This PP does not make any assumptions nor do they impose any requirements on repositories.

Certificate Issuing Management Component (CIMC)

A Certificate Issuing and Management Component (CIMC) consists of the hardware, software, and firmware that are responsible for performing the functions of a CIMS. A CIMC does not include the environmental controls (e.g., controlled access facility, temperature), policies and procedures, personnel controls (e.g., background checks and security clearances), and other administrative controls that complete a CIMS.

This specification describes functional and assurance requirements for a CIMC. This specification makes no assumptions regarding the number of components in a CIMC or the functionality implemented by any particular component. The requirements specified apply to the CIMC as a whole. This specification does make certain assumptions regarding the administrative controls that must be in place for the proper operation of the CIMS.

CIMCs in a Sample PKI

Figure 1 presents a high-level diagram of a sample PKI with multiple CAs. The sample PKI consists of three CAs. Each CA is the central component in a CIMS, which issues, revokes, and manages certificates and certificate status information for a community of users. The corresponding CIMCs include different components.

The CAs in this PKI have issued certificates to each other to enable the formation of trust relationships between communities of users. These trust relationships are depicted by solid lines. In this case, the sample PKI forms a mesh or network PKI. However, the sample PKI could have been any other architecture (e.g., a hierarchical PKI). The core PKI tasks performed by a CIMS are unaffected by the architecture of its PKI.

Dashed lines represent information flows between components of the PKI. Where those lines are inside the CIMC boundary, or cross the CIMC boundary, there are issues that may impact the trustworthiness of a PKI. This specification includes requirements for the integrity and confidentiality of these information flows.

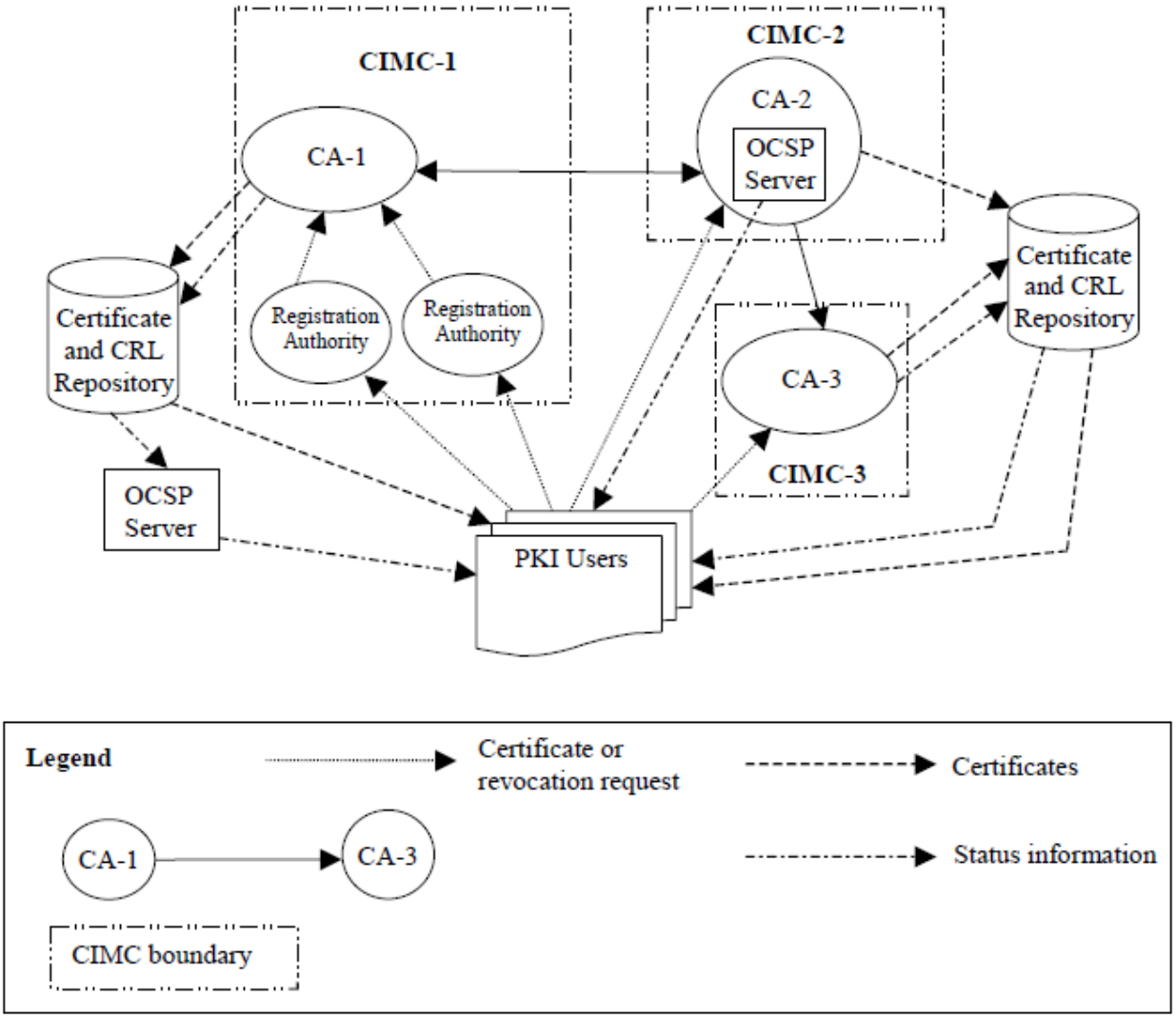

Figure 1 A Public Key Infrastructure (PKI) with Three CAs

CIMC-1 consists of CA-1 and two RAs. Users direct their certification and revocation requests to one of the RAs, which verify the information in the requests before forwarding them to CA-1. CA-1 posts the certificates and CRLs that it issues to a repository for retrieval by users of the PKI. An OCSP server retrieves CA-1's CRLs from the repository and uses the information to provide certificate status information in an alternate format. The OCSP server operates independently of CIMC-1. In fact, the operators of CIMC-1 may not even be aware that an OCSP server is offering this service.

CIMC-2 consists of a single component, CA-2, which provides the functionality of a CA, RA, and an OCSP server. Users direct their certification and revocation requests to CA-2 directly, which performs all validation processes itself. CA-2 distributes certificates through a repository, which is outside the CIMC-2 boundary. CA-2 does not generate CRLs; the *only* source of revocation information is the embedded OCSP server. Consequently, in CIMC-2, the implementation of the OCSP server is within the CIMC boundary.

CIMC-3 consists of one component, CA-3, that performs all the core PKI tasks. Users direct their certification and revocation requests to CA-3 directly, which performs all validation processes itself. CA-3 distributes certificates and CRLs through a repository, which is outside the CIMC-3 boundary. Since CA-3 does *not* delegate any of the core PKI tasks to other components, CIMC-3 includes only CA-3.

Logical Functions of a CIMC

Figure 2 illustrates the boundary of a CIMC in terms of the logical PKI components along with the environment in which they are used. In Figure 2, the boxes labeled CA and RA implement the base, required functionality of a CIMC. This includes the functionality required to issue and revoke public key certificates as well as the security functionality summarized in section 2.1. This specification requires that a CIMC be able to export certificate status information, but does not mandate the use of any particular method.

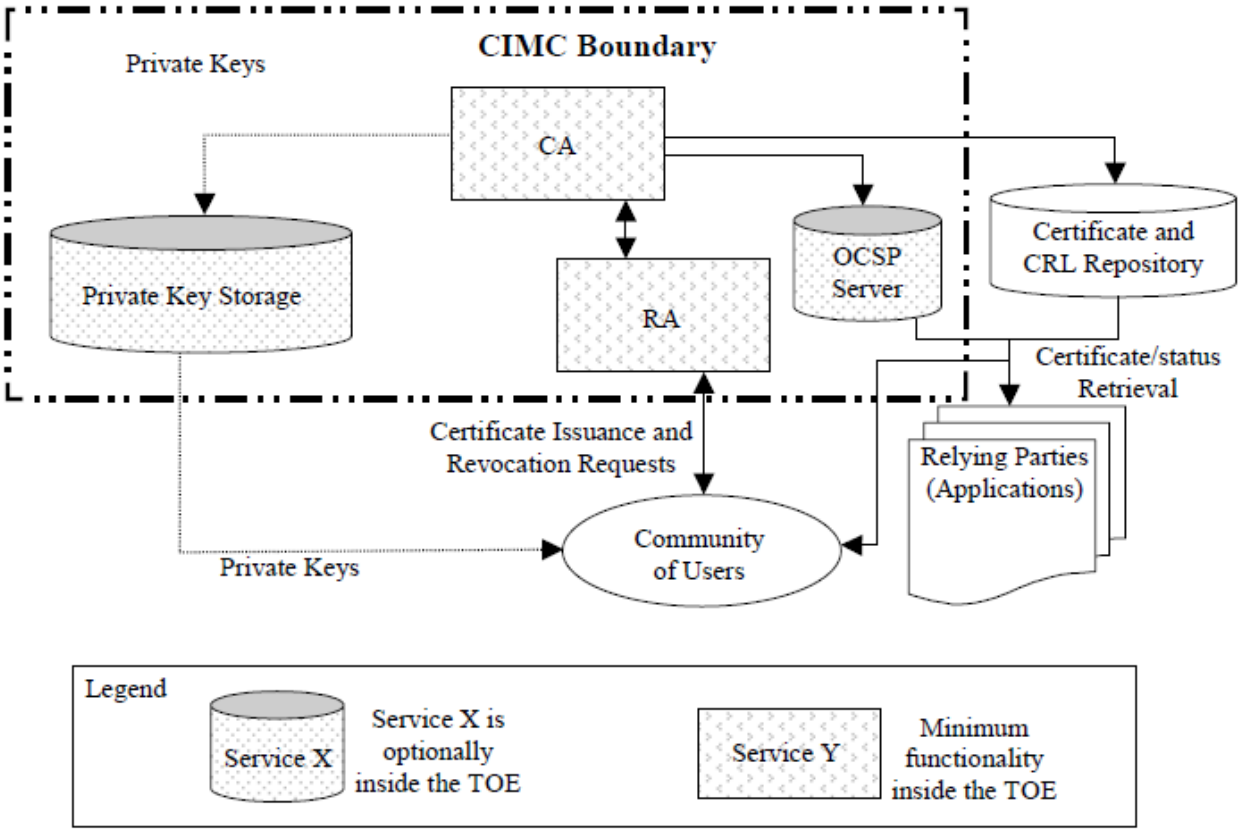

Figure 2 CIMC functionality and PKI Components

The two most popular, standardized methods for providing certificate status information are the issuance of CRLs and the use of a server that implements OCSP. In some PKIs, CAs issue CRLs, which are used by relying parties to determine the status of certificates (e.g., CA-1 and CA-3 in Figure 1). In other PKIs, CAs do not issue CRLs, but act as OCSP servers and provide certificate status information to relying parties by responding to OCSP requests (e.g., CA-2 in Figure 1). Where the CA acts as an OCSP server, this service is within the boundary of the CIMC.

In many PKIs where OCSP is used, however, the OCSP server is not part of the CIMC. As was the case with CA-1 in Figure 1, the CA issues CRLs, which are posted to a repository. An OCSP server retrieves the CRLs from the repository and uses the information contained in the CRLs to respond to relying parties' requests for certificate status information. If a component of a PKI, other than a CIMC, implements an OCSP server then the implementation of that server will have no impact on the security of the CIMC. So, this PP does not make any assumptions nor does it impose any requirements on OCSP servers that may be implemented outside of the CIMC.

Additional services that *may* be provided by a PKI include key recovery and roaming credential servers. Some CIMC vendors offer products that provide this functionality in addition to the base CA/RA functionality of issuing and revoking certificates. CIMCs that offer either of these services must store certificate subject private keys. Since it is vital to the security of a PKI that certificate subject private keys be maintained in a secure manner, this PP

imposes security requirements on the storage and handling of any certificate subject private keys held within the CIMC.

The environment of a CIMC includes a community of users and relying parties. The community of users are the people or systems that obtain certificates containing their own public keys from the CIMC. Relying parties use certificates and certificate status information to establish security services. (Note that most PKI users are also relying parties.) PKIs usually also include one or more repositories to which CAs post the certificates and CRLs (if used) that they generate. The trustworthiness of a CIMC is not dependent upon the actions of PKI users, relying parties, or repositories. This PP imposes no security requirements on and makes no assumptions about these components of the environment.

Target of Evaluation

Even though the functionality of a CIMC *may* be implemented by more than one physical component, this PP specifies functional and assurance security requirements for a CIMC as a whole and does not attempt to separate requirements by subcomponent. The intent of this PP is to ensure specification of the complete set of requirements for a CIMC and not the specification of a subset of requirements implemented in a specific CIMC component. It includes all the technical features of a CIMC, regardless of which CIMC component performs the function.

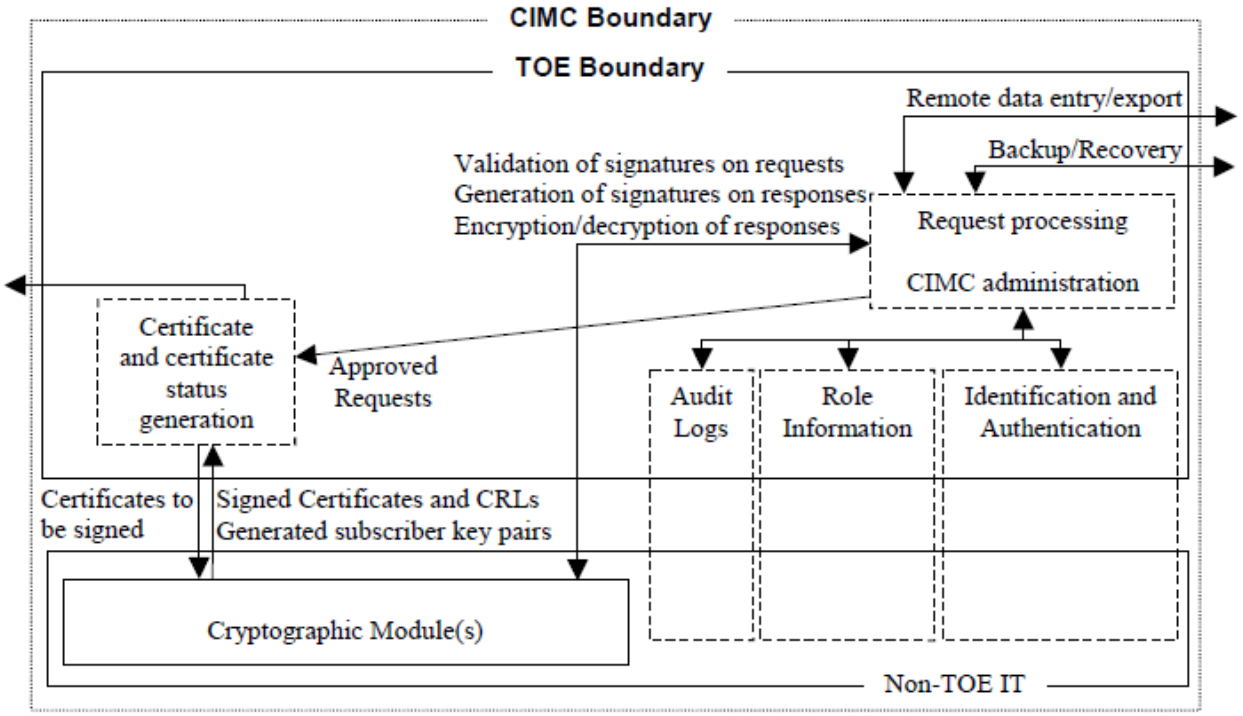

Figure 3 CIMC and Target of Evaluation

Considering all the components of a CIMC as a single entity assists in ensuring that the components compliant with the security requirements in this document will operate in a secure manner. This approach also ensures compatibility because a single vendor (or integrator) typically develops (or bundles) all the components together as a single solution. Typically, this is consistent with the way products are currently designed and built. A single product solution may make purchasing decisions easier because the user (or procurer) will not need to select components that meet a subset of the requirements. Finally, a single solution approach promotes security because the CIMC must:

- Implement all the mandatory security requirements, regardless of how they are allocated to components, and

- Ensure that functions implemented in one component do not compromise the security functions implemented in other components.

The scope of the CIMC is depicted in functional terms in Figure 3. *Please note;* this figure is not intended to show a particular architecture but to show, at a high level, how the functional requirements specified in this document may be met. As is shown in Figure 3, the functions of a CIMC may be divided into three categories: (1) functions that are performed by the TOE; (2) cryptographic functions, which must be performed within FIPS 140-2 validated cryptographic modules; and (3) non-cryptographic functions that may be performed by either the TOE or the environment. Security Requirements for category (1) functions are specified in Section 6, TOE Security Functional Requirements, and must be implemented within the TOE. Security requirements for category (2) and (3) functions are specified in Section 5, Security Requirements for the IT Environment. Security Targets (STs) claiming conformance to this PP may allocate these security requirement to the TOE, the environment, or some combination of both.

Category (1) functions include PKI-specific operations, such as the generation of certificates, and must be implemented within the TOE. Category (2) functions encompass implementation of cryptographic algorithms and protection of the CIMC's private keys. Category (3) functions include operations that are often supported by operating systems (e.g., identification and authentication) or supporting applications (e.g., database management).

Some functions, such as auditing, may be divided between the TOE and the environment. For example, the TOE may be implemented as a software program that runs on top of a general-purpose operating system. In such implementations, the operating system may be used to maintain audit logs, even though the operating system lies outside the boundary of the TOE. The TOE, however, is responsible for ensuring that certain auditable events, such as the generation of a certificate, are recorded in the audit logs.

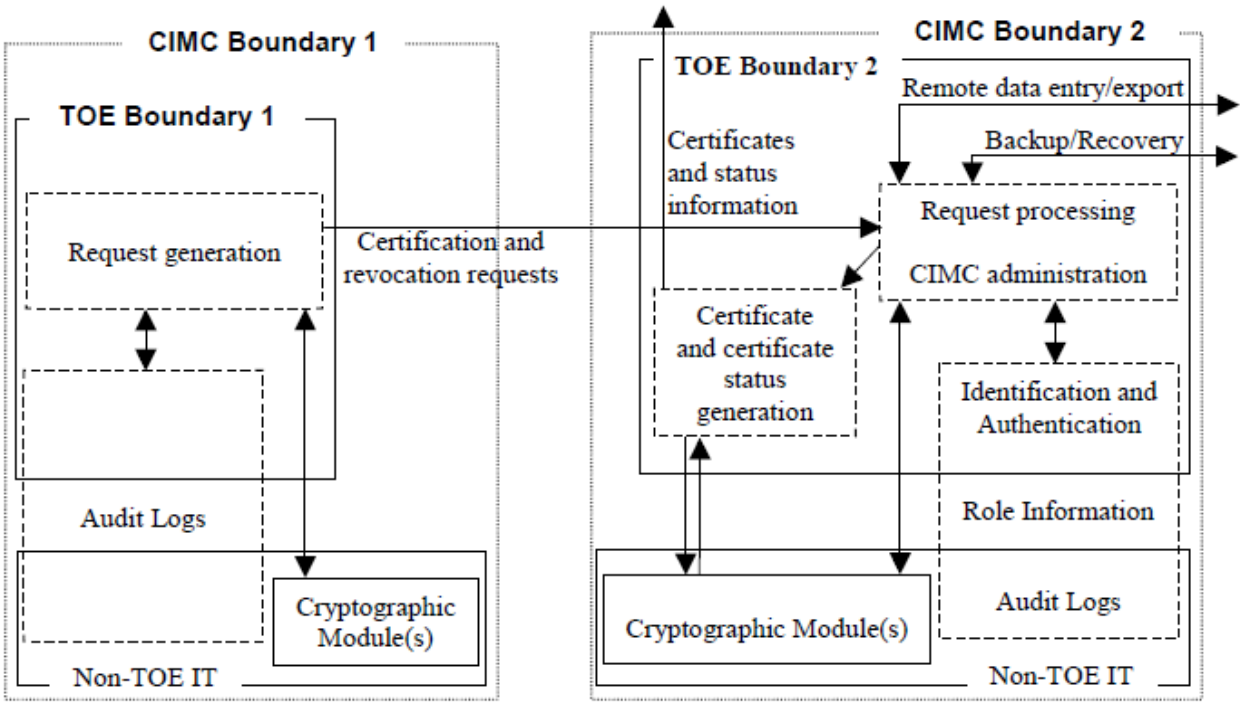

Figure 4 Multi-component implementation of a CIMC

Figure 3 could also be viewed as a illustration of a CIMC in which all functionality is performed by a single physical component. As is depicted in Figure 4, however, a common alternative is to implement a CIMC as a CA and one or more RAs where the CA and the RAs are physically separate components that RAs is not standardized, Figure 4 presents one possibility. In Figure 4, most of the work of the CIMC is performed by the CA (within CIMC Boundary 2). The CA generates all certificates and certificate status information, performs all required backups, maintains role information, and creates most of the required audit log entries. The RA(s) (within CIMC Boundary 1)

are used by Officers to create the certification and revocation requests that are processed by the CA. Since certification and revocation requests must be protected from modification when they are transmitted from the RA(s) to the CA, each RA must have its own cryptographic module. Since the RA(s) are responsible for identifying and authenticating Officers, any auditing related that functionality is performed by the RA.

In this case, neither the CA nor the RA contains all the basic, required functionality and does not form a complete CIMC. The CA and RA must be evaluated together so that all required functionality is present. The TOE includes functions implemented in both the RA and the CA; the environment includes the non-TOE IT in both the CA and RA components. In particular, note that the CA and RA have distinct cryptographic modules in Figure 4. Both of these modules are relevant to the security of the CIMC and this specification imposes requirements on both modules.

2.1 TOE Security Functionality

A CIMC compliant with this PP will provide the following security functionality:

- 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.
- Cryptographic Support (FCS) employs cryptographic functionality and addresses key management and the operational use of cryptographic keys.
- User Data Protection (FDP) relates to the protection of user data including certificate issuance, revocation, backup and recovery, and profile management of certificates, Certificate Revocation List (CRL), and Online Certificate Status Protocol (OCSP).
- Identification and Authentication (FIA) supports the administration and enforcement of the CIMC access control policies to unambiguously identify the person and/or entity performing functions in a CIMC.
- Security Management (FMT) specifies several aspects of management of security functions including distinct roles to maintain the security of the CIMC.
- Protection of the TOE Security Functions (FPT) supports functions that manage and protect the integrity of confidential TSF data from disclosure and modification through the use of encryption, reliable time stamps, backup and recovery procedures, self-tests and audit logs.
- Trusted Path/Channels (FTP) provides protection from modification and disclosure of transmitted data by means of a secure communications path between the CIMC and local and remote users.

As was described earlier, some of this functionality is provided by the TOE, while other functionality may be provided by the non-TOE (IT) environment.

2.2 CIMC Intended Environment

CIMCs will be operated in a wide variety of environments, from a closed secure facility to an open access facility in a hostile environment. Also, 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.

CIMCs designed to meet this PP may be appropriate for environments where risks and consequences of data disclosure and loss of data integrity are moderate to low. This PP requires integrity controls to ensure data is not modified and includes additional assurance requirements to ensure the CIMC is functioning securely.

A CIMC conforming to this PP provides some protection against malicious authorized users by requiring, at a minimum, three distinct roles. One role will be responsible for account administration, key generation, and audit configuration; a second role will be responsible for issuing and revoking certificates; and a third role responsible for maintaining the audit logs. This PP requires two-party control of private key export and additional auditing of import and export of secret and private keys and requests for information. 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 PP, 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.

2.3 CIMC Keys

It is essential that private and secret keys in CIMCs be managed securely. For the purposes of this document, keys are separated into three categories based on the individual or device that is authorized to use the key:

1) *CIMS personnel keys*: Private and secret keys used within a CIMC designated for use by individual identities. CIMS personnel keys may be used for authentication, to sign information contained within or output by a CIMC, or to encrypt information files.
2) *Component keys*: Keys, other than CIMS personnel keys, which are used by the CIMC. CIMCs shall use Component keys to sign certificates and certificate status information. Component public/private key pairs may also be used in key agreements, for signing audit logs and system backups and for ensuring the integrity of transmitted or stored data. Component secret keys may be used to encrypt CIMC stored or transmitted data and to compute authentication codes.
3) *Certificate subject private keys*: Private keys corresponding to the public keys contained in certificates issued by the CIMC where:
    - the private key is held by the CIMC solely to enable key recovery; or
    - the CIMC generates a public/private key pair and the private key is only held by the CIMC until the certificate subject has received it.

2.3.1 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.

1) *Certificate and Status Signing Keys*: Private keys used to sign certificates, CRLs, or other statements about the status of certificates. [Component keys]
2) *Integrity or Approval Authentication Keys*: Private or secret keys used to protect the integrity of transactions between CIMCs or CIMC subcomponents. Private or secret keys used to authenticate transactions between CIMCs that cause or approve the issuance or revocation of certificates. [CIMS personnel keys, Component keys]
3) *General Authentication Keys*: Private or secret keys used to authenticate users, messages, or sessions that do not include the authorization or approval of certificate issuance or revocation, but may include requests to issue or revoke certificates. [CIMS personnel keys, Component keys]
4) *Long Term Private Key Protection Keys*: Secret or private keys that are used to protect private keying material that is used for multiple sessions or messages. [CIMS personnel keys, Component keys]
5) *Long Term Confidentiality Keys*: Secret keys that are used to protect the confidentiality of security-relevant information such as PINS or passwords. This information does not include private keying material. [CIMS personnel keys, Component keys]
6) *Short Term Private Key Protection Keys*: Private keys used to protect keying material for a single session or message. [CIMS personnel keys, Component keys]
7) *Short Term Confidentiality Keys*: Secret keys used to protect a single session or message that does not contain keying material. [CIMS personnel keys, Component keys]

2.4 Data Input

A CIMC may receive information in many different ways. Data input is organized in the following three categories depending on the source of the data (local or remote) and whether the user is authenticated by the CIMC.

1) *Unauthenticated Data Entry:* The message/data may either be entered locally or received over a network. The originator of the message/data cannot be verified, i.e., the user is unauthenticated.
2) *Local Data Entry:* A user, operating locally, enters or accepts data so that the CIMC can associate the data with the user and list the user in the audit log with the accepted data. The data entry could take the form of a user vouching for information that has already been entered into the computer by clicking on an "accept" button or by otherwise indicating acceptance of the information.
3) *Remote Data Entry:* The data could be received over a network in such a way that it can be bound to the identity of the sender of the data (or to the identity of some other remote user). For example, the data could be sent in a signed email.

2.5 Trusted Public Key Entry, Deletion, and Storage

In addition to issuing public key certificates, CIMCs may use public keys for their own purposes. Specifically, a CIMC may use the public key of another entity to encrypt messages that it intends to send to that entity, authenticate messages that it receives from that entity, or perform a key agreement to establish a session key for communicating with that entity.

A public key may be trusted by a CIMC because it is contained in a certificate that was issued by a CA that the CIMC trusts. At the next level, trust in the public key used to verify the signature on that certificate must be established. Trust in this public key may be established by another certificate. This trust validation *path* will continue until the final (or root) public key is reached. In order to bootstrap the process at the root public key, a CIMC must establish trust in this public key through some means other than certificate path processing. While the signatures on public key certificates authenticate and protect most public keys, a digital signature does not protect these public key "trust anchors". Also, these public keys must be protected from modification.

Every CIMC that uses public keys for authentication, encryption, integrity, or access control will maintain a list of trusted public keys. This list may include several keys (e.g., one for each authorized user) or may include only one key, which can be used to verify trust in all other public keys through path validation.

3 SECURITY PROBLEM DEFINITION

This section includes the following:

- Secure usage assumptions,
- Threats, and
- Organizational security policies.

This information provides the basis for the Security Objectives specified in Section 4, the security functional requirements for the TOE and environment specified in Sections 5 and 6, and the TOE Security Assurance Requirements specified in Section 7.

3.1 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, Operators, Officers and Auditors	Competent Administrators, Operators, Officers and Auditors will be assigned to manage the TOE and the security of the information it contains.
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, Operators, 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, Operators, 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, operators, officers and auditors are trained in techniques to thwart social engineering attacks.

Connectivity

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 PP.

Although this PP does not specifically address the operating system, functions/requirements traditionally attributed to an operating system are distributed throughout this PP 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 PP.

Physical

A.Communications Protection	The system is adequately physically protected against loss of communications i.e., availability of communications.
A.Physical Protection	The TOE hardware, software, and firmware critical to security policy enforcement will be protected from unauthorized physical modification.

3.2 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, CRLs, or OCSP Responses. The latter systems are termed relying party systems.

T.Administrative errors of omission	Administrators, Operators, Officers or Auditors fail to perform some function essential to security.
T.Administrators, Operators, Officers and Auditors commit errors or hostile actions	An Administrator, Operator, 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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.

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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.
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, CRLs, or OCSP Responses.

3.3 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.

4 SECURITY OBJECTIVES

This section includes the security objectives for the CIMC PP including security objectives for the TOE, security objectives for the environment, and security objectives for both the TOE and environment.

4.1 Security Objectives for the TOE

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

Authorized Users

O.Certificates	The TSF must ensure that certificates, certificate revocation lists, and certificate status information are valid.

System

O.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.

Cryptography

O.Non-repudiation	Prevent user from avoiding accountability for sending a message by providing evidence that the user sent the message.

External Attacks

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

4.2 Security Objectives for the Environment

This section specifies the security objectives for the environment.

O.Administrators, Operators, Officers and Auditors guidance documentation	Deter Administrator, Operator, Officer or Auditor errors by providing adequate documentation on securely configuring and operating the CIMC.
O.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.
O.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.)
O.Communications Protection	Protect the system against a physical attack on the communications capability by providing adequate physical security.
O.Competent Administrators, Operators, Officers and Auditors	Provide capable management of the TOE by assigning competent Administrators, Operators, Officers and Auditors to manage the TOE and the security of the information it contains.

O.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.
O.CPS	All Administrators, Operators, Officers and Auditors shall be familiar with the certificate policy (CP) and the certification practices statement (CPS) under which the TOE is operated.
O.Cryptographic functions	The TOE must implement approved cryptographic algorithms for encryption/decryption, authentication, and signature generation/verification; approved key generation techniques and use validated cryptographic modules. (Validated is defined as FIPS 140-2 validated.)
O.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).
O.Installation	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which maintains IT security.
O.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.
O.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.
O.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.
O.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.
O.Periodically check integrity	Provide periodic integrity checks on both system and software.
O.Physical Protection	Those responsible for the TOE must ensure that the security-relevant components of the TOE are protected from physical attack that might compromise IT security.
O.Repair identified security flaws	The vendor repairs security flaws that have been identified by a user.
O.Security roles	Maintain security-relevant roles and the association of users with those roles.
O.Social Engineering Training	Provide training for general users, Administrators, Operators, Officers and Auditors in techniques to thwart social engineering attacks.
O.Sufficient backup storage and effective restoration	Provide sufficient backup storage and effective restoration to ensure that the system can be recreated.
O.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.
O.Validation of security function	Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.

4.3 Security Objectives for both the TOE and the Environment

This section specifies the security objectives that are jointly addressed by the TOE and the environment.

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.
O.Data import/export	Protect data assets when they are being transmitted to and from the TOE, either through intervening untrusted components or directly to/from human users.
O.Detect modifications of firmware, software, and backup data	Provide integrity protection to detect modifications to firmware, software, and backup data.
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.Integrity protection of user data and software	Provide appropriate integrity protection for user data and software.
O.Limitation of administrative access	Design administrative functions so that Administrators, Operators, Officers and Auditors do not automatically have access to user objects, except for necessary exceptions. Control access to the system by Operators and 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.Manage behavior of security functions	Provide management functions to configure, operate, and maintain the security mechanisms.
O.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.
O.Procedures for preventing malicious code	Incorporate malicious code prevention procedures and mechanisms.
O.Protect stored audit records	Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.
O.Protect user and TSF data during internal transfer	Ensure the integrity of user and TSF data transferred internally within the system.
O.React to detected attacks	Implement automated notification (or other responses) to the TSF-discovered attacks in an effort to identify attacks and to create an attack deterrent.
O.Require inspection for downloads	Require inspection of downloads/transfers.
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.
O.Restrict actions before authentication	Restrict the actions a user may perform before the TOE authenticates the identity of the user.
O.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.
O.Time stamps	Provide time stamps to ensure that the sequencing of events can be verified.
O.User authorization management	Manage and update user authorization and privilege data to ensure they are consistent with organizational security and personnel policies.

5 SECURITY REQUIREMENTS FOR THE IT ENVIRONMENT

This section specifies the security functional requirements that are applicable to the IT environment. While CCv3.1 does not require the specification of requirements for the IT environment, they are defined here for the benefit of the user instantiating the TOE in a suitable environment and also so that STs claiming conformance to this PP may specify the requirements in this section as security requirements for the TOE, if performed by the TOE.

Table 1 lists all the security functional requirements for the IT environment. They are listed in alphabetical order in Table 1 for ease of reference. Also included are the applicable CIMC PP section to which each requirement applies.

Table 1 CIMC IT Environment Functional Security Requirements

Security Functional Requirement (Component)	CIMC PP Section
FAU_GEN.1 Audit data generation (iteration 1)	5.1 Security Audit
FAU_GEN.2 User identity association (iteration 1)	5.1 Security Audit
FAU_SAR.1 Audit Review	5.1 Security Audit
FAU_SAR.3 Selectable audit review	5.1 Security Audit
FAU_SEL.1 Selective audit (iteration 1)	5.1 Security Audit
FAU_STG.1 Protected audit trail storage (iteration 1)	5.1 Security Audit
FAU_STG.4 Prevention of audit data loss (iteration 1)	5.1 Security Audit
FCS_CKM.1 Cryptographic key generation	5.7.1 Key Generation
FCS_CKM.4 Cryptographic key destruction	5.7.2 Private and Secret Key Destruction
FCS_COP.1 Cryptographic operation	5.9 Cryptographic Modules
FDP_ACC.1 Subset access control (iteration 1)	5.4 Access Control
FDP_ACF.1 Security attribute based access control (iteration 1)	5.4 Access Control
FDP_CIMC_BKP.1 CIMC backup and recovery	5.3 Backup and Recovery
FDP_CIMC_BKP.2 Extended CIMC backup and recovery	5.3 Backup and Recovery
FDP_ITT.1 Basic internal transfer protection (iterations 1 and 2)	5.6 Remote Data Entry and Export
FDP_UCT.1 Basic data exchange confidentiality (iteration 1)	5.6 Remote Data Entry and Export
FIA_AFL.1 Authentication failure handling	5.5 Identification and Authentication
FIA_ATD.1 User attribute definition	5.5 Identification and Authentication
FIA_SOS.1 Verification of secrets (iteration 1)	5.5 Identification and Authentication
FIA_UAU.1 Timing of authentication (iteration 1)	5.5 Identification and Authentication
FIA_UID.1 Timing of identification (iteration 1)	5.5 Identification and Authentication
FIA_USB.1 User-subject binding (iteration 1)	5.5 Identification and Authentication
FMT_MOF.1 Management of security functions behavior (iteration 1)	5.2 Roles
FMT_MSA.1 Management of security attributes	5.2 Roles
FMT_MSA.2 Secure security attributes	5.2 Roles
FMT_MSA.3 Static attribute initialization	5.2 Roles
FMT_MTD.1 Management of TSF data	5.2 Roles
FMT_SMR.2 Restrictions on security roles	5.2 Roles
FPT_TST_CIMC.1 Abstract Machine Testing	5.8 Self-tests
FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 1)	5.6 Remote Data Entry and Export
FPT_ITT.1 Basic internal TSF data transfer protection (iterations 1 and 2)	5.6 Remote Data Entry and Export
FPT_STM.1 Reliable time stamps (iteration 1)	5.1 Security Audit
FPT_TST_CIMC.2 Software/firmware integrity test	5.8 Self-tests
FPT_TST_CIMC.3 Software/firmware load test	5.8 Self-tests
FTP_TRP.1 Trusted path	5.5 Identification and Authentication

5.1 Security Audit

FAU_GEN.1 Audit data generation (iteration 1)

FAU_GEN.1.1 The IT environment 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 2 below.

FAU_GEN.1.2 The IT environment 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 2 below.
Additionally, the audit shall not include plaintext private or secret keys or other critical security parameters.

Table 2 Auditable Events and Audit Data

Section/Function	Component	Event	Additional Details
5.1: Security Audit	FAU_GEN.1 Audit data generation (iteration 1)	Any changes to the audit parameters, e.g., audit frequency, type of event audited	
		Any attempt to delete the audit log	
5.5: Identification and Authentication	FIA_ATD.1 User attribute definition	Successful and unsuccessful attempts to assume a role	
	FIA_AFL.1 Authentication failure handling	The value of maximum authentication attempts is changed	
	FIA_AFL.1 Authentication failure handling	*Maximum authentication attempts* unsuccessful authentication attempts occur during user login	
	FIA_AFL.1 Authentication failure handling	An Administrator unlocks an account that has been locked as a result of unsuccessful authentication attempts	
		An Administrator changes the type of authenticator, e.g., from password to biometrics	
5.2: Roles		Roles and users are added or deleted	
		The access control privileges of a user account or a role are modified	

FAU_GEN.2 User identity association (iteration 1)

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

FAU_SAR.1 Audit review

FAU_SAR.1.1 The IT environment shall provide Auditors with the capability to read all information from the audit records.

FAU_SAR.1.2 The IT environment shall provide the audit records in a manner suitable for the user to interpret the information.

FAU_SAR.3 Selectable audit review

FAU_SAR.3.1 The IT environment shall provide the ability to apply searches of audit data based on the type of event, the user responsible for causing the event, and as specified in Table 3 below.

Table 3 Audit Search Criteria

Section/Function	Search Criteria
Certificate Request Remote and Local Data Entry	Identity of the subject of the certificate being requested
Certificate Revocation Request Remote and Local Data Entry	Identity of the subject of the certificate to be revoked

FAU_SEL.1 Selective audit (iteration 1)

FAU_SEL.1.1 The IT environment 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) [ST selection: *object identity, user identity, subject identity, host identity, event type*]

b) [ST assignment: *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.

FAU_STG.1 Protected audit trail storage (iteration 1)

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

FAU_STG.1.2 The IT environment shall be able to detect unauthorised modifications to the stored audit records in the audit trail..

FAU_STG.4 Prevention of audit data loss (iteration 1)

FAU_STG.4.1 The IT environment shall prevent audited events, except those taken by the Auditor if the audit trail is full.

FPT_STM.1 Reliable time stamps (iteration 1)

FPT_STM.1.1 The IT environment shall be able to provide reliable time stamps.

5.2 Roles

The ability to perform many of the functions specified in this PP will be allocated to distinct roles to maintain the security of a CIMC. This subsection defines a set of roles that will be used throughout this document when allocating responsibilities.

A CIMC is not required to implement all of the roles listed, but is only required to implement roles to meet the role separation requirements. A single identity may be assigned multiple roles except where prohibited by the CIMC requirements. Multiple individuals may be assigned to a specific role, as required by the CIMC implementation.

The role definitions are listed below:

1) *Administrator* – role authorized to install, configure, and maintain the CIMC; establish and maintain user accounts; configure profiles and audit parameters; and generate Component keys.

2) *Operator* – role authorized to perform system backup and recovery.

3) *Officer* – role authorized to request or approve certificates or certificate revocations.

4) *Auditor* – role authorized to view and maintain audit logs.

It is important that one individual cannot perform all the functions specified for a CIMC. One mechanism to deter abuse of power is the separation of CA duties.

FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 The IT environment shall maintain the roles: Administrator, Auditor, and Officer.

FMT_SMR.2.2 The IT environment shall be able to associate users with roles.

FMT_SMR.2.3 The IT environment shall ensure that the conditions
a) no identity is authorized to assume both an Administrator and an Officer role;
b) no identity is authorized to assume both an Auditor and an Officer role; and
c) no identity is authorized to assume both an Administrator and an Auditor role
are satisfied.

NOTE: This document specifies four roles: Administrator, Auditor, Officer, and Operator. However, CIMCs are not required to maintain all four roles. If a CIMC does not implement one of the roles defined above (e.g., Operator), then the capabilities assigned to that role by this PP must be assigned to some other role or roles. In particular, CIMCs are not required to implement the Operator role. If a CIMC does not implement the Operator role, then each of the capabilities assigned to the Operator role by this PP must be assigned to one or more roles implemented by the CIMC.

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

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

Table 4 Authorized Roles for Management of Security Functions Behavior

Section/Function	Function/Authorized Role
5.1: Security Audit	The capability to configure the audit parameters shall be restricted to Administrators.
5.3: Backup and Recovery	The capability to configure the backup parameters shall be restricted to [ST assignment: *authorized user*][1]. The capability to initiate the backup or recovery function shall be restricted to [ST assignment: *authorized user*][2].
5.5: Identification and Authentication	The capability to specify or change *maximum authentication attempts* shall be restricted to Administrators. The capability to change authentication mechanisms shall be restricted to Administrators.
5.2: Roles	The capability to create user accounts and roles shall be restricted to Administrators. The capability to assign privileges to those accounts and roles shall be restricted to Administrators.

[1] Application Note: The ST author should specify one of roles, such as Administrator, defined in the ST.
[2] Application Note: The ST author should specify one of roles, such as Administrator, Operator, defined in the ST.

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to restrict the ability to modify the security attributes [ST assignment: *list of security attribute*s] to Administrators.

> Application Note: The ST must state components of the security attributes that may be modified and any restrictions that may exist for Administrators. The ST must state the components of the access rights that the Administrator is allowed to modify.

FMT_MSA.2 Secure security attributes

FMT_MSA.2.1 The IT environment shall ensure that only secure values are accepted for [ST assignment: *list of security attribute*s].

> Application Note: The list of security attributes should be the same as that in FMT.MSA.1.

FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to provide [ST selection, choose one of: *restrictive, permissive, [ST assignment: other property*]] default values for security attributes that are used to enforce the SFP.

> Application Note: The IT environment shall provide default values for relevant object security attributes, which can be overridden by an initial value. It may be possible for a new object to have different security attributes at creation, if a mechanism exists to specify the permissions at time of creation. The ST author should select whether the default property of the access control attribute will be restrictive, permissive, or another property. In case of another property, the ST author should assign the specific property.

FMT_MSA.3.2 The IT environment shall allow the Administrator to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1 Management of TSF data

FMT_MTD.1.1 The IT environment shall restrict the ability to view (read) or delete the audit logs to Auditors.

5.3 Backup and Recovery

Backup and recovery includes reconstructing a system in the event of a system failure or other serious error.

In order to be able to recover from failures and other unanticipated undesired events, CIMCs must be able to be backed up. The backup will be used to restore the CIMC to an operational status at a previous point in time. The frequency of performing backups (e.g., hourly, daily, or weekly) is based on the criticality of the application or system.

FDP_CIMC_BKP.1 CIMC backup and recovery

FDP_CIMC_BKP.1.1 The IT environment shall include a backup function.

FDP_CIMC_BKP.1.2 The IT environment shall provide the capability to invoke the backup function on demand.

FDP_CIMC_BKP.1.3 The data stored in the system backup shall be sufficient to recreate the state of the system at the time the backup was created using only:

a) a copy of the same version of the CIMC as was used to create the backup data;
b) a stored copy of the backup data;

c) the cryptographic key(s), if any, needed to verify the digital signature, keyed hash, or authentication code protecting the backup; and
d) the cryptographic key(s), if any, needed to decrypt any encrypted critical security parameters.

FDP_CIMC_BKP.1.4 The IT environment shall include a recovery function that is able to restore the state of the system from a backup. In restoring the state of the system, the recovery function is only required to create an "equivalent" system state in which information about all relevant CIMC transactions has been maintained.

Dependencies: FMT_MOF.1 Management of security functions behavior

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 objectives O.Object and data recovery free from malicious code and O.Preservation/trusted recovery of secure state.

FDP_CIMC_BKP.2 Extended CIMC backup and recovery

FDP_CIMC_BKP.2.1 The backup data shall be protected against modification through the use of digital signatures, keyed hashes, or authentication codes.

FDP_CIMC_BKP.2.2 Critical security parameters and other confidential information shall be stored in encrypted form only.

Dependencies: FDP_CIMC_BKP.1 CIMC backup and recovery

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 objectives O.Object and data recovery free from malicious code and O.Preservation/trusted recovery of secure state.

5.4 Access Control

FDP_ACC.1 Subset access control (iteration 1)

FDP_ACC.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 on [ST assignment: *list of subjects, objects, and operations 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.

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

FDP_ACF.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to objects based on the following: the identity of the subject and the set of roles that the subject is authorized to assume.

FDP_ACF.1.2 The IT environment shall enforce the following rule to determine if an operation among controlled subjects and controlled objects is allowed: the capability to zeroize plaintext private and secret keys shall be restricted to Administrators, Auditors, Officers, and Operators.

FDP_ACF.1.3 The IT environment shall explicitly authorize access of subjects to objects based on the following additional rules: [ST assignment: *rules, based on security attributes, that explicitly authorize access of subjects to object*s].

Application Note: The rules that govern the CIMC IT Environment Access Control Policy may vary between IT environments; 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.

FDP_ACF.1.4 The IT environment shall explicitly deny access of subjects to objects based on the following additional rules: [ST assignment: *rules, based on security attributes that explicitly deny access of subjects to object*s].

Application Note: The rules that govern the CIMC IT Environment Access Control Policy may vary between IT environments; 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.

5.5 Identification and authentication

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

FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 If authentication is not performed in a cryptographic module that has been FIPS 140-2 validated to an overall Level of 2 or higher with Level 3 or higher for Roles and Services, the IT environment shall detect when an Administrator configurable maximum limit for unsuccessful authentication attempts has occurred since the last successful authentication for the indicated user identity.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the IT environment shall [ST assignment: *list of actions to take when* maximum authentication attempts *unsuccessful authentication attempts have occurred*].

Application Note: The ST must either (a) specify that authentication is performed in a cryptographic module that has been FIPS 140-2 validated to an overall Level of 2 or higher with Level 3 or higher for Roles and Services or (b) specify the actions to be taken in case the threshold is met or surpassed.

Actions taken in case the threshold is met or surpassed could include disabling of an account for five minutes or disabling of the account until unlocked by the administrator and simultaneously informing the administrator. (In order to prevent a denial-of-service attack, accounts that belong to Administrators should not be disabled.) The actions should specify the measures and, if applicable, the duration of the measure (or the conditions under which the measure will be ended).

FIA_ATD.1 User attribute definition

FIA_ATD.1.1 The IT environment shall maintain the following list of security attributes belonging to individual users: the set of roles that the user is authorized to assume, [ST assignment: *other security attribute*s].

Application Note: The specified attributes are those that are required by the IT environment to enforce the CIMC IT Environment Access Control Policy, the generation of audit records, and proper identification and authentication of users. The user identity must be uniquely associated with a single individual user. Group membership may be expressed in a number of ways: a list per user specifying to which groups the user belongs, a list per group which includes which users are members, or implicit association between certain user identities and certain groups. The ST author should specify the security attributes that are associated with an individual user. An example of such a list is {'clearance, 'group identifier', 'rights'}.

FIA_SOS.1 Verification of secrets (iteration 1)

FIA_SOS.1.1 The IT environment 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 1)

FIA_UAU.1.1 The IT environment shall allow [ST assignment: *list of IT environment mediated actions that are not security relevant*] on behalf of the user to be performed before the user is authenticated.

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

FIA_UID.1 Timing of identification (iteration 1)

FIA_UID.1.1 The IT environment shall allow [ST assignment: *list of IT environment-mediated actions that are not security relevant*] on behalf of the user to be performed before the user is identified.

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

Application Note: FIA_UAU.1 and FIA_UID.1 allow the ST author to specify IT environment-mediated actions that may be performed on behalf of a user before that user is identified and/or authenticated. However, the IT environment 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 IT environment-mediated actions that may be performed on behalf of a user before that user is identified and/or authenticated include:

a) 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).

FIA_USB.1 User-subject binding (iteration 1)

FIA_USB.1.1 The IT environment shall associate the following user security attributes with subjects acting on the behalf of that user: roles, [ST assignment: *list of user security attributes]*.

FIA_USB.1.2 The IT environment shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [ST assignment: *rules for the initial association of attributes*].

FIA_USB.1.3 The IT environment shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [ST assignment: *rules for the changing of attributes*].

FTP_TRP.1 Trusted path

FTP_TRP.1.1 The IT environment shall provide a communication path between itself and [ST selection: *remote, local*] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2 The IT environment shall permit [ST selection: *the IT environment, the TSF, local users, remote user*s] to initiate communication via the trusted path.

FTP_TRP.1.3 The IT environment shall require the use of the trusted path for initial user authentication, [ST assignment: *other services for which trusted path is require*d].

> Application Note: The ST should identify other services for which a trusted path is required, if any. A trusted path may be required for any security-relevant interaction.

5.6 Remote Data Entry and Export

FDP_ITT.1 Basic internal transfer protection (iteration 1)

FDP_ITT.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to prevent the modification of security-relevant user data when it is transmitted between physically-separated parts of the IT environment.

FDP_ITT.1 Basic internal transfer protection (iteration 2)

FDP_ITT.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to prevent the disclosure of confidential user data when it is transmitted between physically-separated parts of the IT environment.

FDP_UCT.1 Basic data exchange confidentiality (iteration 1)

FDP_UCT.1.1 The IT environment shall enforce the CIMC IT Environment Access Control Policy specified in section 9.1 to transmit confidential user data in a manner protected from unauthorized disclosure.

FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 1)

FPT_ITC.1.1 The IT environment shall protect confidential IT environment data transmitted from the IT environment to a remote trusted IT product from unauthorized disclosure during transmission.

FPT_ITT.1 Basic internal TSF data transfer protection (iteration 1)

FPT_ITT.1.1 The IT environment shall protect security-relevant IT environment data from modification when it is transmitted between separate parts of the IT environment.

FPT_ITT.1 Basic internal TSF data transfer protection (iteration 2)

FPT_ITT.1.1 The IT environment shall protect confidential IT environment data from disclosure when it is transmitted between separate parts of the IT environment.

5.7 Key Management

5.7.1 Key Generation

This subsection specifies the requirements for the generation of cryptographic keys by the IT environment.

FCS_CKM.1 Cryptographic key generation

FCS_CKM.1.1 The FIPS 140-2 validated cryptographic module shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [ST assignment: *FIPS-approved or recommended cryptographic key generation algorithms*] and specified cryptographic key sizes [ST assignment: *cryptographic key sizes*] that meet the following: [ST assignment: *list of FIPS*].

5.7.2 Private and Secret Key Destruction

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

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The IT environment shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [ST assignment: *cryptographic key destruction metho*d] that meets the following: [ST assignment: *list of standard*s].

> Application Note: The ST should specify the key destruction method to be used to destroy cryptographic keys. The ST should specify the assigned standard that documents the method used to destroy cryptographic keys. The assigned standard may comprise none, one or more actual standards publications, for example, from international, national, industry or organizational standards.

5.8 Self-tests

FPT_TST_CIMC.1 Abstract Machine Testing

FPT_TST_CIMC.1.1 The IT environment shall run a suite of tests [ST selection: *during initial start-up, periodically during normal operation, at the request of an authorized user, other conditions*] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the IT environment.

FPT_TST_CIMC.1.2 The IT environment shall provide authorised users with the capability to verify the integrity of the security assumptions provided by the abstract machine that underlies the IT environment.

FPT_TST_CIMC.1.2 The IT environment shall provide authorised users with the capability to verify the integrity of stored TSF executable code.

> Application Note: The ST author should specify when the IT environment will execute the abstract machine testing. The ST author, through this selection, has the ability to indicate the frequency with which the self-tests will be run. If the tests are run often, then the end users should have more confidence that the IT environment is operating correctly then if the tests are run less frequently. However, this must be balanced with the potential impact on the availability of the IT environment.
>
> Dependencies: None.

Rationale: This component is necessary since the CC no longer includes a requirement to test required properties of the abstract machine that supports, but is not part of, the TSF. It satisfies the security objective O.Validation of security function and software and O.Periodically check integrity.

FPT_TST_CIMC.2 Software/firmware integrity test

FPT_TST_CIMC.2.1 An error detection code (EDC) or FIPS-approved or recommended authentication technique (e.g., the computation and verification of an authentication code, keyed hash, or digital signature algorithm) shall be applied to all security-relevant software and firmware residing within the CIMC (e.g., within EEPROM and RAM). The EDC shall be at least 16 bits in length.

FPT_TST_CIMC.2.2 The error detection code, authentication code, keyed hash, or digital signature shall be verified at power-up and on-demand. If verification fails, the IT environment shall [ST assignment: *action to be taken if the verification fails*].

Application Note: The ST should specify the actions to be taken if signature verification fails.

Dependencies: FPT_TST_CIMC.1 Abstract Machine testing.

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC. It satisfies the security objective O.Integrity protection of user data and software and O.Periodically check integrity.

FPT_TST_CIMC.3 Software/firmware load test

FPT_TST_CIMC.3.1 A cryptographic mechanism using a FIPS-approved or recommended authentication technique (e.g., an authentication code, keyed hash, or digital signature algorithm) shall be applied to all security-relevant software and firmware that can be externally loaded into the CIMC.

FPT_TST_CIMC.3.2 The IT environment shall verify the authentication code, keyed hash, or digital signature whenever the software or firmware is externally loaded into the CIMC. If verification fails, the IT environment shall [ST assignment: *action to be taken if the verification fails*].

Application Note: The ST should specify the action to be taken if the signature verification fails.

Dependencies: FPT_TST_CIMC.1 Abstract Machine Testing

Rationale: This component is necessary to specify a unique requirement for certificate issuing and management components that is not addressed by the CC. It satisfies the security objective O.Integrity protection of user data and software and O.Periodically check integrity.

5.9 Cryptographic Modules

In many cases, a CIMC may use a single cryptographic module to perform all cryptographic functions. However performance and cost considerations may require a design that uses several separate cryptographic modules performing distinct functions. For example, a CIMC might use a hardware cryptographic module validated to FIPS 140-2 Level 3 to sign certificates and CRLs, but use a software cryptographic module that has only been validated to Level 2 to compute authentication codes for general transaction messages.

FCS_COP.1 Cryptographic operation

FCS_COP.1.1 The FIPS 140-2 validated cryptographic module shall perform [ST assignment: *list of cryptographic operations performed by the IT environment. The ST author shall include every type of cryptographic operation performed by the IT environment in completing this assignment*] in accordance with a specified cryptographic algorithm [ST assignment: *For each cryptographic operation performed by the IT environment the ST shall specify the standard in accordance with which the operation is performed (e.g., digital signatures are*

generated in accordance with DSA algorithm as specified in FIPS 186-3). A FIPS-approved or recommended algorithm shall be used unless there are no FIPS-approved or recommended algorithms for the type of operation to be performed. If an algorithm that is not FIPS-approved or recommended is used] and cryptographic key sizes [ST assignment: *cryptographic key sizes*] that meet the following: [ST assignment: *list of standards*].

> Application Note: The ST should specify the cryptographic operations that are being performed. Examples of cryptographic operations that may be performed include encryption, decryption, random number generation, signature generation, signature verification, authentication code generation, authentication code verification, hash generation, hash verification, keyed-hash message authentication code generation, keyedhash message authentication code verification. For each cryptographic operation performed, the ST should specify the algorithm or algorithms used and the standard with which that algorithm conforms.

In Section 6.12, cryptographic functions and keys are categorized based on their uses within a CIMC. Security requirements are then imposed on the cryptographic modules within a CIMC, the types of cryptographic functions that are performed by the cryptographic module, and the types of keys that are stored within the cryptographic module.

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 5 lists all the functional security requirements for the TOE that are included in this PP. They are listed in alphabetical order in Table 5 for ease of reference. Also included are the applicable CIMC PP sections to which each requirement applies.

Table 5 CIMC TOE Functional Security Requirements

Security Functional Components	CIMC PP 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
FCS_SOF_CIMC.1 CIMC Strength of Functions	6.12 Strength of Function Requirements
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_ACF_CIMC.2 User private key confidentiality protection	6.6.1 Private Key Storage
FDP_ACF_CIMC.3 User secret key confidentiality protection	6.6.3 Secret Key Storage
FDP_CIMC_CER.1 Certificate Generation	6.10 Certificate Registration
FDP_CIMC_CRL.1 Certificate Revocation	6.11.1 Certificate Revocation List Validation
FDP_CIMC_CSE.1 Certificate status export	6.5.1 Certificate Status Export
FDP_CIMC_OCSP.1 Basic Response Validation	6.11.2 OCSP Basic Response Validation
FDP_ETC_CIMC.5 Extended user private and secret key export	6.6.5 Private and Secret Key Export
FDP_ITT.1 Basic internal transfer protection (iterations 3 and 4)	6.5 Remote Data Entry and Export
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	6.6.2 Public Key Storage
FDP_UCT.1 Basic data exchange confidentiality (iteration 2)	6.5 Remote Data Entry and Export
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

Security Functional Components	CIMC PP Section
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	6.8 Certificate Revocation List Profile Management
FMT_MOF_CIMC.6 OCSP Profile Management	6.9 Online Certificate Status Protocol (OCSP) Profile Management
FMT_MTD_CIMC.4 TSF private key confidentiality protection	6.6.1 Private Key Storage
FMT_MTD_CIMC.5 TSF secret key confidentiality protection	6.6.3 Secret Key Storage
FMT_MTD_CIMC.7 Extended TSF private and secret key export	6.6.5 Private and Secret Key Export
FPT_CIMC_TSP.1 Audit log signing event	6.1 Security Audit
FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 2)	6.5Remote Data Entry and Export
FPT_ITT.1 Basic internal TSF data transfer protection (iterations 3 and 4)	6.5 Remote Data Entry and Export
FPT_STM.1 Reliable time stamps (iteration 2)	6.1 Security Audit

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 CIMC may maintain either a single audit log or multiple audit logs. If multiple audit logs are used, the CIMC may either maintain a different audit log at each of the physically separated parts of the CIMC (e.g., the CA may maintain an audit log in addition to each of the RAs) or may divide audit entries among the audit logs based on the type of event being audited (e.g., audit entries that are to be maintained for a very long time may be placed in a separate audit log to be used as an archive). If multiple audit logs are maintained, each event to be audited (as specified in FAU_GEN.1) must be included in at least one of the audit logs. All other audit requirements apply to each audit log.

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 6 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 6 below.

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

Table 6 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	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	
5.7.1: 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	
6.6.1: Private Key Storage		All access to certificate subject private keys retained within the TOE for key recovery purposes	
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	
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)	

Section/Function	Component	Event	Additional Details
	FMT_MTD_CIMC.7 Extended TSF private and secret key export		
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

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) [ST selection: *object identity, user identity, subject identity, host identity, event type*]

b) [ST assignment: *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.

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 unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to detect unauthorised 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.

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.

FPT_STM.1 Reliable time stamps (iteration 2)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

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 PP will be allocated to distinct roles to maintain the security of a CIMC.

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 7 to the authorized roles as specified in Table 7.

Table 7 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.

Section/Function	Component Function	Authorized Role
		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, Auditor, or Operator.
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 Management	FMT_MOF_CIMC.6 OCSP profile management	The capability to modify the OCSP profile shall be restricted to Administrators.

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 9.2 on [ST assignment: *list of subjects, objects, and operations 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.

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 9.2 to objects based on the following: the identity of the subject and the set of roles 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 8.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [ST assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*].

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.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [ST assignment: *rules, based on security attributes that explicitly deny access of subjects to objects*].

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.

Table 8 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.
5.6.1: 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.
Private Key Load	The capability to request the loading of Component private keys into cryptographic modules shall be restricted to Administrators.
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

Section/Function	Event
	Administrator, Auditor, or Operator shall be required to request the decryption of a certificate subject private key.
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.
6.7.4: Private and Secret Key Destruction	The capability to zeroize CIMC plaintext private and secret keys shall be restricted to Administrators, Auditors, Officers, and Operators.
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, Auditor, or Operator.
Certificate Status Change Approval[3]	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 [

[3] 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.

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 [ST assignment: *list of TSF mediated actions that are not security relevant*] 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 [ST assignment: *list of TSF-mediated actions that are not security relevant*] 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.

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:

a) 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).

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: [ST assignment: *list of user security attributes]*.

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: [ST assignment: *rules for the initial association of attributes*].

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: [ST assignment: *rules for the changing of attributes*].

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). These requirements also apply to communications between physically distributed parts of a single TOE over an untrusted network (e.g., receipt of a signed certificate request message by a CA from an RA would be considered a message receipt even if the RA and CA were being validated as a single CIMC).

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 [ST assignment: *other attribute*s] 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:

a) 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 can not be verified under in the following cases:

a) 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.

FDP_ITT.1 Basic internal transfer protection (iteration 3)

FDP_ITT.1.1 The TSF shall enforce the CIMC TOE Access Control Policy specified in section 9.2 to prevent the modification of security-relevant user data when it is transmitted between physically-separated parts of the TOE.

FDP_ITT.1 Basic internal transfer protection (iteration 4)

FDP_ITT.1.1 The TSF shall enforce the CIMC TOE Access Control Policy specified in section 9.2 to prevent the disclosure of confidential user data when it is transmitted between physically separated parts of the TOE.

FDP_UCT.1 Basic data exchange confidentiality (iteration 2)

FDP_UCT.1.1 The TSF shall enforce the CIMC TOE Access Control Policy specified in section 9.2 to transmit confidential objects in a manner protected from unauthorized disclosure.

FPT_ITC.1 Inter-TSF confidentiality during transmission (iteration 2)

FPT_ITC.1.1 The TSF shall protect all confidential TSF data transmitted from the TSF to a remote trusted IT product from unauthorized disclosure during transmission.

FPT_ITT.1 Basic internal TSF data transfer protection (iteration 3)

FPT_ITT.1.1 The TSF shall protect all security-relevant TSF data from modification when it is transmitted between separate parts of the TOE.

FPT_ITT.1 Basic internal TSF data transfer protection (iteration 4)

FPT_ITT.1.1 The TSF shall protect all confidential TSF data from disclosure when it is transmitted between separate parts of the TOE.

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 [ST assignment: *the X.509 standard for CRLs, the OCSP standard as defined by RFC 2560, other standard (ST shall specify the standard and ST author shall ensure that a description of the format is available), or ST specified format (ST shall include a description of the format)*].

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.

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: If certificate status information is exported using the X.509 CRL format, then the functional security requirements FDP_CIMC_CRL.1 and FMT_MOF_CIMC.5 apply. If certificate status information is exported using the RFC 2560 compliant OCSP format, then the functional security requirements FDP_CIMC_OCSP.1 and FMT_MOF_CIMC.6 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. In some cases, private and secret keys are distributed.

6.6.1 Private Key Storage

Private keys may be used by a CIMC for many different purposes and stored for long periods. CIMCs may store Component keys, CIMS personnel keys, and, for key recovery purposes, certificate subject private keys.

FDP_ACF_CIMC.2 User private key confidentiality protection

FDP_ACF_CIMC.2.1 CIMS personnel private keys shall be stored in a FIPS 140-2 validated cryptographic module or stored in encrypted form. If CIMS personnel private keys are stored in encrypted form, the encryption shall be performed by the FIPS 140-2 validated cryptographic module.

FDP_ACF_CIMC.2.2 If certificate subject private keys are stored in the TOE, they shall be encrypted using a Long Term Private Key Protection Key. The encryption shall be performed by the FIPS 140-2 validated cryptographic module.

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.

FMT_MTD_CIMC.4 TSF private key confidentiality protection

FMT_MTD_CIMC.4.1 CIMC private keys shall be stored in a FIPS 140-2 validated cryptographic module or stored in encrypted form. If CIMC private keys are stored in encrypted form, the encryption shall be performed by the FIPS 140-2 validated cryptographic module.

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 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 [ST assignment: *action to be taken if the verification fails*].

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

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.

6.6.3 Secret Key Storage

Secret (symmetric) keys may be used for several purposes in a CIMC. They may be used to encrypt other secret or private keys when they are stored within or exported from the CIMC. They may also be used to authenticate subscribers (users) and CIMCs. Secret keys must be protected against unauthorized modification and disclosure.

Applicants for certificates may be given PIN or password authenticators. The process for generating and delivering these authenticators to applicants is outside the scope of this document.

The following requirements are mandatory if the CIMC stores secret keys.

FDP_ACF_CIMC.3 User secret key confidentiality protection

FDP_ACF_CIMC.3.1 User secret keys stored within the CIMC, but not within a FIPS 140-2 validated cryptographic module, shall be stored in encrypted form. The encryption shall be performed by the FIPS 140-2 validated cryptographic module.

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.

FMT_MTD_CIMC.5 TSF secret key confidentiality protection

FMT_MTD_CIMC.5.1 TSF secret keys stored within the TOE, but not within a FIPS 140-2 validated cryptographic module, shall be stored in encrypted form. The encryption shall be performed by the FIPS 140-2 validated cryptographic module.

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.

6.6.4 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.

6.6.5 Private and Secret Key Export

Keys may be exported from cryptographic modules for a variety of reasons, including key backup, replication, and transmission of user private keys generated in CIMCs.

FDP_ETC_CIMC.5 Extended user private and secret key export

FDP_ETC_CIMC.5.1 Private and secret keys shall only be exported from the TOE in encrypted form or using split knowledge procedures. Electronically distributed secret and private keys shall only be exported from the TOE in encrypted form.

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.

FMT_MTD_CIMC.7 Extended TSF private and secret key export

FMT_MTD_CIMC.7.1 Private and secret keys shall only be exported from the TOE in encrypted form or using split knowledge procedures. Electronically distributed secret and private keys shall only be exported from the TOE in encrypted form.

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.

6.7 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 Online Certificate Status Protocol (OCSP) Profile Management

An online certificate status protocol profile is used to define the set of acceptable values for the fields in an OCSP response. The OCSP profile may specify the type(s) of responses that the CIMC may generate (i.e., acceptable values for **responseType**) as well as the set of acceptable values for the fields within the acceptable response types. An examples of a value that may be covered by an OCSP profile for the basic response type is **ResponderID**, the identifier of the OCSP responder.

FMT_MOF_CIMC.6 OCSP profile management

FMT_MOF_CIMC.6.1 If the TSF issues OCSP responses, the TSF shall implement an OCSP profile and ensure that issued OCSP responses are consistent with the OCSP profile.

FMT_MOF_CIMC.6.2 If the TSF issues OCSP responses, the TSF shall require the Administrator to specify the set of acceptable values for the **responseType** field (unless the CIMC can only issue responses of the basic response type).

FMT_MOF_CIMC.6.3 If the TSF is configured to allow OCSP responses of the basic response type, the TSF shall require the Administrator to specify the set of acceptable values for the **ResponderID** field within the basic response type.

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.

6.10 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 [ST assignment*: the X.509 standard for public key certificates, other standard (ST shall specify the standard and ST author shall ensure that a description of the format is available), or ST specified format (ST shall include a description of the format)*].

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.

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.

6.11 Certificate Revocation

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

6.11.1 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.

6.11.2 OCSP Basic Response Validation

OCSP basic responses issued by CIMCs shall be compliant with IETF RFC 2560. Any fields or extensions to be included in an OCSP response shall be created by the CIMC according to IETF RFC 2560.

FDP_CIMC_OCSP.1 OCSP basic response validation

FDP_CIMC_OCSP.1.1 If a TSF is configured to allow OCSP responses of the basic response type, the TSF shall verify that all mandatory fields in the OCSP basic response contain values in accordance with IETF RFC 2560. At a minimum, the following items shall be validated:
1. The **version** field shall contain a **0**.
2. If the **issuer** field contains a null **Name** (e.g., a sequence of zero relative distinguished names), then the response shall contain a critical **issuerAltName** extension.
3. The **signatureAlgorithm** field shall contain the OID for a FIPS-approved digital signature algorithm.
4. The **thisUpdate** field shall indicate the time at which the status being indicated is known to be correct.
5. The **producedAt** field shall indicate the time at which the OCSP responder signed the response.
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.

6.12 Strength of Function Requirements

This section specified strength of functions for the cryptographic mechanisms in addition to the algorithm and key size requirements.

FCS_SOF_CIMC.1 CIMC Strength of Functions

FCS_SOF_CIMC.1.1 The TSF shall provide cryptographic mechanisms that fulfill the specific Strength of Function requirements of section 6.12.1.

Dependencies: No dependencies

Rationale: This component is necessary to require specific Strength of Function metrics for cryptographic mechanisms of the TSF.

6.12.1 Cryptographic Modules

FIPS 140-2 validated cryptographic modules must perform all cryptographic functions performed by CIMCs. FIPS 140-2 validated cryptographic modules are also required to generate cryptographic keys and to store plaintext private and secret keys.

6.12.1.1 Encryption and FIPS 140-2 Validated Modules

As noted earlier in the document, references to FIPS 140-2 refer to the most current version of the standard and the most current version can be found at http://csrc.nist.gov/cryptval.

6.12.1.1.1 Encryption Algorithms

The encryption specified for:

FAU_STG.1	Protected audit trail storage
FCO_NRO_CIMC.4	Advanced verification of origin
FDP_ACF_CIMC.2	User private key confidentiality protection
FDP_ACF_CIMC.3	User secret key confidentiality protection
FDP_ETC_CIMC.5	Extended user private and secret key export
FDP_SDI_CIMC.3	Stored public key integrity monitoring and action
FMT_MTD_CIMC.4	TSF private key confidentiality protection
FMT_MTD_CIMC.5	TSF secret key confidentiality protection
FMT_MTD_CIMC.7	Extended TSF private and secret key export
FPT_CIMC_TSP.1	Audit log signing event

shall be performed using a FIPS-approved or recommended algorithm.

6.12.1.1.2 FIPS 140-2 Validated Cryptographic Modules

Cryptographic modules specified for:

FDP_ACF_CIMC.2	User private key confidentiality protection
FDP_ACF_CIMC.3	User secret key confidentiality protection
FDP_ETC_CIMC.5	Extended user private and secret key export
FDP_SDI_CIMC.3	Stored public key integrity monitoring and action
FMT_MTD_CIMC.4	TSF private key confidentiality protection
FMT_MTD_CIMC.5	TSF secret key confidentiality protection
FMT_MTD_CIMC.7	Extended TSF private and secret key export
FPT_CIMC_TSP.1	Audit log signing event

shall be validated against FIPS 140-2.

6.12.1.1.3 Split Knowledge Procedures

Split-knowledge procedures specified in:

FDP_ETC_CIMC.5	Extended user private and secret key export
FMT_MTD_CIMC.7	Extended TSF private and secret key export

shall be implemented and validated as specified in FIPS 140-2.

6.12.1.1.4 Authentication Codes

The authentication code specified in:

FAU_STG.1	Protected audit trail storage
FCO_NRO_CIMC.4	Advanced verification of origin
FPT_CIMC_TSP.1	Audit log signing event

FDP_SDI_CIMC.3 Stored public key integrity monitoring and action

shall be a FIPS-approved or recommended authentication code.

6.12.1.2 Cryptographic module levels for cryptographic functions that involve private or secret keys

All cryptographic operations performed (including key generation) at the request of the TOE shall be performed in a FIPS 140-2 validated cryptographic module operating in a FIPS-approved or recommended mode of operation.

Table 9 specifies for each category of use for a private or secret key, the required overall FIPS 140-2 level for the validated cryptographic module. If the CIMC generates certificate subject private keys, the required overall FIPS 140-2 level for *Long Term Private Key Protection* keys shall apply.

Table 9 FIPS 140-2 Level for Validated Cryptographic Module

Required Overall FIPS 140-2 Level for CIMC Cryptographic Modules	
Category of Use	**FIPS 140-2 Level**
Certificate and Status Signing - single party signature - multiparty signature	 3 2
Integrity or Approval Authentication - single approval - dual approval	 2 2
General Authentication	2
Long Term Private Key Protection	3
Long Term Confidentiality	2
Short Term Private key Protection	2
Short Term Confidentiality	1

6.12.1.3 Cryptographic Functions That Do Not Involve Private or Secret Keys

There are two other cryptographic functions that may be performed in CIMCs that do not require private or secret keys. These include:

1) *Hash Generation*: One-way hash functions may be used in the process of signature generation and verification (a signature is typically generated by applying a private key to the hash of the message). The generation of a hash does not require a key. Therefore, hash generation does not have the same confidentiality requirements of other cryptographic functions.
2) *Signature Verification*: Signatures are verified from a message text and a public key.

For a cryptographic module that only performs signature verification and/or keyless hash generation functions, the overall required FIPS 140-2 level shall be Level 1.

7 TOE SECURITY ASSURANCE REQUIREMENTS

The assurance requirements for CIMCs conforming to this PP are the requirements for EAL 4 augmented with ALC_FLR.2 (flaw reporting procedures). 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 10 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

7.1.1 Development (ADV)

7.1.1.1 Security architecture description (ADV_ARC.1)

ADV_ARC.1.1d The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV_ARC.1.2d The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.

ADV_ARC.1.3d The developer shall provide a security architecture description of the TSF.

ADV_ARC.1.1c The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.

ADV_ARC.1.2c The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV_ARC.1.3c The security architecture description shall describe how the TSF initialisation process is secure.

ADV_ARC.1.4c The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV_ARC.1.5c The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.

ADV_ARC.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.1.2 Complete functional specification (ADV_FSP.4)

ADV_FSP.4.1d The developer shall provide a functional specification.
ADV_FSP.4.2d The developer shall provide a tracing from the functional specification to the SFRs.
ADV_FSP.4.1c The functional specification shall completely represent the TSF.
ADV_FSP.4.2c The functional specification shall describe the purpose and method of use for all TSFI.
ADV_FSP.4.3c The functional specification shall identify and describe all parameters associated with each TSFI.
ADV_FSP.4.4c The functional specification shall describe all actions associated with each TSFI.
ADV_FSP.4.5c The functional specification shall describe all direct error messages that may result from an invocation of each TSFI.
ADV_FSP.4.6c The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.
ADV_FSP.4.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.4.2e The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

7.1.1.3 Implementation Representation of the TSF (ADV_IMP.1)

ADV_IMP.1.1c The implementation representation shall define the TSF to a level of detail such that the TSF can be generated without further design decisions.
ADV_ IMP.1.2c The implementation representation shall be in the form used by the development personnel.
ADV_ IMP.1.3c The mapping between the TOE design description and the sample of the implementation representation shall demonstrate their correspondence.
ADV_ IMP.1.1e The evaluator *shall* confirm that, for the selected sample of the implementation representation, the information provided meets all requirements for content and presentation of evidence.

7.1.1.4 Basic modular design (ADV_TDS.3)

ADV_TDS.3.1d The developer shall provide the design of the TOE.
ADV_TDS.3.2d The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.
ADV_TDS.3.1c The design shall describe the structure of the TOE in terms of subsystems.
ADV_TDS.3.2c The design shall describe the TSF in terms of modules.
ADV_TDS.3.3c The design shall identify all subsystems of the TSF.
ADV_TDS.3.4c The design shall provide a description of each subsystem of the TSF.
ADV_TDS.3.5c The design shall provide a description of the interactions among all subsystems of the TSF.
ADV_TDS.3.6c The design shall provide a mapping from the subsystems of the TSF to the modules of the TSF.
ADV_TDS.3.7c The design shall describe each SFR-enforcing module in terms of its purpose and relationship with other modules.
ADV_TDS.3.8c The design shall describe each SFR-enforcing module in terms of its SFR-related interfaces, return values from those interfaces, interaction with other modules and called SFR-related interfaces to other SFR-enforcing modules.
ADV_TDS.3.9c The design shall describe each SFR-supporting or SFR-non-interfering module in terms of its purpose and interaction with other modules.
ADV_TDS.3.10c The mapping shall demonstrate that all TSFIs trace to the behaviour described in the TOE design that they invoke.
ADV_TDS.3.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_TDS.3.2e The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

7.1.2 Guidance documents (AGD)

7.1.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1d The developer shall provide operational user guidance.

AGD_OPE.1.1c The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6c The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.2.2 Preparative procedures (AGD_PRE.1)

AGD_PRE.1.1d The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1c The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

7.1.3 Life-cycle support (ALC)

7.1.3.1 Production support, acceptance procedures and automation (ALC_CMC.4)

ALC_CMC.4.1d The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.4.2d The developer shall provide the CM documentation.

ALC_CMC.4.3d The developer shall use a CM system.

ALC_CMC.4.1c The TOE shall be labelled with its unique reference.

ALC_CMC.4.2c The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.4.3c The CM system shall uniquely identify all configuration items.

ALC_CMC.4.4c The CM system shall provide automated measures such that only authorised changes are made to the configuration items.

ALC_CMC.4.5c The CM system shall support the production of the TOE by automated means.

ALC_CMC.4.6c The CM documentation shall include a CM plan.

ALC_CMC.4.7c The CM plan shall describe how the CM system is used for the development of the TOE

ALC_CMC.4.8c The CM plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

ALC_CMC.4.9c The evidence shall demonstrate that all configuration items are being maintained under the CM system.

ALC_CMC.4.10c The evidence shall demonstrate that the CM system is being operated in accordance with the CM plan.

ALC_CMC.4.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.3.2 Problem tracking CM coverage (ALC_CMS.4)

ALC_CMS.4.1d The developer shall provide a configuration list for the TOE.

ALC_CMS.4.1c The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation representation; and security flaw reports and resolution status.

ALC_CMS.4.2c The configuration list shall uniquely identify the configuration items.

ALC_CMS.4.3c For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

ALC_CMS.4.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.3.3 Delivery procedures (ALC_DEL.1)

ALC_DEL.1.1d The developer shall document and provide procedures for delivery of the TOE or parts of it to the consumer.

ALC_DEL.1.2d The developer shall use the delivery procedures.

ALC_DEL.1.1c The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

ALC_DEL.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.3.4 Identification of security measures (ALC_DVS.1)

ALC_DVS.1.1d The developer shall produce and provide development security documentation.

ALC_DVS.1.1c The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ALC_DVS.1.2e The evaluator *shall* confirm that the security measures are being applied.

7.1.3.5 Flaw reporting procedures (ALC_FLR.2)

ALC_FLR.2.1d The developer shall document and provide flaw remediation procedures addressed to TOE developers.

ALC_FLR.2.2d The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

ALC_FLR.2.3d The developer shall provide flaw remediation guidance addressed to TOE users.

ALC_FLR.2.1c The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.2.2c The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.2.3c The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.2.4c The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC_FLR.2.5c The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

ALC_FLR.2.6c The procedures for processing reported security flaws shall ensure that any reported flaws are remediated and the remediation procedures issued to TOE users.

ALC_FLR.2.7c The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

ALC_FLR.2.8c The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

ALC_FLR.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.3.6 Developer defined life-cycle model (ALC_LCD.1)

ALC_LCD.1.1d The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.

ALC_ LCD.1.2d The developer shall provide life-cycle definition documentation.

ALC_ LCD.1.1c The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_ LCD.1.2c The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.

ALC_ LCD.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.3.7 Well-defined development tools (ALC_TAT.1)

ALC_TAT.1.1d The developer shall provide the documentation identifying each development tool being used for the TOE.

ALC_ TAT.1.2d The developer shall document and provide the selected implementation-dependent options of each development tool.

ALC_ TAT.1.1c Each development tool used for implementation shall be well-defined.

ALC_ TAT.1.2c The documentation of each development tool shall unambiguously define the meaning of all statements as well as all conventions and directives used in the implementation.

ALC_ TAT.1.3c The documentation of each development tool shall unambiguously define the meaning of all implementation-dependent options.

ALC_ TAT.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.4 Tests (ATE)

7.1.4.1 Analysis of coverage (ATE_COV.2)

ATE_COV.2.1d The developer shall provide an analysis of the test coverage.

ATE_COV.2.1c The analysis of the test coverage shall demonstrate the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

ATE_COV.2.2c The analysis of the test coverage shall demonstrate that all TSFIs in the functional specification have been tested.

ATE_COV.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.4.2 Testing: basic design (ATE_DPT.1)

ATE_DPT.1.1d The developer shall provide the analysis of the deoth of testing.

ATE_ DPT.1.1c The analysis of the depth of testing shall demonstrate the correspondence between the tests in the test documentation and the TSF subsystems in the TOE design.

ATE_ DPT.1.2c The analysis of the depth of testing shall demonstrate that all TSF subsystems in the TOE design have been tested.

ATE_ DPT.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.4.3 Functional testing (ATE_FUN.1)

ATE_FUN.1.1d The developer shall test the TSF and document the results.

ATE_FUN.1.2d The developer shall provide test documentation.

ATE_FUN.1.1c The test documentation shall consist of test plans, expected test results and actual test results.

ATE_FUN.1.2c The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.3c The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.4c The actual test results shall be consistent with the expected test results.

ATE_FUN.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.1.4.4 Independent testing - sample (ATE_IND.2)

ATE_IND.2.1d The developer shall provide the TOE for testing.

ATE_IND.2.1c The TOE shall be suitable for testing.

ATE_IND.2.2c The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

ATE_IND.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2e The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

ATE_IND.2.3e The evaluator shall test a subset of the TSF interfaces to confirm that the TSF operates as specified.

7.1.5 Vulnerability assessment (AVA)

7.1.5.1 Focused vulnerability analysis (AVA_VAN.3)

AVA_VAN.3.1d The developer shall provide the TOE for testing.

AVA_VAN.3.1c The TOE shall be suitable for testing.

AVA_VAN.3.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.3.2e The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.3.3e The evaluator shall perform an independent, focused vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, security architecture description and implementation representation to identify potential vulnerabilities in the TOE.

AVA_VAN.3.4e The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Enhanced-Basic attack potential.

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 11 maps security objectives for the TOE to threats, Table 12 maps security objectives for the environment to threats, and Table 13 maps security objectives for both the TOE and the environment to threats. Table 14 maps the organizational security policies to security objectives. Table 15 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 11 Relationship of Security Objectives for the TOE to Threats

IT Security Objective	Threat
O.Certificates	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Control unknown source communication traffic	T.Hacker gains access
O.Non-repudiation	T.Sender denies sending information
O.Preservation/trusted recovery of secure state	T.Critical system component fails
O.Sufficient backup storage and effective restoration	T.Critical system component fails, T.User error makes data inaccessible

Table 12 Relationship of Security Objectives for the Environment to Threats

Non-IT Security Objective	Threat
O.Administrators, Operators, Officers and Auditors guidance documentation	T.Disclosure of private and secret keys, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions, T.Social engineering
O.Competent Administrators, Operators, Officers and Auditors	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.CPS	T.Administrative errors of omission
O.Cryptographic functions	T.Disclosure of private and secret keys, T.Modification of secret/private keys
O.Installation	T.Critical system component fails
O.Lifecycle security	T.Critical system component fails, T.Malicious code exploitation
O.Notify Authorities of Security Issues	T.Hacker gains access
O.Periodically check integrity	T.Malicious code exploitation
O.Physical Protection	T.Hacker physical access
O.Repair identified security flaws	T.Flawed code, T.Critical system component fails
O.Security roles	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Social Engineering Training	T.Social Engineering
O.Trusted path	T.Hacker gains access, T.Message content modification
O.Validation of security function	T.Malicious code exploitation,

Non-IT Security Objective	Threat
	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions

Table 13 Relationship of Security Objectives for Both the TOE and the Environment to Threats

Non-IT Security Objective	Threat
O.Configuration management	T.Critical system component fails, T.Malicious code exploitation
O.Data import/export	T.Message content modification
O.Detect modifications of firmware, software, and backup data	T.User error makes data inaccessible, T.Administrators, Operators, 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, Operators, Officers and Auditors commit errors or hostile actions, T.User abuses authorization to collect and/or send data
O.Integrity protection of user data and software	T.Modification of private/secret keys, T.Malicious code exploitation
O.Limitation of administrative access	T.Disclosure of secret and private keys, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Maintain user attributes	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Manage behavior of security functions	T.Critical system component fails, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Object and data recovery free from malicious code	T.Modification of secret/private keys, T.Malicious code exploitation
O.Procedures for preventing malicious code	T.Malicious code exploitation, T.Social engineering
O.Protect stored audit records	T.Modification of secret/private keys, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Protect user and TSF data during internal transfer	T.Message content modification, T.Disclosure of private and secret keys
O.React to detected attacks	T.Hacker gains access
O.Require inspection for downloads	T.Malicious code exploitation
O.Respond to possible loss of stored audit records	T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Restrict actions before authentication	T.Hacker gains access, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions
O.Security-relevant configuration management	T.Administrative errors of omission
O.Time stamps	T.Critical system component fails, T.Administrators, Operators, Officers and Auditors commit errors or hostile actions

Table 14 Relationship of Organizational Security Policies to Security Objectives

Security Policy	Objective
P.Authorized use of information	O.Auditors review audit logs O.Maintain user attributes O.Restrict actions before authentication

Security Policy	Objective
	O.Security roles O.User authorization management
P.Cryptography	O.Cryptographic functions

Table 15 Relationship of Assumptions to IT Security Objectives

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

O.CPS provides Administrators, Operators, 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.

O.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:

O.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.

O.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 can not be detected, the backup copy is not a reliable source for restoration of user data.

T.Administrators, Operators, 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:

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

O.Administrators, Operators, 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.

O.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.

O.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.

O.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 faildue 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.

O.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.

O.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.

O.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).

O.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..

O.Lifecycle security provides tools and techniques that are used throughout the development phase reducing the likelihood of hardware or software imperfections. **O.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.

O.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:

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

O.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.

O.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.

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

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

O.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.

O.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. **O.Lifecycle security** also addresses the detection and resolution of flaws discovered during the operational phase, such as modifications of components by malicious code.

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:

O.Data Import/Export protects data when being transmitted to or from the TOE. Protection of data in transit permits the TOE or the external user to detect modified messages, message replay, or fraudulent messages.

O.Protect user and TSF data during internal transfer protects data being transmitted between separated parts of the TOE. Protection of data in transit permits the TOE to detect modified messages, message replay, or fraudulent messages.

O.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:

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

O.Cryptographic functions ensures that TOE implements 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.

O.Protect user and TSF data during internal transfer protects private and secret keys from unauthorized disclosure during transmission between separated parts of the TOE.

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

It is countered by:

O.Cryptographic functions ensures that TOE implements 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 and software that ensures that appropriate integrity protection is provided for secret and private keys.

O.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.

O.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.

O.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:

O.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:

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

O.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.

O.Social Engineering Training which ensures that general users, Administrators, Operators, 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, **O.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. This is addressed by **O.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 **O.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 **O.Authentication Data Management**, which ensures that users modify their authentication data in accordance with appropriate security policy.

A.Competent Administrators, Operators, Officers and Auditors establishes that security of the TOE is dependent upon those that manage it. This is addressed by **O.Competent Administrators, Operators, Officers and Auditors**, which ensures that the system managers will be competent in its administration.

A.CPS establishes that Administrators, Operators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated. This is addressed by **O.CPS,** which ensures that Administrators, Operators, Officers, and Auditors are familiar with the CP and CPS under which the TOE is operated.

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 **O.Disposal of Authentication Data**, which ensures that access to the system will be denied after a user's privileges have been removed.

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 **O.Malicious Code Not Signed,** which ensures that code must be signed by a trusted party or it will not be loaded onto the system.

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 **O.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 **O.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 **O.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 **O.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 **O.Communications Protection**, which ensures that adequate physical protections are afforded the necessary communications infrastructure.

A.Physical Protection establishes that physical modification of the TOE hardware, software, and firmware will compromise system security. This is addressed by **O.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 16, addresses the mapping of security functional requirements to

security objectives. The second table, Table 17, addresses the mapping of security assurance requirements to security objectives.

Table 16 Security Functional Requirements Related to Security Objectives

Functional Requirement	Objective
FAU_GEN.1 Audit data generation (iterations 1 and 2)	O.Individual accountability and audit records
FAU_GEN.2 User identity association (iterations 1 and 2)	O.Individual accountability and audit records
FAU_SAR.1 Audit review	O.Individual accountability and audit records
FAU_SAR.3 Selectable audit review	O.Individual accountability and audit records
FAU_SEL.1 Selective audit (iterations 1 and 2)	O.Individual accountability and audit records
FAU_STG.1 Protected audit trail storage (iterations 1 and 2)	O.Protect stored audit records
FAU_STG.4 Prevention of audit data loss (iterations 1 and 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.1 Cryptographic key generation	O.Cryptographic functions
FCS_CKM.4 Cryptographic key destruction	O.Procedures for preventing malicious code, O.React to detected attacks
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	O.Procedures for preventing malicious code, O.React to detected attacks
FCS_COP.1 Cryptographic operation	O.Cryptographic functions
FCS_SOF_CIMC.1 CIMC Strength of Functions	O.Cryptographic functions
FDP_ACC.1 Subset access control (iterations 1 and 2)	O.Limitation of administrative access
FDP_ACF.1 Security attribute based access control (iterations 1 and 2)	O.Limitation of administrative access
FDP_ACF_CIMC.2 User private key confidentiality protection	O.Certificates, O.Procedures for preventing malicious code
FDP_ACF_CIMC.3 User secret key confidentiality protection	O.Certificates, O.Procedures for preventing malicious code
FDP_CIMC_BKP.1 CIMC backup and recovery	O.Object and data recovery free from malicious code, O.Preservation/trusted recovery of secure state, O.Sufficient backup storage and effective restoration
FDP_CIMC_BKP.2 Extended CIMC backup and recovery	O.Detect modifications of firmware, software, and backup data, O.Object and data recovery free from malicious code
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_CIMC_OCSP.1 OCSP basic response validation	O.Certificates
FDP_ETC_CIMC.5 Extended user private and secret key export	O.Data import/export
FDP_ITT.1 Basic internal transfer protection (iterations 1 and 3)	O.Integrity protection of user data and software, O.Protect user and TSF data during internal transfer
FDP_ITT.1 Basic internal transfer protection (iterations 2 and 4)	O.Protect user and TSF data during internal transfer
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	O.Integrity protection of user data and software
FDP_UCT.1 Basic data exchange confidentiality (iterations 1 and 2)	O.Data import/export
FIA_AFL.1 Authentication failure handling	O.React to detected attacks

Functional Requirement	Objective
FIA_ATD.1 User attribute definition	O.Maintain user attributes
FIA_SOS.1 Verification of secrets (iterations 1 and 2)	O.Limitation of administrative access
FIA_UAU.1 Timing of authentication (iterations 1 and 2)	O.Limitation of administrative access, O.Restrict actions before authentication
FIA_UID.1 Timing of identification (iterations 1 and 2)	O.Individual accountability and audit records, O.Limitation of administrative access
FIA_USB.1 User-subject binding (iterations 1 and 2)	O.Maintain user attributes
FMT_MOF.1 Management of security functions behavior (iterations 1 and 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_MOF_CIMC.6 OCSP Profile Management	O.Configuration management
FMT_MSA.1 Management of security attributes	O.Maintain user attributes, O.User authorization management
FMT_MSA.2 Secure security attributes	O.Security-relevant configuration management
FMT_MSA.3 Static attribute initialization	O.Security-relevant configuration management
FMT_MTD.1 Management of TSF data	O.Individual accountability and audit records, O.Protect stored audit records
FMT_MTD_CIMC.4 TSF private key confidentiality protection	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software
FMT_MTD_CIMC.5 TSF secret key confidentiality protection	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software
FMT_MTD_CIMC.7 Extended TSF private and secret key export	O.Data import/export
FMT_SMR.2 Restrictions on security roles	O.Security roles
FPT_TST_CIMC.1 Abstract Machine testing	O.Periodically check integrity, O.Validation of security function
FPT_CIMC_TSP.1 Audit log signing event	O.Protect stored audit records
FPT_ITC.1 Inter-TSF confidentiality during transmission (iterations 1 and 2)	O.Data import/export
FPT_ITT.1 Basic internal TSF data transfer protection (iterations 1-4)	O.Protect user and TSF data during internal transfer
FPT_STM.1 Reliable time stamps (iterations 1 and 2)	O.Individual accountability and audit records, O.Time stamps
FPT_TST_CIMC.2 Software/firmware integrity test	O.Detect modifications of firmware, software, and backup data, O.Integrity protection of user data and software, O.Object and data recovery free from malicious code, O.Periodically check integrity, O.Procedures for preventing malicious code, O.Validation of security function
FPT_TST_CIMC.3 Software/firmware load test	O.Integrity protection of user data and software, O.Object and data recovery free from malicious code, O.Periodically check integrity, O.Require inspection for downloads
FTP_TRP.1 Trusted path	O.Trusted path

Table 17 Security Assurance Requirements Related to Security Objectives

Assurance Requirement	Objective

ADV_ARC.1: Security architecture description	selection of EAL 4, O.Lifecycle security
ADV_FSP.4 Complete functional specification	selection of EAL 4, O.Lifecycle security
ADV_IMP.1 Implementation representation of the TSF	selection of EAL 4, O.Lifecycle security
ADV_TDS.3 Basic modular design	selection of EAL 4, O.Lifecycle security
AGD_ OPE.1: Operational user guidance	O.Administrators, Operators, Officers and Auditors guidance documentation, O.Auditors Review Audit Logs, O.Competent Administrators, Operators, Officers and Auditors, O.Configuration Management, O.Installation, O.Malicious Code Not Signed, O.Procedures for preventing malicious code, O.Require inspection for downloads, O.Security-relevant configuration management, O.User authorization management, selection of EAL 1, EAL-CSPP, EAL 3, EAL 4
AGD_PRE.1: Preparative procedures	selection of EAL 4, O.Installation
ALC_CMC.4: Production support, acceptance procedures and automation	selection of EAL 4, O.Configuration management
ALC_CMS.4: Problem tracking CM coverage	selection of EAL 4, O.Configuration management
ALC_DEL.1: Delivery procedures	selection of EAL 4
ALC_DVS.1 Identification of security measures	selection of EAL 4
ALC_FLR.2 Flaw reporting procedures	O.Lifecycle security, O.Repair identified security flaws
ALC_LCD.1 Developer defined life-cycle model	selection of EAL 4
ALC_TAT.1 Well-defined development tools	selection of EAL 4
ATE_COV.2 Analysis of coverage	selection of EAL 4
ATE_DPT.1 Testing: Basic Design	selection of EAL 4
ATE_FUN.1 Functional testing	selection of EAL 4
ATE_IND.2 Independent Testing – Sample	selection of EAL 4
AVA_VAN.3 Focused vulnerability analysis	selection of EAL 4

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), FDP_CIMC_CSE.1 (Certificate status export), and FDP_CIMC_OCSP.1 (OCSP basic response validation) which ensure that certificate revocation lists and certificate status information are valid. In the case that the TOE maintains a copy of the certificate subject's private key, FDP_ACF_CIMC.2 (User private key confidentiality protection) ensures that the certificate is not invalidated by the disclosure of the private key by the TOE. In the case that a secret key is used by the certificate subject as an authenticator in requesting a certificate, FDP_ACF_CIMC.3 (User secret key confidentiality protection) ensures that an attacker can not obtain a bad certificate by obtaining a user's authenticator from the TOE and then using that authenticator to obtain a bad certificate.

8.2.2.1.2 System

O.Preservation/trusted recovery of secure state is provided by **FDP_CIMC_BKP.1 (CIMC backup and recovery)** which covers the requirement that the state of the system be preserved so that it can be recovered in the event of a secure component failure.

O.Sufficient backup storage and effective restoration is provided by **FDP_CIMC_BKP.1 (CIMC backup and recovery)** which covers the requirement that sufficient backup data is created and stored and that an effective restoration procedure is provided.

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.

8.2.2.2 Non-IT Security Objectives for the Environment

O.Administrators, Operators, Officers and Auditors guidance documentation is provided by **AGD_OPE.1 (Operational user guidance)** which ensures that adequate guidance on the secure operation of the TOE is provided to Administrators, Operators, Officers, and Auditors.

O.Auditors Review Audit Logs is provided by **A.Auditors Review Audit Logs** which ensures that auditors review the audit logs. It is also supported by **AGD_OPE.1 (Operational user guidance)** which ensures that Auditors are provided with the information they need to understand the contents of the audit logs.

O.Authentication Data Management is provided by **A.Authentication Data Management** which covers the requirement that an authentication data management policy be enforced.

O.Communications Protection is provided by **A.Communications Protection** which covers the requirement that the system be adequately physically protected against loss of communications.

O.Competent Administrators, Operators, Officers and Auditors is provided **by A.Competent Administrators, Operators, Officers and Auditors** which covers the requirement that Administrators, Operators, Officers, and Auditors be capable of managing the TOE and the security of the information it contains. It is also supported by **AGD_OPE.1 (Operational user guidance)**which ensures that Administrators, Operators, Officers, and Auditors are provided with the information they need to properly manage the TOE and its security functionality.

O.CPS is provided by **A.CPS** which covers the requirement that Administrators, Operators, Officers, and Auditors be familiar with the CP and CPS under which the TOE is operated.

O.Installation is provided by **AGD_OPE.1 (Operational user guidance)** and **AGD_PRE.1 (Preparative procedures)** which cover the requirement that Administrators, Operators, Officers, and Auditors be provided with documentation describing the procedures necessary to securely install and operate the TOE. **A.Competent Administrators, Operators, Officers and Auditors** covers the requirement that competent Administrators, Operators, Officers, and Auditors, who are capable of securely managing the TOE, are used.

O.Malicious Code Not Signed is provided by **A.Malicious Code Not Signed** which covers the requirement that malicious code destined for the TOE is not signed by a trusted entity. It is also supported by **AGD_OPE.1 (Operational user guidance)** which ensures that entities that are trusted to sign code are aware of their responsibilities.

O.Notify Authorities of Security Issues is provided by **A.Notify Authorities of Security Issues** which covers the requirement that proper authorities be notified of any security issues that impact their systems.

O.Physical Protection is provided by **A.Physical Protection** which covers the requirement that TOE hardware, software, and firmware critical to security policy enforcement be protected from unauthorized physical modification.

O.Social Engineering Training is provided by **A.Social Engineering Training** which covers the requirement that general users, administrators, operators, officers, and auditors are trained in techniques to thwart social engineering attacks.

O.Cooperative Users is provided by **A.Cooperative Users** which covers the requirement that users act in a cooperative manner.

O.Lifecycle security is provided by ADV_ARC.1 (Security architecture description), ADV_FSP.4 (Complete functional specification), ADV_IMP.1 (Implementation representation of the TSF), and ADV_TDS.3 (Basic modular design) which cover the requirement that security is designed into the CIMC. ALC_FLR.2 (Flaw reporting procedures) covers the requirement that flaws are detected and resolved during the operational phase.

O.Repair identified security is provided by **ALC_FLR.2 (Flaw reporting procedures)** which covers the requirement that vendor repair security flaws that have been identified by a user.

8.2.2.3 IT Security Objectives for the Environment

O.Cryptographic functions is provided by **FCS_CKM.1 (Cryptographic key generation)** and **FCS_COP.1 (Cryptographic operation)** which cover the requirement that approved algorithms be used for encryption/decryption, authentication, and signature generation/verification and that approved key generation techniques be used. In addition, **FCS_SOF_CIMC.1 (CIMC Strength of Functions)** ensures that the selected cryptographic algorithms and their implementation and engineering have acceptable strength.

O.Operating System is provided by **A.Operating System** which covers the requirement that the operating system(s) on which the TSF operates provides security functions required by the CIMC to counter the perceived threats identified in this PP.

O.Periodically check integrity is provided by FPT_TST_CIMC.1 (Abstract Machine testing) which covers the requirement provide periodic integrity checks on the system and FPT_TST_CIMC.2 (Software/firmware integrity test) and FPT_TST_CIMC.3 (Software/firmware load test) cover the requirement to periodically check the integrity of software.

O.Security roles is provided by **FMT_SMR.2 (Restrictions on security roles)** which covers the requirement that a set of security roles be maintained and that users be associated with those roles.

O.Validation of security function is provided by **FPT_TST_CIMC.1 (Abstract Machine testing)** which covers the requirement to ensure that security-relevant hardware and firmware are functioning correctly and **FPT_TST_CIMC.2 (Software/firmware integrity test)** which covers the requirement to ensure that security-relevant software is functioning correctly.

O.Trusted Path is provided by **FTP_TRP.1 (Trusted path)** which covers the requirement that a trusted path between the user and the system be provided.

8.2.2.4 Security Objectives for the TOE and Environment

O.Configuration Management is provided by FMT_MOF.1 (Management of security functions behavior) (iterations 1 and 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. FMT_MOF_CIMC.6 (OCSP Profile Management) covers the requirement that Administrators be able to control to the types of information that are included in generated OCSP responses. O.Configuration Management is supported by **AGD_OPE.1 (Operational user guidance)** which covers the requirement that Administrators be provided with documentation describing the configuration management features of the TOE and by A.Competent Administrators, Operators, Officers and Auditors and A.CPS which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated. O.Configuration Management is also supported by ALC_CMC.4 (Production support,

acceptance procedures and automation) and ALC_CMS.4 (Problem tracking CM coverage) which ensure that a configuration management system is implemented and used.

O.Data import/export is provided by FDP_UCT.1 (Basic data exchange confidentiality) (iterations 1 and 2) and FPT_ITC.1 (Inter-TSF confidentiality during transmission) (iterations 1 and 2) which cover the requirement that data other than private and secret keys be protected when they are transmitted and from the CIMC. FDP_ETC_CIMC.5 (Extended user private and secret key export) and FMT_MTD_CIMC.7 (Extended TSF private and secret key export) cover the requirement that private and secret keys be protected when they are transmitted to and from the TOE.

O.Detect modifications of firmware, software, and backup data is provided by FPT_TST_CIMC.2 (Software/firmware integrity test) which covers the requirement that modifications to software or firmware be detected and FDP_CIMC_BKP.2 (Extended CIMC backup and recovery) which covers the requirement that modifications to backup data be detected. Since FPT_TST_CIMC.2 and FDP_CIMC_BKP.2 make use of digital signatures, keyed hashes, or authentication codes to detect modifications, FMT_MTD_CIMC.4 (TSF private key confidentiality protection) and FMT_MTD_CIMC.5 (TSF secret key confidentiality protection) are necessary to ensure that an attacker who has modified firmware, software, or backup data can not prevent detection of the modification by computing a new digital signature, keyed hash, or authentication code.

O.Disposal of Authentication Data is provided by **A.Disposal of Authentication Data,** which covers the requirement that authentication data be disposed of properly after access has been removed.

O.Individual accountability and audit records is provided by a combination of requirements. **FIA_UID.1 (Timing of identification) (iterations 1 and 2)** covers the requirement that users be identified before performing any security-relevant operations. **FAU_GEN.1 (Audit data generation) (iterations 1 and 2)** and **FAU_SEL.1 (Selective audit) (iterations 1 and 2)** cover the requirement that security-relevant events be audited while **FAU_GEN.2 (User identity association) (iterations 1 and 2)** and **FPT_STM.1 (Reliable time stamps) (iterations 1 and 2)** cover the requirement that the date and time of audited events are recorded in the audit records along with the 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, can not delete audit logs. Finally, **FAU_SAR.1 (Audit review)** and **FAU_SAR.3 (Selectable audit review)** cover the requirement that the audit records are made available for review so that individuals can be held accountable for their actions.

O.Integrity protection of user data and software is provided by FDP_ITT.1 (Basic internal transfer protection) (iterations 1 and 3) and FDP_SDI_CIMC.3 (Stored public key integrity monitoring and action) which cover the requirement that user data be protected and FPT_TST_CIMC.2 (Software/firmware integrity test) and FPT_TST_CIMC.3 (Software/firmware load test) which cover the requirement that software and firmware be protected. Since data and software are protected using cryptography, FMT_MTD_CIMC.4 (TSF private key confidentiality protection) and FMT_MTD_CIMC.5 (TSF secret key confidentiality protection) are required to protect the confidentiality of the private and secret keys used to protect the data and software.

O.Limitation of administrative access is provided by FDP_ACC.1 (Subset access control) (iterations 1 and 2), FDP_ACF.1 (Security attribute based access control) (iterations 1 and 2), FIA_SOS.1 (Verification of secrets) (iterations 1 and 2), FIA_UAU.1 (Timing of authentication) (iterations 1 and 2), and FIA_UID.1 (Timing of identification) (iterations 1 and 2). FIA_UAU.1 (Timing of authentication) (iterations 1 and 2), FIA_SOS.1 (Verification of secrets) (iterations 1 and 2), and FIA_UID.1 (Timing of identification) (iterations 1 and 2) ensure that Administrators, Operators, Officers, and Auditors can not perform any security-relevant operations until they have been identified and authenticated and FDP_ACC.1 (Subset access control) (iterations 1 and 2) and FDP_ACF.1 (Security attribute based access control) (iterations 1 and 2) ensure that Administrators, Operators, 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) (iterations 1 and 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.Manage behavior of security functions is provided by **FMT_MOF.1 (Management of security functions behavior) (iterations 1 and 2)** which covers the requirement that authorized users be able to configure, operate, and maintain the security mechanisms.

O.Object and data recovery free from malicious code is provided by FPT_TST_CIMC.2 (Software/firmware integrity test) and FPT_TST_CIMC.3 (Software/firmware load test) which cover the requirement that the recovered state is free from malicious code. FDP_CIMC_BKP.1 (CIMC backup and recovery), FDP_CIMC_BKP.2 (Extended CIMC backup and recovery) covers the requirement to be able to recover to a viable state.

O.Procedures for preventing malicious code is provided by FPT_TST_CIMC.2 (Software/firmware integrity test) which ensures that only signed code can be executed and **AGD_OPE.1 (Operational user guidance)** and A.Malicious Code Not Signed which ensure that those who are capable of signing code do not to sign malicious code. It is also supported by FDP_ACF_CIMC.2 (User private key confidentiality protection), FDP_ACF_CIMC.3 (User secret key confidentiality protection), FCS_CKM.4 (Cryptographic key destruction) and FCS_CKM_CIMC.5 (CIMC private and secret key zeroization) which ensure that an untrusted entity can not use a trusted entity's key to sign malicious code.

O.Protect stored audit records is provided by FAU_STG.1 (Protected audit trail storage) (iterations 1 and 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.Protect user and TSF data during internal transfer is provided by FDP_ITT.1 (Basic internal transfer protection) (iterations 1-4) which covers the requirement that user data be protected during internal transfer and FPT_ITT.1 (Basic internal TSF data transfer protection) (iterations 1-4) which covers the requirement that TSF data be protected during internal transfer.

O.Require inspection for downloads is provided by FPT_TST_CIMC.3 (Software/firmware load test) which covers the requirement that downloaded software can not be loaded until it has been signed and by **AGD_OPE.1 (Operational user guidance)**, and A.Malicious Code Not Signed which ensure that those who are capable of signing code do not to sign malicious code.

O.Respond to possible loss of stored audit records is provided by **FAU_STG.4 (Prevention of audit data loss) (iterations 1 and 2)** which covers the requirement that no auditable events, except those taken by the Auditor, can be performed when audit trail storage is full.

O.Restrict actions before authentication is provided by **FIA_UAU.1 (Timing of authentication) (iterations 1 and 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-relevant configuration management is provided by FMT_MSA.3 (Static attribute initialisation) and FMT_MSA.2 (Secure security attributes) which cover the requirement that security attributes have secure values. FMT_MOF.1 (Management of security functions behavior) (iterations 1 and 2) ensures that security-relevant configuration data can only be modified by those who are authorized to do so. O.Security-relevant configuration management is also supported by **AGD_OPE.1 (Operational user guidance)** which covers the requirement that Administrators be provided with documentation describing the configuration management features of the TOE and by A.Competent Administrators, Operators, Officers and Auditors and A.CPS which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated.

O.Time stamps is provided by **FPT_STM.1 (Reliable time stamps) (iterations 1 and 2)** which covers the requirement that the time stamps be reliable.

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. **O.User authorization**

management is also supported by **AGD_OPE.1 (Operational user guidance)** which covers the requirement that Administrators be provided with documentation describing the user authorization management features of the TOE and by **A.Competent Administrators, Operators, Officers and Auditors** and **A.CPS** which ensure that Administrators are competent and are familiar with the CPS under which the TOE is to be operated.

O.React to detected attacks is provided by **FCS_CKM.4 (Cryptographic key destruction)** and **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_AFL.1 (Authentication failure handling)** 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, Officer, or Operator.

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 18 Summary of Security Functional Requirements Dependencies

Component	Dependencies	Which is:
FAU_GEN.1 Audit data generation	FPT_STM.1 Reliable time stamps	Included
FAU_GEN.2 User identity association	FAU_GEN.1 Audit data generation	Included
	FIA_UID.1 Timing of identification	Included
FAU_SAR.1 Audit review	FAU_GEN.1 Audit data generation	Included
FAU_SAR.3 Selectable audit review	FAU_SAR.1 Audit review	Included
FAU_SEL.1 Selective audit	FAU_GEN.1 Audit data generation	Included
	FMT_MTD.1 Management of TSF data	Included
FAU_STG.1 Protected audit trail storage	FAU_GEN.1 Audit data generation	Included
FAU_STG.4 Prevention of audit data loss	FAU_STG.1 Protected audit trail storage	Included
FCO_NRO_CIMC.3 Enforced proof of origin and verification of origin	FIA_UID.1 Timing of identification	Included
FCO_NRO_CIMC.4 Advanced verification of origin	FCO_NRO_CIMC.3	Included
FCS_CKM.1 Cryptographic key generation	FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation	FCS_COP.1 Included
	FCS_CKM.4 Cryptographic key destruction	Included

Component	Dependencies	Which is:
FCS_CKM.4 Cryptographic key destruction	FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 Included
FCS_CKM_CIMC.5 CIMC private and secret key zeroization	FCS_CKM.4 Cryptographic key destruction	Included
	FDP_ACF.1 Security attribute based access control	Included
FCS_COP.1 Cryptographic operation	FCS_CKM.4 Cryptographic key destruction	Included
	FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation	FCS_CKM.1 Included
FDP_ACC.1 Subset access control	FDP_ACF.1 Security attribute based access control	Included
FDP_ACF.1 Security attribute based access control	FDP_ACC.1 Subset access control	Included
	FMT_MSA.3 Static attribute initialization	Included
FDP_ACF_CIMC.2 User private key confidentiality protection	None	
FDP_ACF_CIMC.3 User secret key confidentiality protection	None	
FDP_CIMC_BKP.1 CIMC backup and recovery	FMT_MOF.1 Management of security functions behavior	Included
FDP_CIMC_BKP.2 Extended CIMC backup and recovery	FDP_CIMC_BKP.1 CIMC backup and recovery	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_CIMC_OCSP.1 OCSP basic response validation	None	
FDP_ETC_CIMC.5 Extended user private and secret key export	None	
FDP_ITT.1 Basic internal transfer protection	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Included
FDP_SDI_CIMC.3 Stored public key integrity monitoring and action	None	
FDP_UCT.1 Basic data exchange confidentiality	FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Included
	FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path	FTP_TRP.1 Included in the IT Environment
FIA_AFL.1 Authentication failure handling	FIA_UAU.1 Timing of authentication	Included
FIA_ATD.1 User attribute definition	None	
FIA_SOS.1 Verification of secrets	None	
FIA_UAU.1 Timing of	FIA_UID.1 Timing of identification	Included

Component	Dependencies	Which is:
authentication		
FIA_UID.1 Timing of identification	None	
FIA_USB.1 User-subject binding	FIA_ATD.1 User attribute definition	Included
FMT_MOF.1 Management of security functions behavior	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
	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	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MOF_CIMC.5 Extended certificate revocation list profile management	FMT_MOF.1 Management of security functions behavior	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MOF_CIMC.6 OCSP profile management	FMT_MOF.1 Management of security functions behavior	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MSA.1 Management of security attributes	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
	FMT_SMF.1 Specification of Management Functions	Not Included
FMT_MSA.2 Secure security attributes	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control	FDP_ACC.1 Included
	FMT_MSA.1 Management of security attributes	Included
	FMT_SMR.1 Security Roles	Included (hierarchical to FMT_SMR.2)
FMT_MSA.3 Static attribute initialization	FMT_MSA.1 Management of security attributes	Included
	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
FMT_MTD.1 Management of TSF data	FMT_SMR.1 Security roles	Included (hierarchical to FMT_SMR.2)
	FMT_SMF.1 Specification of Management Functions	Not Included
FMT_MTD_CIMC.4 TSF private key confidentiality protection	None	
FMT_MTD_CIMC.5 TSF secret key confidentiality protection	None	
FMT_MTD_CIMC.6 TSF private and secret key export	None	
FMT_MTD_CIMC.7 Extended TSF private and secret key export	FMT_MTD_CIMC.6	Included
FMT_SMR.2 Restrictions on security roles	FIA_UID.1 Timing of identification	Included
FPT_TST_CIMC.1 Abstract Machine testing	None	
FPT_CIMC_TSP.1 Audit log	FAU_GEN.1 Audit data generation	Included

Component	Dependencies	Which is:
signing event	FMT_MOF.1 Management of security functions behavior	Included
FPT_ITC.1 Inter-TSF confidentiality during transmission	None	
FPT_ITT.1 Basic internal TSF data transfer protection	None	
FPT_STM.1 Reliable time stamps	None	
FPT_TST_CIMC.2 Software/firmware integrity test	FPT_TST_CIMC.1 Abstract Machine testing	Included
FPT_TST_CIMC.3 Software/firmware load test	FPT_TST_CIMC.1 Abstract Machine Testing	Included
FTP_TRP.1 Trusted path	None	

8.3.1.1.1 Justification of Unsupported Dependencies Regarding FTP_ITC.1 or FTP_TRP.1

Component FDP_UCT.1 Basic data exchange confidentiality has a direct dependency on FTP_ITC.1 Inter- TSF trusted channel or FTP_TRP.1 Trusted path that is unmet. This product uses basic encryption to ensure basic data exchange confidentiality. It is unnecessary for this product to require Inter-TSF trusted channel or trusted path for the TOE. Note that FTP_TRP.1 Trusted path is included in the IT Environment requirements.

8.3.1.1.2 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 and Table 7 meet or exceed the requirement for FMT_SMF.1 Specification of Management Functions.

8.3.1.2 Security Assurance Requirements Dependencies

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

Table 19 Summary of Security Assurance Requirements Dependencies

Component	Depends On:	Which is:
ADV_ARC.1	ADV_FSP.1	included (hierarchical to ADV_FSP.4)
	ADV_TDS.1	included (hierarchical to ADV_TDS.3)
ADV_FSP.4	ADV_TDS.1	included (hierarchical to ADV_TDS.3)
ADV_IMP.1	ADV_TDS.3	included
	ALC_TAT.1	included
ADV_TDS.3	ADV_FSP.4	included
AGD_OPE.1	ADV_FSP.1	included (hierarchical to ADV_FSP.4)
AGD_PRE.1	no dependencies	not applicable
ALC_CMC.4	ALC_CMS.1	included (hierarchical to ALC_CMS.4)
	ALC_DVS.1	included
	ALC_LCD.1	included
ALC_CMS.4	no dependencies	not applicable

Component	Depends On:	Which is:
ALC_DEL.1	no dependencies	not applicable
ALC_DVS.1	no dependencies	not applicable
ALC_FLR.2	no dependencies	not applicable
ALC_LCD.1	no dependencies	not applicable
ALC_TAT.1	ADV_IMP.1	included
ATE_COV.2	ADV_FSP.2	included (hierarchical to ADV_FSP.4)
	ATE_FUN.1	included
ATE_DPT.1	ADV_ARC.1	included
	ADV_TDS.2	included (hierarchical to ADV_TDS.3)
	ATE_FUN.1	included
ATE_FUN.1	ATE_COV.1	included (hierarchical to ATE_COV.2)
ATE_IND.2	ADV_FSP.2	included (hierarchical to ADV_FSP.4)
	AGD_OPE.1	included
	AGD_PRE.1	included
	ATE_COV.1	included (hierarchical to ATE_COV.2)
	ATE_FUN.1	included
AVA_VAN.3	ADV_ARC.1	included
	ADV_FSP.4	included
	ADV_TDS.3	included
	ADV_IMP.1	included
	AGD_OPE.1	included
	AGD_PRE.1	included
	ATE_DPT.1	included

8.3.2 Rationale that Requirements are Mutually Supportive

The requirements represented in this PP were developed from a variety of sources. The security requirements work mutually so that each SFR is protected against bypassing, tampering, deactivation, and detection attacks by other SFRs.

8.3.2.1 Bypass

Prevention of bypass is derived as described below:

FIA_UID.1 and FIA_UAU.1 support other functions' allowing user access to data by limiting the actions the user can take prior to identification and authentication. FIA_SOS.1 supports this by offering reasonable assurance that authentication cannot be bypassed (e.g., by guessing).

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1 support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for bypass.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from bypass to those SFRs dependent on that data.

8.3.2.2 Tamper

Prevention of tamper is derived as described below:

FAU_STG.1 protects the integrity of the audit trail.

FCS_CKM.1 and FCS_COP.1 provide for the secure generation and handling of keys, and therefore support those SFRs that may rely on the use of those keys.

FIA_UID.1 and FIA_UAU.1 support other functions allowing user access to data by limiting the actions the user can take prior to identification and authentication. FIA_SOS.1 supports this by offering reasonable assurance that authentication cannot be bypassed (e.g., by guessing).

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1 support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for tampering.

FDP_ETC_CIMC.5 prevents modification errors during export of secret and/or private keys.

FIA_AFL.1 supports all SFRs dealing with authentication by limiting the number of entry attempts, and then mandating an appropriate action to protect the TOE if too many attempts have been made.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from tampering to those SFRs dependent on that data.

8.3.2.3 Deactivation

Prevention of deactivation is derived as described below:

The access control SFP detailed in FDP_ACF.1 along with the other SFRs dealing with access control, provide for rigorous control of allowed data manipulations and thus prevent unauthorized deactivation.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1, support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FPT_TST_CIMC.2 provides for integrity testing to ensure that selected security functions are operational, thus checking for tampering.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing protection from deactivation to those SFRs dependent on that data.

8.3.2.4 Detection

Detection is derived as described below:

The security audit functions, including FAU_GEN.1, FAU_GEN.2, and FAU_SEL.1 provide for the generation of audit data that may be used to detect attempts to defeat specific SFRs or potential misconfiguration that could leave the TOE prone to attack.

FAU_SAR.1 and FAU_SAR.3, support the audit generation SFRs by providing the capability to selectively search the audit records.

FAU_STG.1, and FAU_STG.4 provide for the protection of the audit records.

The management functions, including FMT_MOF.1, FMT_MSA.1, and FMT_MTD.1, support all other SFRs by restricting the ability to change certain management functions to certain specified roles, thus ensuring that other users cannot circumvent these SFRs.

FMT_MSA.2 and FMT_MSA.3 limit the acceptable values for secure data, thus providing detection protection to those SFRs dependent on that data.

FMT_SMR.2 provides for the specification of multiple roles, thus supporting the other detection SFRs.

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 IT Environment Access Control Policy

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

Subjects (human users) will be granted access to objects (data/files) based upon the:

- Identity of the subject requesting access,
- Role (or roles) the subject is authorized to assume,
- Type of access requested,
- Content of the access request, and,
- Possession of a secret or private key, if required.

Subject identification includes:

- Individuals with different access authorizations
- Roles with different access authorizations
- Individuals assigned to one or more roles with different access authorizations

Access type, with explicit allow or deny:

- Read
- Write
- Execute

For each object, an explicit owning subject and role will be identified. Also, the assignment and management of authorizations will be the responsibility of the owner of an object or a role(s), as specified in this ST.

9.2 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 (human users) will be granted access to objects (data/files) based upon the:

- Identity of the subject requesting access,
- Role (or roles) the subject is authorized to assume,
- Type of access requested,
- Content of the access request, and,
- Possession of a secret or private key, if required.

Subject identification includes:

- Individuals with different access authorizations
- Roles with different access authorizations
- Individuals assigned to one or more roles with different access authorizations

Access type, with explicit allow or deny:

- Read
- Write
- Execute

For each object, an explicit owning subject and role will be identified. Also, the assignment and management of authorizations will be the responsibility of the owner of an object or a role(s), as specified in this PP.

10 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.

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.

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.

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.

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.

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.

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 an operator 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.

11 ACRONYMS

ANSI	American National Standards Institute
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
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