Common Criteria Security Target

IDnomic ID CA

Common Criteria Security Target

IDnomic

23/04/2021

Common Criteria Security Target

Version:	7.3	Pages:	58
Document Status:	☐ Draft	☒ Final	
Author:	IDnomic	IDnomic	

Diffusion List:	☒ External	☐ Internal IDnomic
	ANSSI Oppida	

History:				
Date	Version	Author	Comments	Verified by
08/08/2016	0.991	C. Blad	Initial version	A. Duchamp
04/05/2017	1.0	M. Houradi	Replace “OpenTrust” by “IDnomic” Remove “Client supporting XRMP”	A. Duchamp
27/09/2017	2.0	D. Kaczmarek	Add the SFR FCS_CKM.4	M. Houradi
02/10/2017	3.0	M. Houradi	Update Figure 1 “TOE Architecture” Update the evaluated configuration	A. Duchamp
14/06/2018	4.0	M. Houradi	Update the evaluated configuration	A. Duchamp
17/12/2018	5.0	O. Mary	Remove FCS_CKM.1(1), FCS_COP.1(4), FTP_ITC.1 and FTP_TRP.1 and add assumptions accordingly Remove FAU_STG_EXT and FAU_STG.4, replaced by FAU_STG.3 Add missing “ciphered user encryption keys" in list of objects handled by the TSF	A. Duchamp
09/04/2019	6.0	A. Duchamp	Remove FTA_TAB.1, FPT_TST.2, FPT_STM.1, FTA_SSL.3 Remove refinement on TLS traffic in FCS_COP.1(1) Remove capability to check for update in FPT_TUD_EXT.1 Change in FCS_COP.1(5) to cover key recovery operation	
10/05/2019	7.0	OPPIDA	Remove FPT_FLS.1 and FCS_CKM.5 Remove OT.RECOVERY and OT.INTEGRITY_PROTECTION	A. Duchamp
09/07/2019	7.1	A. Duchamp	Review FCS_CKM.1(2) Add refinement in section 5.3.1 Add missing SFR in section 6 compared to 5.5	
08/01/2020	7.2	A. Duchamp	Update Tomcat version in section 1.6	

			Change AVA_VAN.4 into AVA_VAN.3 in section 5.4	
23/04/2021	7.3	A. Duchamp	Update TOE version number in section 1.2	

CONTENTS

FIGURES & TABLES

1 SECURITY TARGET INTRODUCTION

1.1 ST reference

ST author IDnomic

ST title Common Criteria Security Target – IDnomic ID CA

ST version 7.3

1.2 TOE reference

TOE developer IDnomic

TOE name IDnomic ID CA

TOE version number 1.3.7

1.3 TOE overview

1.3.1 TOE type

The TOE is a set of software packages installed on distributed general computing platforms, providing Certification Authority (CA) services.

1.3.2 Usage and major security features of the TOE

IDnomic ID CA is a software that issues and manages public-key certificates. The Certification Authority (CA) is the primary component of a public key infrastructure (PKI), which consists of programs, data formats, procedures, communication protocols, security policies, and public key cryptographic mechanisms working together to enable people in various locations to establish trust through secure communications.

To achieve this goal, the TOE provides the following security services:

- Key generation/storage
- X509 certificate generation and distribution
- Certificate revocation list (CRL) generation and distribution
- Key escrow and recovery
- System management functions (e.g., security audit, configuration management, archive)

1.3.3 TOE users

The users of the TOE are:

- Registration Authorities (RA) components requesting certificates or revocations to the CA.
- Privileged users (administrators, CA operators, auditors, …) having a direct access to the TOE.

In this document, the term non-person entity (NPE) covers all the IT products that interact with the TOE. It includes the RA components and IT components used by the TOE to provided its services (HSM, NTP servers, LDAP directories…).

1.4 TOE description

The TOE is the complete set of software delivered by IDnomic. It is composed of the following software components:

- JavaScript components that will be executed in the user browser;
- Components running in the Application Server performing the core functions of the CA:
    1. the Admin Application Server Component
    2. the Connector Application Server Component
    3. the Batch Application Server Component computing the configured batch operations
    4. the Cryptographic Server (also named HSS: Hardware Security Services) operating the hardware cryptographic module

All the component that are included in the scope of the evaluation are in green in the following figure.

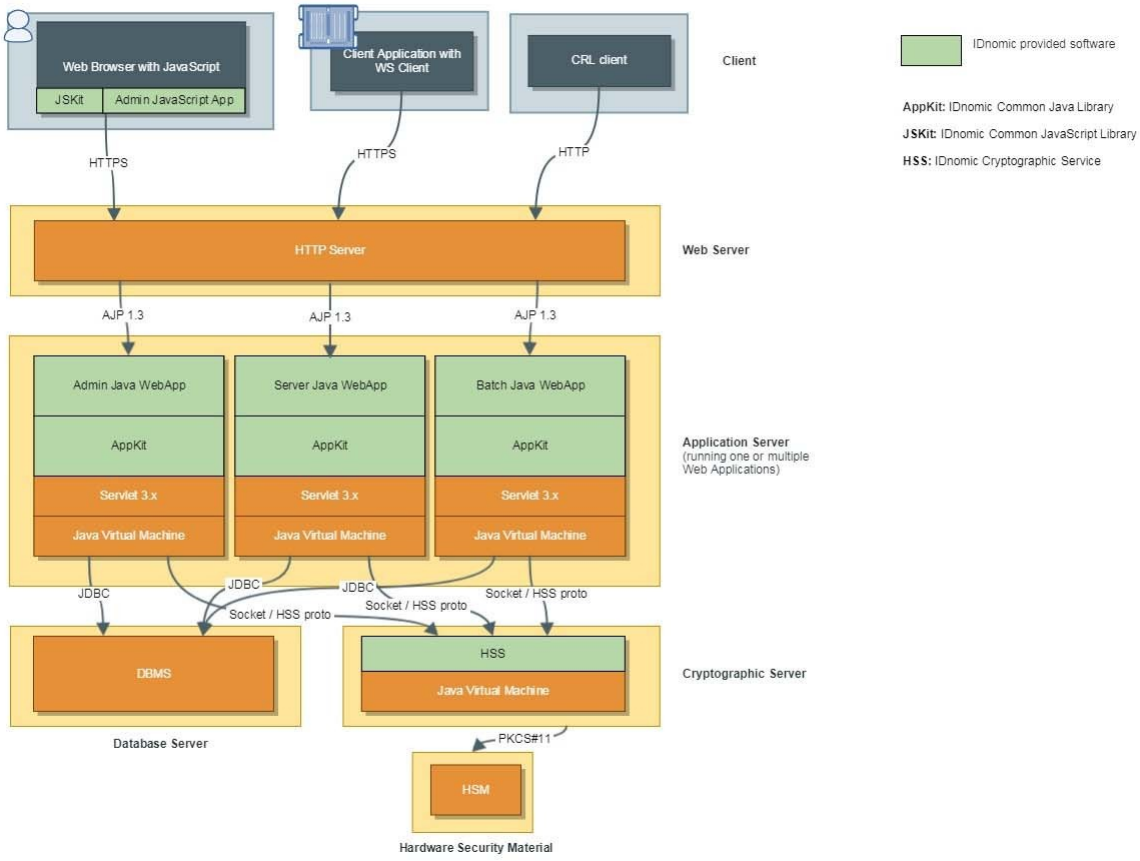


Figure 1: TOE Architecture

1.5 Required non-TOE hardware/software/firmware

The following components are required by the TOE but are out of the scope of the TOE:

- Client supporting SOAP (as described in IDnomic ID CA SOAP Developer Guide) protocols
- Cryptographic module with PKCS#11 interface
- Web Front-ends Software (Apache HTTP Server)

- Application Servers Software (either Apache Tomcat or RedHat JBoss)
- Database Server Software (either Oracle or PostgreSQL)

1.6 Evaluated configuration

The TOE is evaluated in the following configuration:

- N-tiers architecture (Web server, Application server, Database server, Cryptographic server in separated enclaves)
- Operating systems: RHEL version 7.4
- Web servers: Apache httpd version 2.4.6
- Java Virtual Machine: Oracle JRE 1.8 + JCE
- Application server: Apache Tomcat 9
- Database: PostgreSQL 9.4
- Cryptographic module: HSM Bull Proteccio version X147 V149

All these components are necessary for the execution of the TOE but are out of the scope of the evaluation.

2 CONFORMANCE CLAIMS

2.1 CC Conformance Claim

This security target claims conformance to the Common Criteria version 3.1, revision 4, with:

- CC Part 1 strict,
- CC Part 2 extended,
- CC Part 3 strict.

2.2 PP Claim

This security target doesn't claim conformance to a PP. Nevertheless, this security target is based on the PP [PP CA].

2.3 Package Claim

This ST claims conformance to the assurance package EAL4 augmented with ALC_FLR.3.

2.4 Conformance rationale

This security target does not claim compliance with any protection profile.

3 SECURITY PROBLEM DEFINITION

3.1 Assumptions

A.NO_GENERAL_PURPOSE

It is assumed that there are no general purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

This assumption is directly covered by OE.NO_GENERAL_PURPOSE.

A.PHYSICAL

Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.

This assumption is directly covered by OE.PHYSICAL.

A.TRUSTED_ADMIN

TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

This assumption is directly covered by OE.TRUSTED_ADMIN.

A.BACK_UP

TOE Administrators proceed to backup when TOE storage capacity is low in particular when audit records are increasing in size.

This assumption is directly covered by OE.BACKUP.

A.COMSEC

External machines (i.e. administrators workstation, Registration Authority) communicate with the TOE through a secure channel using cipher suites that are compliant with RGS (Référentiel Général de Sécurité – ANSSI). Secure channels are processed by an Apache Httpd Server.

This assumption is directly covered by OE.COMSEC.

A.NTP

Every machine that support components of the TOE have an accurate and synchronized time.

This assumption is directly covered by OE.NTP.

A.OPERATING_SYSTEM

The operating system has been selected to provide the functions required by the TOE.

This assumption is directly covered by OE.OPERATING_SYSTEM.

A.HSM

The HSM (Hardware Security Module) is in charge of generation, protection, deletion, import, export of cryptographic keys. It is also ensured the secure use of these keys based on information provided by the TOE. It has been selected according to its compliance to Common Criteria evaluation to provide functions required by the TOE. Access to private keys are performed with a minimal number of custodian (i.e. 3 out 5)

This assumption is directly covered by OE.HSM.

A.RPM_SIGNATURE

RPMs composing the TOE are digitally signed using a secure component like an HSM or a smartcard. The signature is verified by the OS supporting the TOE.

This assumption is directly covered by OE.RPM_SIGNATURE.

3.2 Threats

3.2.1 Threat agents

The following threat agents have been identified for the security problem definition. External IT entities are all the IT components located in the system where the TOE is operated. It includes both the legitimate components (NPE) interacting with the TOE and any other IT devices.

- Privileged user or legitimate non-person entity (NPE) committing errors affecting the TOE security,
- Malicious person having access to the system or external IT entity intending to circumvent TOE security mechanisms,

- Malicious person having access to the system or external IT entity gaining access to the operating system hosting the TOE,
- Malicious third party attempting to supply a product update,
- Component failure.

3.2.2 Assets to be protected

The TOE shall protect the following assets:

- User data to be protected in confidentiality and integrity. User data are here data associated to subscribers and their key pairs, if generated and stored within the TOE,
- CA signing keys to be protected in confidentiality and integrity,
- Generated certificates to be protected in integrity,
- TOE software update to be protected in integrity,
- Audit data to be protected in integrity.

3.2.3 Undetected System Activity

While several threats are directed at specific capabilities of the TOE, there is also the threat that activity that could indicate an impending or on-going security compromise could go undetected.

Privileged users or non-person entity (NPE) can fail to perform, or can commit errors, in actions that compromise the security provided by the TOE by, for instance, misconfiguring security parameters, permissions, etc. Likewise, users could improperly collect and/or send security-critical data, or accidentally delete data rendering it inaccessible. External NPEs may also deny sending data or information to the TOE or may perform actions that could adversely affect the TOE. Malicious users may also intercept and modify information in transit before it reaches the TOE, attempt to gain access to cryptographic keying material, masquerade as a privileged user, or exploit vulnerabilities in the physical environment, all in attempts to circumvent TOE security mechanisms.

Processing performed in response to user data (for example, the issuance of a certificate in response to an unauthorized or malformed certificate request) may give indications of a failure or compromise of a TOE security mechanism (e.g., issuance of a certificate in response to an invalid request). When indications of activity that may impact the security of the TOE are not generated and monitored, it is possible for harmful activity to take place on the TOE without administrators being aware and able to correct the problem. Further, if no data is kept or records are not generated, reconstruction of the TOE and the ability to understand and remediate the extent of any compromise could be very difficult.

While the TOE generates audit data, these data are not required to be stored on the TOE, but rather should be sent to a trusted external NPE (e.g., a syslog or archive server). These data may be read or altered by an intervening system, thus potentially masking indicators of suspicious activity. It may also be the case that the TOE could lose connectivity to the external NPE, meaning that the audit information could not be sent to the repository.

T.PRIVILEGED_USER_ERROR

A privileged user or non-person entity (NPE) improperly exercises or adversely affects the TOE, resulting in unauthorized services, ineffective security mechanisms, or unintended circumvention of security mechanisms.

Impacted assets: integrity of all assets

The threat is mainly countered by OT.TOE_ADMINISTRATION that requires access control on TOE functions. OT.AUDIT_PROTECTION and OT.AUDIT_LOSS_RESPONSE are required to protect the audit records.

T.UNDETECTED_ACTIONS

A malicious person having access to the system or external IT entity may take actions that adversely affect the security of the TOE.

Impacted assets: integrity and confidentiality of all assets

The threat is mainly countered by OT.SYSTEM_MONITORING that requires to record logs of operations. OT.AUDIT_PROTECTION and OT.AUDIT_LOSS_RESPONSE are required to protect the audit records.

3.2.4 Unauthorized access to the TOE

A CA communicates with a number of different users and other network devices, including:

- Subscribers (or their authorized agents) whose certificates they manage;
- Privileged users (registration authorities, CA operators, Auditors, …)
- Relying parties;
- Other CAs, networks components, or supporting services.

When these communications occur over the network, the endpoints of the communication can be both geographically and logically distant from the TOE, and pass through a variety of other systems. These intermediate systems may be under the control of the adversary, and offer an opportunity for communications with the TOE to be compromised, resulting in possible unauthorized access to the TOE by the adversary.

Some threats to the communication between these endpoints are the same, regardless of the endpoints. Unprotected communication with the CA may allow critical data (such as passwords, configuration settings, sensitive keys or key materials, and service requests or responses) to be read and/or manipulated directly by intermediate systems, leading to a compromise of the CA security functions. Several protocols can be used to provide protection; however, each of these protocols has myriad options that can be implemented and still have the overall protocol implementation remain compliant to the protocol specification. Some of these options can have negative impacts on the security of the connection. For instance, using a weak encryption algorithm can allow an adversary to read and even manipulate the data on the encrypted channel, thus circumventing countermeasures in place to prevent such attacks. Further, if the protocol is implemented with little-used or non-standard options, it may be compliant with the protocol specification but will not be able to interact with diverse relying party equipment that is typically found in large enterprises.

Even though the communication path is protected, there is a possibility that the external entity could be duped into thinking that a malicious third-party user or system is the TOE. For instance, an attacker could

intercept a connection request to the TOE, and respond to the external entity as if it were the TOE. In a similar manner, the TOE could also be duped into thinking that it is establishing communications with an authorized remote entity when in fact it is not. An attacker could also mount a malicious man-in-the-middle-type of attack, in which an intermediate system is compromised, and the traffic is proxied, examined, and modified by this system. This attack can even be mounted via encrypted communication channels if appropriate countermeasures are not applied. These attacks are, in part, enabled by a malicious attacker capturing network traffic (for instance, an authentication session) and "playing back" that traffic in order to fool an endpoint into thinking it was communicating with a legitimate remote entity.

In addition to the threats dealing with the TOE communicating with various external parties that focus on the communications themselves, there are also threats that arise from attempts to gain unauthorized access to the TOE, or the means by which these unauthorized access attempts are accomplished.

For example, if the TOE does not discriminate between administrative users that are allowed to access the TOE interactively (through a locally connected console, or with a session-oriented protocol such as Secure Shell (SSH)) and an administrative user with no authority to use the TOE in this manner, the configuration of the TOE cannot be trusted. Assuming that there is this distinction, there is still the threat that one of the privileged accounts may be compromised and used by an attacker that does not otherwise have access to the TOE.

One vector for such an attack is the use of poor passwords by authorized administrators of the TOE. Passwords that are too short, are easily-guessed dictionary words, or are not changed very often, are susceptible to a brute force attack. Additionally, if the password is plainly visible for a period of time (such as when a legitimate user is typing it in during logon) then it might be obtained by a non-administrative user of the TOE and used to illegitimately access the system.

Once a legitimate privileged user is logged on, there still are a number of threats that need to be considered. During the password change process, if the TOE does not verify that it is the privileged user associated with the account changing the password, then anyone can change the password on a legitimate account and take that account over. If a privileged user walks away from a logged-in session, then another person with no access to the device could sit down and illegitimately start accessing the TOE.

T.UNAUTHORIZED_ACCESS

A malicious person having access to the system or external IT entity intentionally circumvents TOE security mechanisms.

Impacted assets: integrity and confidentiality of all assets

> *The threat is covered by OE.COMSEC that requires the protection of the data assets when they are being transmitted to and from the TOE and OT.TOE_ADMINISTRATION that requires mechanisms to ensure that only privileged users are able to log in and configure the TOE, and provide protections for logged-in users.*

3.2.5 TSF Failure

Security mechanisms of the CA TOE generally build up from a primitive set of mechanisms (e.g., memory management, privileged modes of process execution) to more complex sets of mechanisms. Failure of the

primitive mechanisms could lead to a compromise in more complex mechanisms, resulting in a compromise of the TSF. Furthermore, a CA may be dependent on other, potentially complex, components such as Hardware Security Modules (HSMs), Registration Authorities, and Validation Authorities. Failure of those components could directly or indirectly have a negative impact on the security functions of the CA, it's relying third party systems, or an overall PKI solution.

T.TSF_FAILURE

Security mechanisms of the TOE may fail, leading to a compromise of the TSF. Impacted assets: integrity of all assets

> *The threat is covered by OT.AUDIT_LOSS_RESPONSE that requires* the TOE alerts administrators when audit trail storage is full.

3.2.6 Malicious "Updates"

Since the most common attack vector used involves attacking unpatched versions of software containing well-known flaws, updating CA component firmware and software is necessary to ensure that changes to threat environment are addressed. Timely application of patches ensures that the system is a "hard target", thus increasing the likelihood that product will be able to maintain and enforce its security policy. However, the updates to be applied to the product must be trustable in some manner; otherwise, an attacker can write their own "update" that instead contains malicious code of their choosing, such as a rootkit, bot, or other malware. Once this "update" is installed, the attacker then has control of the system and all of its data.

Even when the cryptographic algorithm is strong and root of trust and intervening CAs are not compromised, there is a legitimate threat that an entity that has obtained a certificate from a trusted CA can perform one or more of the following. These are of concern since the subscriber population under the root of trust could be large, thus increasing the probability of successful attack.

The entity can maliciously sign the updates; the entity may or may not have code signing privileges explicitly.

The entity credentials may not be compromised, but the system the entity uses to exercise the credentials may be compromised to create unauthorized updates.

T.UNAUTHORIZED_UPDATE

A malicious third party attempts to supply a product update that may compromise the security features of the TOE.

Impacted assets: integrity of the TOE software update first then integrity and confidentiality of all assets

> *The threat is covered by OT.VERIFIABLE_UPDATES that requires the TOE to be able to verify the integrity of the update before installation.*

3.2.7 Subscriber Data Disclosure

While most of the threats deal with TSF and administrative data, there is also a threat against subscriber data submitted to CAs that all CAs should mitigate. Data, especially key recovery data, stored at or passing through the TOE could inadvertently be accessed by a different user or NPE; since these data may be sensitive, this may cause a compromise that is unacceptable. The specific threat that must be addressed concerns subscriber data that is not cleared when resources are reallocated; when sensitive values are no longer

needed, access to these data must be prevented. The TOE must ensure that residual data is appropriately handled such that sensitive information is not accessible by other users/processes after it is no longer needed. Data that could be compromised includes authentication data, session keys, security mechanisms, and the data the TOE protects.

T.USER_DATA_REUSE

A malicious person having access to the system or external IT entity gaining access to the operating system hosting the TOE gains access to residual user data that are not cleared or protected from disclosure (i.e. private ciphering key stored for key escrow) when resources are reallocated.

Impacted assets: confidentiality of user data

> *The threat is covered by OT.RESIDUAL_INFORMATION_CLEARING that requires clearing of the sensitive user data after usage and OT.KEY_ESCROW that requires protection of private ciphering keys stored in the TOE for key escrow purpose.*

3.2.8 Dependence on a TOE by Relying Parties

Relying parties within an information system depend on the TOE to accurately bind subjects to their credentials for use in authenticating and providing privacy for transactions. Sensitive applications in the healthcare, finance and government sectors, for example, enforce access rights to resources and services based on these credentials. To meet the expectations of these relying parties, the TOE must be able to issue and manage certificates to a variety of subjects, including human users, network devices, and processes. Without the proper binding, relying parties cannot ensure adequate access controls on sensitive information, ensure transactional integrity, ensure proper accountability and/or enforce non-repudiation.

Furthermore, even when means are available to ensure the authenticity of subject identities, the authentication means might be subject to tampering or other failures that could lead to incorrect authentication. Reliance on the TOE for subject authentication is possible only if that means can be trusted and is interoperable with other components in the environment in which it is placed. Interoperability requires that available standards be used and that CAs be designed to comply with these standards. Trust is not possible without the appropriate physical, policy, and operational controls that are addressed in the subsequent threats.

T.UNAUTHENTICATED_TRANSACTIONS

Relying parties within an information system depend on the TOE to accurately bind subjects to their credentials for use in authenticating and providing privacy for transactions. Without the proper binding provided by the TOE, relying parties cannot ensure adequate access controls on sensitive information, ensure transactional integrity, ensure proper accountability, and/or enforce non-repudiation. This flaw can be caused by a privileged user or legitimate non-person entity (NPE) committing errors or by a malicious person or an external IT entity gaining access to the operating system hosting the TOE.

Impacted assets: integrity of generated certificates

> *The threat is mainly covered by OT.CERTIFICATES that requires the generation of trusted certificates and certificate revocation lists and OT.NON_REPUDIATION that requires the recording of the*

subscribers' requests. In addition, OT.CONFIGURATION_MANAGEMENT requires the TOE to offer functions to manage the active features.

3.2.9 Weak Crypto

The complexity associated with cryptographic methods used to secure communications or provide integrity protections for updates introduces additional threats. For instance, a weak hash function could result in the attacker being able to modify a legitimate update in such a way that the hash remained unchanged. For cryptographic signature schemes, there are dependencies on:

1. the strength of the cryptographic algorithm used to provide the signature, and
2. the ability of the end user to verify the signature (which typically involves checking a hierarchy of digital signatures back to a root of trust (a certification authority)).

If a cryptographic signature scheme is weak, then it may be compromised by an attacker and the administrator will install a malicious update, thinking that it is legitimate. Similarly, if the root of trust can be compromised, then a strong digital signature algorithm will not stop the malicious update from being installed (the attacker will just create their own signature on the update using the compromised root of trust, and the malicious update will then be installed without detection).

T.WEAK_CRYPTO

A weak hash or signature scheme may be compromised by an attacker and used to apply integrity checks to malicious content so that it appears legitimate. This flaw can be caused by a privileged user or legitimate non-person entity (NPE) committing errors or by a malicious person or an external IT entity gaining access to the operating system hosting the TOE.

Impacted assets: integrity of generated certificates

The threat is covered by OE.COMSEC and OT.VERIFIABLE_UPDATES where strong cryptography is required.

4 SECURITY OBJECTIVES

4.1 Security objectives for the TOE

4.1.1 Certificate issuance

The primary purpose of a CA is to issue public key certificates that provide assurance that a public key belongs to its owner by binding the owner's identity to the public key according to identity verification previously performed by a RA (not part of the TOE). This binding is accomplished when the CA digitally signs the public key certificate with its private key; this signature, along with the owner's public key and other identifying information, is contained in the certificate. The binding between owner identity and public key is important for electronic transactions where there is a need to authenticate a subject (i.e., determine that a subject is who they claim to be) before continuing with the transaction or determining what types of transactions are allowed for the subject's authenticated identity. Thus, a CA produces certificates that are used as authenticators to prevent unauthorized access. These certificates can also be used to support integrity assurances (e.g., modifications to a signed message can be detected), confidentiality assurances (e.g., an

encrypted message can be sent such that only the designated recipients can decrypt it), and non-repudiation assurances (e.g., the sender of a signed message cannot deny that he/she is the sender). The CA must support administrative roles that are capable of managing certificate issuance and certificate status functions. The CA must also use its own security mechanisms to ensure its own integrity, its sensitive data (e.g., keys), and its operation are protected. The CA must perform its functions in accordance with a Certificate Policy (CP) and Certification Practice Statement (CPS).

OT.CERTIFICATES

The TSF must ensure that certificates, certificate revocation lists, and certificate status information are valid, and conformant to RGS

The objective is ensured by the following security functional requirements : FDP_CER_EXT.1 defines the content of the certificates; FDP_CER_EXT.2 requires the link between the certificate requests and the issued certificates; FDP_CER_EXT.3 requires the conformity of the certificates with a configured policy; FDP_CSI_EXT.1 and FDP_CRL_EXT.1 require the generation of X.509v2 CRLs; FIA_X509_EXT.1 and FIA_X509_EXT.2 require the validation of the certificates used for users authentication and for update authentication ; FCS_COP.1(1) defines the requirements on the encryption algorithms, FCS_COP.1(2) deals with the generation of the keys used for authentication; FCS_COP.1(3) defines the requirements on the hashing algorithms; and FDP_STG_EXT.1 requires the protection of the defined Trust Anchor database for these authentications, FCS_STG_EXT.1 requires the use of HSM.

OT.NON_REPUDIATION

The TOE will prevent a subscriber from avoiding accountability for sending a message by providing evidence that the subscriber sent the message; and control communications from unknown source.

The objective is ensured by FCO_NRO_EXT.2 that requires to authenticate the requests and FCO_NRR_EXT.2 that requires to log received requests.

OT.CONFIGURATION_MANAGEMENT

The TOE will conduct 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.

The objective is ensured by the following SFRs: FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(4) and FMT_MTD.1 define the management functions accessible to each profile.

4.1.2 Key Escrow

OT.KEY_ESCROW

The TOE will provide a key escrow mechanism, allowing the retrieving of users private ciphering keys.

This mechanism is based on the ciphering of users' private keys by the master key contained in the HSM.

The objective is ensured by the following SFRs: FCS_CKM_EXT.1, FCS_CKM_EXT.2 define the requirement for the protection of the sensitive data in storage, FCS_CKM.4 deals with secure destruction of keys, FCS_COP.1(1) defines the requirements on the encryption algorithms; FCS_COP.1(2) defines the requirements on the signature algorithms, FCS_COP.1(5) defines the protection of encrypted files; FPT_SKP_EXT.1.

4.1.3 Verifiable Updates

Failure by the Administrator to verify that updates to the system can be trusted may lead to compromise of the entire system. A basic approach to establishing trust in the update is to publish a hash of the update that can be verified by the Administrator prior to installing the update. In this way, the Administrator can download the update, compute the hash, and compare it to the published hash. However, the Administrator must confirm the published hash is authoritative and has not been compromised. Digital signatures can convey additional authorizations through the use of extensions such as keyUsage and extendedKeyUsage that can be automatically processed. It is the responsibility of the TOE to ensure these authorizations are correctly processed so only authorized updates are accepted.

OT.VERIFIABLE_UPDATES

The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.

The objective is ensured by the following SFRs: FPT_TUD_EXT.1 requires the existence of a mechanism allowing to verify the integrity of the TOE updates.

4.1.4 System Monitoring

To provide Security Administrators with the necessary information to discover intentional and unintentional issues with the configuration and/or operation of the system, compliant TOEs have the capability of generating audit data targeted at detecting such activity. Auditing of privileged user activities provides information that may hasten corrective action should the system be configured incorrectly. Auditing of select system events can provide an indication of failure of critical portions of the TOE (e.g., a cryptographic provider process not running) or anomalous activity (e.g., establishment of an administrative session at a suspicious time, repeated failures to establish sessions or authenticate to the system) of a suspicious nature, or inappropriate use (e.g., users attempting perform actions without appropriate authorizations). To preserve the integrity of these records, the audit information must itself be protected to prevent unauthorized access, modification, or deletion. The TOE must also be capable of limiting auditable events when the audit trail is full or nearly full.

In some instances, there may be a large amount of audit information produced that could overwhelm the TOE or privileged user in charge of reviewing the audit information. The TOE must be capable of sending audit information to an external trusted entity so that the TOE can continue managing the certificates it issues even if the TOE or TOE environment encounters a failure. This information must carry reliable timestamps, which will help order the information when sent to the external device.

OT.SYSTEM_MONITORING

The TOE will provide the capability to generate audit data and send those data to an external IT entity. The TOE will record in audit records: date and time of action and the entity responsible for the action.

The objective is ensured by the following SFRs: FAU_GEN.1 and FAU_GEN.2 require the generation of audit records; FAU_SAR.1 and FAU_SAR.3 require the TOE to provide the capability to read all information from the audit records; FAU_SEL.1 requires to provide interface to select audited events.

OT.AUDIT_PROTECTION

The TOE will protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.

The objective is ensured by FAU_STG.1 that requires the TOE to protect the audit records.

OT.AUDIT_LOSS_RESPONSE

The TOE will respond to possible loss of audit records when audit trail storage is full or nearly full by alerting administrators and inviting them to save audit files.

The objective is ensured by FAU_STG.3 that requires the TOE to avoid loss of audit records.

4.1.5 TOE Authorized Use

In order to minimize the potential damage caused by an attack against a privileged account, the TOE or TOE environment should partition the privileged functions into security relevant functions and associate the functions by role.

OT.TOE_ADMINISTRATION

The TOE will provide mechanisms to ensure that only privileged users are able to log in and configure the TOE, and provide protections for logged-in users. The TOE will control access to the system by Operators and Administrators who troubleshoot the system and perform system updates. The TOE's certificate management and handling will be controlled by CA Operations Staff. The viewing and maintaining of audit logs will be controlled by Auditor users.

The objective is ensured by the following SFRs: FIA_UAU.2 and FIA_UID.2 require the authentication of the users; FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4) and FMT_MTD.1 define the management functions accessible to each profile; FMT_SMF.1 defines the management functions; FMT_SMR.2 defines the user profiles; FTA_SSL.4 defines the requirements on the user sessions.

4.1.6 Residual Information Clearing

In order to counter the threat of subscriber data disclosure, the TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. The TSF will ensure that any residual information contained in an allocated resource is rendered unavailable upon reallocation.

OT.RESIDUAL_INFORMATION_CLEARING

The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.

The objective is ensured by FDP_RIP.2 that requires the clearing of residual information.

4.2 Security objectives for the TOE operational environment

OE.NO_GENERAL_PURPOSE

There shall not be any general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

OE.PHYSICAL

Physical security, commensurate with the value of the TOE and the data it contains, shall be provided by the environment.

OE.TRUSTED_ADMIN

TOE Administrators shall be trusted to follow and apply all administrator guidance in a trusted manner.

OE.BACKUP

The Toe shall have a backup system allowing administrators to store data externally to the filesystem, in particular the audit records when their size is increasing.

OE.NTP

Machines that support TOE components shall use an accurate and synchronized time

OE.COMSEC

When communicating with external machines, the TOE shall use secure channels with cipher suites that are compliant with RGS (Référentiel Général de Sécurité – ANSSI). Secure channels shall be processed by the Apache server included in the TOE environment.

OE.OPERATING_SYSTEM

The operating system shall be selected to provide the functions required by the TOE.

OE.HSM

The HSM (Hardware Security Module) shall be selected according to its compliance to Common Criteria evaluation to provide functions required by the TOE. Access to private key shall be performed by a minimal number of custodians (i.e. 3 out 5).

OE.RPM_SIGNATURE

RPMs that compose the TOE software shall be electronically signed by an external secure component (HSM or smartcard). The OS supporting the TOE shall verify the signature of the RPMs.

4.3 Security objectives rationale

	A.NO_GENERAL_PURPOSE	A.PHYSICAL	A.TRUSTED_ADMIN	A.BACKUP	A.NTP	A.COMSEC	A.OPERATING_SYSTEM	A.HSM	A.RPM_SIGNATURE	T.PRIVILEGED_USER_ERROR	T.UNDETECTED_ACTIONS	T.UNAUTHORIZED_ACCESS	T.TSF_FAILURE	T.UNAUTHORIZED_UPDATE	T.USER_DATA_REUSE	T.UNAUTHENTICATED_TRANSACTIONS	T.WEAK_CRYPTO
OT.CERTIFICATES																X	X
OT.NON_REPUDIATION																X	
OT.CONFIGURATION_MANAGEMENT																X	
OT.KEY_ESCROW															X		
OT.VERIFIABLE_UPDATES														X			X
OT.SYSTEM_MONITORING											X						
OT.AUDIT_PROTECTION										X	X						
OT.AUDIT_LOSS_RESPONSE										X	X		X				
OT.TOE_ADMINISTRATION										X		X					
OT.RESIDUAL_INFORMATION_CLEARING															X		
OE.NO_GENERAL_PURPOSE	X																
OE.PHYSICAL		X															
OE.TRUSTED_ADMIN			X														
OE.BACKUP				X													
OE.NTP					X												
OE.COMSEC						X						X					X
OE.OPERATING_SYSTEM							X										
OE.HSM								X									
OE.RPM_SIGNATURE									X								

5 SECURITY REQUIREMENTS

The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, with additional extended functional components.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment and selections indicated with italicized text;
- Refinement: Indicated by the word “Refinement”
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3). Extended SFRs are identified by having a label “_EXT” after the requirement name for TOE SFRs.

5.2 Subjects / Operations / Objects

The following subjects are required to operate the TSF:

Subjects	Description	Security attributes
Administrator	Include the roles “Super Administrator”, “Rights Administrator”, “Configuration Administrator”, “Revoke Administrator”	-
Auditor	Include the role “Audit administrator”	-
CA operations staff	Include the roles “Recover Administrator”, “Key Administrator”, “Enroll Administrator”	-
HSS Component	The cryptographic server, HSS = Hardware Security Services	-
RA	Registration Authority	-
Audit server	An external audit server	-

The operations are performed by the TSF:

Subjects	Operations
Administrator	Management functions, update, log export
Auditor	Management functions, read audit records
CA operations staff	Management functions
HSS component	Encryption and decryption of sensitive data stored in the TOE, signature services (certificate and CRL)
RA	Send approved requests to the TOE

Subjects	Operations
Audit server	Receive logs form the TSF

The following objects are handled by the TSF:

Objects	Description	Security attributes
Certificates	Certificate delivered by the TOE	-
CRL	Certificate Revocation List	-
CSR	Certificate Signature Request	-
Public keys	Public keys contained in the CSR	-
Ciphered user encryption keys	User encryption keys handled in encrypted format for escrow service	-

5.3 Security functional requirements

5.3.1 Certificates and CRLs generation

FDP_CER_EXT.1 Extended: Certificate Profiles

FDP_CER_EXT.1.1 The TSF shall implement a certificate profile function and shall ensure that issued certificates are consistent with configured profiles.

FDP_CER_EXT.1.2 The TSF shall generate certificates using profiles that comply with requirements for certificates as specified in IEFT RFC 5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”. At a minimum, the TSF shall ensure that:

a) The version field shall contain the integer 2.

b) The issuerUniqueID or subjectUniqueID fields are not populated.

c) The serialNumber shall be unique with respect to the issuing Certification Authority.

d) 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.

e) The issuer field is not empty.

f) The signature field and the algorithm in the subjectPublicKeyInfo field shall contain the OID for a signature algorithm specified in FCS_COP.1(2).

g) The following extensions are supported:

- subjectKeyIdentifier
- authorityKeyIdentifier
- basicConstraints
- keyUsage
- extendedKeyUsage

- certificatePolicy

h) A subject field containing a null Name (e.g., a sequence of zero relative distinguished names) is accompanied by a populated critical subjectAltName extension.

i) The subjectKeyIdentifier extension is populated with a value unique for each public key contained in a certificate issued by the TSF.

j) The authorityKeyIdentifier extension in any certificate issued by the TOE must be populated and must be the same as the subjectKeyIdentifier extension contained in the TOE's signing certificate.

k) Populated keyUsage and extendedKeyUsage fields in the same certificate contain consistent values.

l) FDP_CER_EXT.1.3 The TSF shall be able to generate at least 20 bits of random for use in issued certificates to be included in serialNumber fields.

Refinement: for the random generation, the TSF uses the HSM.

FDP_CER_EXT.2 Extended: Certificate Request Matching

FDP_CER_EXT.2.1 The TSF shall establish a linkage from certificate requests to issued certificates.

FDP_CER_EXT.3 Extended: Certificate Issuance Approval

FDP_CER_EXT.3.1 The TSF shall support the approval of certificates issued according to a configured certificate profile.

FDP_CSI_EXT.1 Extended: Certificate Status Information

FDP_CSI_EXT.1.1 The TSF shall provide certificate status information whose format complies with ITU-T Recommendation X.509v2 CRL.

FDP_CSI_EXT.1.2 The TSF shall support the approval of changes to the status of a certificate.

FDP_CRL_EXT.1 Extended: Certificate revocation list validation

FDP_CRL_EXT.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:

- If the version field is present, then it shall contain a 1.
- If the CRL contains any critical extensions, then the version field shall be present and contain the integer 1.
- 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.
- The signature and signatureAlgorithm fields shall contain the OID for a digital signature algorithm in accordance with FCS_COP.1(2).
- The thisUpdate field shall indicate the issue date of the CRL.

- The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.

FCO_NRO_EXT.2 Extended: Certificate-based proof of origin

FCO_NRO_EXT.2.1 The TSF shall provide proof of origin for certificates it issues in accordance with the digital signature requirements using mechanism in accordance with RFC 5280 and FCS_COP.1(2).

FCO_NRO_EXT.2.2: The TSF shall provide proof of origin for certificate status information it issues in accordance with the digital signature requirements in CRLs (RFC 5280) and FCS_COP.1(2).

FCO_NRO_EXT.2.3 The TSF shall require and verify proof of origin for certificate requests it receives by authentication of the source.

FCO_NRO_EXT.2.4 The TSF shall require and verify proof of origin for public keys contained in certificate requests it receives via proof-of-possession mechanisms.

FCO_NRO_EXT.2.5 The TSF shall require and verify proof of origin for revocation requests it receives in accordance with authentication of the request origin (RA).

FCO_NRR_EXT.2 Extended: Certificate-based Proof of Receipt

FCO_NRR_EXT.2.1 The TSF shall provide proof of receipt for certificate requests by providing signed responses using mechanisms in accordance with FCS_COP.1(2).

Refinement: answers to certificate generation requests are signed certificates

5.3.2 Certificates usage

FIA_X509_EXT.1 Extended: Certificate Validation

FIA_X509_EXT.1.1: The TSF shall validate certificates in accordance with the following rules:

- IETF RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a certificate in the Trust Anchor Database.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the cA is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using a Certificate Revocation List (CRL) as specified in FDP_CSI_EXT.1.
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3),
    - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field,
    - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.

FIA_X509_EXT.1.2 The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 Extended: Certificate based Authentication

FIA_X509_EXT.2.1 TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for TLS.

FIA_X509_EXT.2.2 When the TSF cannot determine the validity of a certificate, the TSF shall not accept the certificate.

FIA_X509_EXT.2.3 The TSF shall not establish a trusted communication channel if the peer certificate is deemed invalid.

5.3.3 Subscriber data protection

FDP_STG_EXT.1 Extended: Certificate Data Storage

FDP_STG_EXT.1.1 The TSF shall provide access controlled storage for the Trust Anchor Database.

5.3.4 Management

FMT_SMF.1: Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- Ability to manage the TOE locally and remotely;
- Ability to perform updates to the TOE;
- Ability to perform archival and recovery;
- Ability to manage the audit mechanism;
- Ability to configure and manage certificate profiles;
- Ability to approve and execute the issuance of certificates;

Refinement: the approval is done by the RA external component but the TOE process requests only received from this authenticated RA

- Ability to approve certificate revocation;

Refinement: the approval is done by the RA external component but the TOE process requests only received from this authenticated RA

- Ability to modify revocation configuration;
- Ability to configure subscriber self-service request constraints;
- Ability to perform on-demand integrity tests;
- Ability to destroy sensitive user data when no longer needed;
- Ability to import and remove X.509v3 certificates into/from the Trust Anchor Database;
- Ability to configure the NPE ruleset;

- Ability to modify the CRL configuration;
- Ability to configure the cryptographic functionality;

FMT_SMR.2: Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- Administrator,
- Auditor,
- CA Operations Staff.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

- No identity is authorized to assume both an Auditor role and any of the other roles in FMT_SMR.2.1; and
- No identity is authorized to assume both a CA Operations Staff role and any of the other roles in FMT_SMR.2.1

are satisfied.

FMT_MOF.1(1): Management of security functions behavior (Administrators functions)

FMT_MOF.1.1(1) The TSF shall restrict the ability to

- manage the TOE locally and remotely;
- manage the audit mechanism;
- configure and manage certificate profiles;
- modify revocation configuration;
- configure subscriber self-service constraints;
- perform updates to the TOE;
- perform on-demand integrity tests;
- import and remove X.509v3 certificates into/from the Trust Anchor Database;
- configure certificate revocation list function;

to Administrators.

FMT_MOF.1(2): Management of security functions behavior (CA/RA Functions)

FMT_MOF.1.1(2) The TSF shall restrict the ability to

- approve and execute the issuance of certificates;
- configure subscriber self-service request constraints;

- configure automated certificate approval management;
- approve rulesets that govern the authorizations of AORs to manage particular certificates on behalf of an organization;

to CA Operations Staff.

FMT_MOF.1(3): Management of security functions behavior (CA operations Functions)

FMT_MOF.1.1(3) The TSF shall restrict the ability to

- approve certificate revocation;
- approve rulesets that govern the authorizations of RAs to manage particular certificates on behalf of an organization;

to CA Operations Staff.

FMT_MOF.1(4): Management of security functions behavior (Admin Functions)

FMT_MOF.1.1(4) The TSF shall restrict the ability to

- perform archival and recovery;
- perform destruction of sensitive data when no longer needed;

to Administrators.

FMT_MTD.1: Management of TSF Data

FMT_MTD.1.1 The TSF shall restrict the ability to manage the TSF data to Administrators.

5.3.5 Keys generation

FCS_CKM.1(2) Cryptographic Key Generation

FCS_CKM.1.1(2) The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm

- “Digital Signature Standard (DSS)” for RSA schemes
- “Digital Signature Standard (DSS)”, for ECDSA schemes

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits that meet the following:

- FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 and for RSA schemes;
- FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4 for ECDSA schemes and implementing “NIST curves” P-256, P-384 and [selection: P-521, no other curves].

FCS_CKM_EXT.1 Extended: Asymmetric Key Generation for DEKs

FCS_CKM_EXT.1.1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes and specified cryptographic key sizes at least 2048 bits.

FCS_CKM_EXT.2 Extended: Symmetric Key Generation for KEKs (TOE Key Archival)

FCS_CKM_EXT.2.1 The TSF shall be able to generate symmetric KEKs of 128-bit, 256-bit key size for the archival of TOE keys from two or more shares according to a key sharing mechanism.

Refinement: for the key escrow features, the encryption keys of the subscriber key pairs are protected by a master key located in the HSM.

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method rewriting memory with 0 that meets the following: none.

5.3.6 Cryptographic operations

FCS_COP.1(1) Cryptographic Operation (for encryption/decryption)

FCS_COP.1.1(1) The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm AES-CBC and cryptographic key size 128-bit, 256-bit that meet the following: NIST SP 800-38A.

Refinement: encryption of sensitive data stored in the TOE is done by the HSS component and the HSM.

FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)

FCS_COP.1.1(2) The TSF shall perform cryptographic signature services in accordance with a specified cryptographic algorithm

- RSA Digital Signature Algorithm (rDSA)
- Elliptic Curve Digital Signature Algorithm (ECDSA)

and cryptographic key sizes

- 2048 bits or greater (rDSA),
- 256 bits or greater (ECDSA) that meet the following:
- FIPS-PUB 186-4, "Digital Signature Standard", (rDSA)
- FIPS PUB 186-4, "Digital Signature Standard" with "NIST curves" P-256, P-384 and P-521 (as defined in FIPS PUB 186-4, "Digital Signature Standard") (ECDSA)

Refinement: certificate and CRLs signature is done by the HSS component and the HSM

FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)

FCS_COP.1.1(3) The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm SHA-256, SHA-384, SHA-512 and message digest sizes 256, 384, 512 bits that meet the following: FIPS Pub 180-4, “Secure Hash Standard”.

FCS_COP.1(5) Cryptographic Operation (for key recovery)

FCS_COP.1.1(5) The TSF shall perform encryption of the user escrowed private key in accordance with a specified cryptographic algorithm RSA and cryptographic key size 2048 bits or greater that meet the following: RFC 5652 Cryptographic Message Syntax.

5.3.7 Keys protection

FPT_SKP_EXT.1 Extended: Protection of TSF Data (keys)

FPT_SKP_EXT.1.1 The TSF shall implement the ability to prevent reading of all pre-shared keys, private and secret keys (e.g., KEKs, DEKs, session keys).

FCS_STG_EXT.1 Extended: Cryptographic Key Storage

FCS_STG_EXT.1.1 Persistent private and secret keys shall be stored within the TSF in a hardware cryptographic module.

5.3.8 Trusted update

FPT_TUD_EXT.1 Extended: Trusted Update

FPT_TUD_EXT.1.1 The TSF shall implement the ability to provide Administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF shall implement the ability to verify firmware/software updates to the TOE using a digital signature prior to installing those updates.

FPT_TUD_EXT.1.3 The TSF shall implement the ability to verify the digital signature whenever the software or firmware is externally loaded into the TOE and if verification fails, the TSF shall stop the update operation.

5.3.9 Log generation & audit

FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

- Start-up of the audit functions;
- All auditable events for the not specified level of audit; and
- Specifically defined auditable events listed in Table 1.

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

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

- For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of Table 1.

Table 1: Auditable events

Requirement	Auditable events	Additional Audit Record Contents
FDP_CER_EXT.1	Failed certificate generation.	Reason for failure
FDP_CER_EXT.2	None.	None.
FDP_CER_EXT.3	Failed certificate approvals.	Reason for failure.
FDP_CSI_EXT.1	Failure of certificate status information generation.	Reason for failure.
FIA_X509_EXT.1	Failed certificate validations.	None
FIA_X509_EXT.2	Failed authentications.	None
FDP_CRL_EXT.1	Failure to generate CRL	None
FCO_NRO_EXT.2	None	None
FCO_NRR_EXT.2	None	None
FDP_STG_EXT.1	All changes to the trusted public keys, including additions and deletions	The public key and all information associated with the key.
FMT_SMF.1	None.	None.
FMT_SMR.2	Modifications to the group of users that are part of a role.	Modifications to the group of users that are part of a role.
FMT_MOF.1(1) FMT_MOF.1(2) FMT_MOF.1(3) FMT_MOF.1(4)	All modifications in the behaviors of the functions in the TSF.	The old and new values for audit events specified by this function.
FMT_MTD.1	All modifications of the values of TSF data.	The old and new values of the TSF data.
FCS_CKM.1(2)	All occurrences of key generation.	Success: public key generated
FCS_CKM.4	All occurrences destruction. of key	None
FCS_CKM_EXT.1	Failure of symmetric key generation.	None
FCS_CKM_EXT.2	Failure of symmetric key generation.	None
FCS_COP.1(1)	None.	None.

Requirement	Auditable events	Additional Audit Record Contents
FCS_COP.1(2)	All occurrences of signature generation.	Name/identifier of object being signed Identifier of key used for signing.
FCS_COP.1(3)	Failure of hashing function.	None
FCS_COP.1(5)	None.	None
FPT_SKP_EXT.1	None.	None
FCS_STG_EXT.1	None.	None
FPT_TUD_EXT.1	None	None
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_SAR.1	None	None
FAU_SAR.3	None	None
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	None
FAU_STG.1	Any attempt to delete the audit log.	None
FIA_UAU.2	Unsuccessful use of the authentication mechanism	Origin of the attempt (e.g., IP address).
FAU_STG.3	None	None
FIA_UID.2	Unsuccessful use of the user identification mechanism, including the user identity provided	Provided user identity. Origin of the attempt (e.g., IP address).
FTA_SSL.4	The termination of an interactive session.	None
FDP_RIP.2	None	None

FAU_GEN.2 User Identity Association

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_SAR.1 Audit review

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

FAU_SAR.1.2 The TSF 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 TSF shall provide the ability to apply searches of audit data based on the type of event, the subscriber, privileged user or process responsible for causing the event, and the following certificate fields

- subject name associated with the event.

FAU_SEL.1 Selective Audit

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) event type

b) none

Refinement: All the events defined in FAU_GEN.1 are audited in the default configuration of the TOE.

FAU_STG.1 Protected Audit Trail Storage

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

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

Refinement: this protection is assured when storage of the log in the HSS component is activated. Logs are then signed and chained.

FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 The TSF shall provide an alert to administrators if the audit trail will exceed in size the capacity of the TOE media storage capacity.

5.3.10 Identification and authentication

FIA_UAU.2 User authentication before any action

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UID.2 User identification before any action

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

5.3.11 Session management

FTA_SSL.4 User-initiated Termination

FTA_SSL.4.1 The TSF shall allow user-initiated termination of the user's own interactive session. Refinement: this SFR concerns only privileged users.

5.3.12 Residual information clearing

FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from all objects.

5.4 Security assurance requirements

The selected package of security assurance requirements is EAL4 augmented with ALC_FLR.3. List of comments is listed in table below and details are in [CC Part 3].

Table 2: List of security assurance requirements

Assurance Class	Requirements
Development ADV	ADV_ARC.1
	ADV_FSP.4
	ADV_IMP.1
	ADV_TDS.3
Guidance Documents AGD	AGD_OPE.1
	AGD_PRE.1
Life cycle support ALC	ALC_CMC.4
	ALC_CMS.4
	ALC_DEL.1
	ALC_DVS.1
	ALC_FLR.3
	ALC_LCD.1
	ALC_TAT.1
Security Target evaluation ASE	ASE_CCL.1
	ASE_ECD.1
	ASE_INT.1
	ASE_OBJ.2
	ASE_REQ.2
	ASE_SPD.1
	ASE_TSS.1
	ATE_COV.2

Assurance Class	Requirements
Tests ATE	ATE_DPT.1
	ATE_FUN.1
	ATE_IND.2
Vulnerability assessment AVA	AVA_VAN.3

5.5 Security requirements rationale

The detailed rationales are available after each security objective description in section 4.1.

The following table is a summary of the mapping of the SFRs with the security objectives for the TOE.

Table 3: Security requirements rationale

	OT.CERTIFICATES	OT.NON_REPUDIATION	OT.CONFIGURATION_MANAGEMENT	OT.KEY_ESCROW	OT.VERIFIABLE_UPDATES	OT.SYSTEM_MONITORING	OT.AUDIT_PROTECTION	OT.AUDIT_LOSS_RESPONSE	OT.TOE_ADMINISTRATION	OT.RESIDUAL_INFORMATION_CLEARING
FDP_CER_EXT.1	X									
FDP_CER_EXT.2	X									
FDP_CER_EXT.3	X									
FDP_CSI_EXT.1	X									
FIA_X509_EXT.1	X									
FIA_X509_EXT.2	X									
FDP_CRL_EXT.1	X									
FCO_NRO_EXT.2		X								
FCO_NRR_EXT.2		X								
FDP_STG_EXT.1	X									
FCS_CKM_EXT.1				X						
FCS_CKM_EXT.2				X						
FMT_SMF.1									X	
FMT_SMR.2									X	

	OT.CERTIFICATES	OT.NON_REPUDIATION	OT.CONFIGURATION_MANAGEMENT	OT.KEY_ESCROW	OT.VERIFIABLE_UPDATES	OT.SYSTEM_MONITORING	OT.AUDIT_PROTECTION	OT.AUDIT_LOSS_RESPONSE	OT.TOE_ADMINISTRATION	OT.RESIDUAL_INFORMATION_CLEARING
FMT_MOF.1(1)			X						X	
FMT_MOF.1(2)			X						X	
FMT_MOF.1(3)			X						X	
FMT_MOF.1(4)			X						X	
FMT_MTD.1			X						X	
FCS_CKM.1(2)	X									
FCS_CKM.4				X						
FCS_COP.1(1)				X						
FCS_COP.1(2)	X									
FCS_COP.1(3)	X									
FCS_COP.1(5)				X						
FPT_SKP_EXT.1				X						
FCS_STG_EXT.1	X			X						
FTP_TUD_EXT.1					X					
FAU_GEN.1						X				
FAU_GEN.2						X				
FAU_SAR.1						X				
FAU_SAR.3						X				
FAU_SEL.1						X				
FAU_STG.1							X			
FAU_STG.3								X		
FTA_SSL.4									X	
FDP_RIP.2										X

The following table describes the required dependencies.

Table 4: Required dependencies

SFR	Required dependencies	Satisfied dependencies
FDP_CER_EXT.1	FCS_COP.1	FCS_COP.1 (2)
FDP_CER_EXT.2	None	N/A
FDP_CER_EXT.3	FDP_CER_EXT.1	FDP_CER_EXT.1
FDP_CSI_EXT.1	None	N/A
FIA_X509_EXT.1	FDP_CSI_EXT.1	FDP_CSI_EXT.1
FIA_X509_EXT.2	None	N/A
FDP_CRL_EXT.1	FCS_COP.1	FCS_COP.1(2)
FCO_NRO_EXT.2	FCS_COP.1	FCS_COP.1(2)
FCO_NRR_EXT.2	FCS_COP.1 or FIA_CMC_EXT.1 or FIA_EST_EXT.1	FCS_COP.1(2)
FDP_STG_EXT.1	None	N/A
FCS_CKM_EXT.1	None	N/A
FCS_CKM_EXT.2	[FCS_CKM.1 or none]	None
FMT_SMF.1	None	N/A
FMT_SMR.2	FIA_UID.1	FIA_UID.2
FMT_MOF.1(1)	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.2 and FMT_SMF.1
FMT_MOF.1(2)	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.2 and FMT_SMF.1
FMT_MOF.1(3)	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.2 and FMT_SMF.1
FMT_MOF.1(4)	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.2 and FMT_SMF.1
FMT_MTD.1	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.2 and FMT_SMF.1
FCS_CKM.1(2)	(FCS_CKM.2 or FCS_COP.1) and FCS_CKM.4	FCS_COP.1(1) and FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1
FCS_COP.1(1)	(FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1) and FCS_CKM.4	FCS_CKM.1(2) and FCS_CKM.4
FCS_COP.1(2)	(FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1) and FCS_CKM.4	FCS_CKM.1(2) and FCS_CKM.4
FCS_COP.1(3)	(FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1) and FCS_CKM.4	FCS_CKM.1(2) and FCS_CKM.4
FCS_COP.1(5)	(FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1) and FCS_CKM.4	FCS_CKM.1(2) and FCS_CKM.4
FPT_SKP_EXT.1	None	N/A
FCS_STG_EXT.1	None	N/A
FPT_TUD_EXT.1	None	N/A

SFR	Required dependencies	Satisfied dependencies
FAU_GEN.1	FPT_STM.1	N/A Justification: according to A.NTP, the time stamp is provided by the environment.
FAU_GEN.2	FAU_GEN.1 and FIA_UID.1	FAU_GEN.1 and FIA_UID.2
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1
FAU_SAR.3	FAU_SAR.1	FAU_SAR.1
FAU_SEL.1	FAU_GEN.1 and FMT_MTD.1	FAU_GEN.1 and FMT_MTD.1
FAU_STG.1	FAU_GEN.1	FAU_GEN.1
FAU_STG.3	FAU_STG.1	FAU_STG.1
FIA_UAU.2	FIA_UID.1	FIA_UID.2
FIA_UID.2	None	N/A
FTA_SSL.4	None	N/A
FDP_RIP.2	None	N/A

And the SAR dependencies:

Table 5: SAR dependencies

SFR	Required dependencies	Satisfied dependencies
ADV_ARC.1	ADV_FSP.1 and ADV_TDS.1	ADV_FSP.4 and ADV_TDS.3
ADV_FSP.4	ADV_TDS.1	ADV_TDS.3
ADV_IMP.1	ADV_TDS.3 and ALC_TAT.1	ADV_TDS.3 and ALC_TAT.1
ADV_TDS.3	ADV_FSP.4	ADV_FSP.4
AGD_OPE.1	ADV_FSP.1	ADV_FSP.4
AGD_PRE.1	None	N/A
ALC_CMC.4	ALC_CMS.4 and ALC_DVS.1 and ALC_LCD.1	ALC_CMS.4 and ALC_DVS.1 and ALC_LCD.1
ALC_CMS.4	None	N/A
ALC_DEL.1	None	N/A
ALC_DVS.1	None	N/A
ALC_FLR.3	None	N/A
ALC_LCD.1	None	N/A
ALC_TAT.1	ADV_IMP.1	ADV_IMP.1

SFR	Required dependencies	Satisfied dependencies
ASE_CCL.1	ASE_INT.1 and ASE_ECD.1 and ASE_REQ.1	ASE_INT.1 and ASE_ECD.1 and ASE_REQ.2
ASE_ECD.1	None	N/A
ASE_INT.1	None	N/A
ASE_OBJ.2	ASE_SPD.1	ASE_SPD.1
ASE_REQ.2	ASE_OBJ.2 and ASE_ECD.1	ASE_OBJ.2 and ASE_ECD.1
ASE_SPD.1	None	N/A
ASE_TSS.1	ASE_INT.1 and ASE_REQ.1 and ADV_FSP.1	ASE_INT.1 and ASE_REQ.2 and ADV_FSP.4
ATE_COV.2	ADV_FSP.2 and ATE_FUN.1	ADV_FSP.4 and ATE_FUN.1
ATE_DPT.1	ADV_ARC.1 and ADV_TDS.2 and ATE_FUN.1	ADV_ARC.1 and ADV_TDS.3 and ATE_FUN.1
ATE_FUN.1	ATE_COV.1	ATE_COV.2
ATE_IND.2	ADV_FSP.2 and AGD_OPE.1 and AGD_PRE.1 and ATE_COV.1 and ATE_FUN.1	ADV_FSP.4 and AGD_OPE.1 and AGD_PRE.1 and ATE_COV.2 and ATE_FUN.1
AVA_VAN.4	ADV_ARC.1 and ADV_FSP.4 and ADV_TDS.3 and ADV_IMP.1 and AGD_OPE.1 and AGD_PRE.1 and ATE_DPT.1	ADV_ARC.1 and ADV_FSP.4 and ADV_TDS.3 and ADV_IMP.1 and AGD_OPE.1 and AGD_PRE.1 and ATE_DPT.1

6 TOE SUMMARY SPECIFICATIONS

This section summarizes the security features of the TOE that permit the satisfaction of the security functional requirements described in section 5.

6.1 Certificates and CRLs issuance

The TOE generates X509 certificates and CRL using profiles that comply with requirements for certificates as specified in IEFT RFC 5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”.

The TOE is a pure CA component. It does not have the role of Registration Authority. The approval of the requests is delegated to this external RA component. The TOE only process requests received from this authenticated RA component. Profiles management then assures that the RA can only generate certificates in his CA hierarchy.

Two modes of operations are possible:

1. The RA transfers the public key included in a Certificate Signature Request (CSR). In this mode, the TOE generates a certificate for this public key;
2. The RA does not provide the public key. In this mode, the TOE generates the subscriber key pair before to generate the certificate for the public key.

In the evaluated configuration, all certificates and revocation requests are signed and are transmitted through a mutually authenticated channel.

> *This function covers the following SFRs: FDP_CER_EXT.1 Extended, FDP_CER_EXT.2 Extended, FDP_CER_EXT.3 Extended, FDP_CSI_EXT.1 Extended, FDP_CRL_EXT.1 Extended, FCO_NRO_EXT.2 Extended, FCO_NRR_EXT.2 Extended, FCS_COP.1(2), FCS_COP.1(3), FDP_RIP.2, FCS_STG_EXT.1, FCS_CKM.1(2)*

6.2 Authentication by certificates

X509v3 certificates are used to support the TLS authentication. Validation of these certificates is done before to grant access.

> *This function covers the following SFRs: FIA_X509_EXT.1 Extended, FIA_X509_EXT.2 Extended, FDP_RIP.2*

6.3 Identification and authentication

All the users of the TOE (RA, administrators and auditors) are authenticated by a certificate-based authentication mechanism.

> *This function covers the following SFRs: FIA_UAU.2, FIA_UID.2, FDP_RIP.2.*

6.4 Access control and role management

The following roles are required by the [PP CA]: Administrator, Auditor, CA operations staff. These PP roles corresponds to the following TOE built-in roles:

1. Administrator: Configuration administrator + Key Administrator
2. Auditor: Audit administrator
3. CA Operation staff: Enroll Administrator + Revoke Administrator + Web Service

This function covers the following SFRs: FMT_SMR.2, FDP_RIP.2

6.5 Session management

The TOE manages the users' sessions. In particular, the TOE terminates the sessions after a configured time period if inactivity.

This function covers the following SFRs: FTA_SSL.4, FDP_RIP.2

6.6 Data protection

The TOE controls the access on the sensitive data: Trust Anchor Database used to validate TLS certificates, pre-shared keys, private and secret keys.

Database integrity mechanisms are activated to assure integrity of data stored in the database.

The TOE implements key escrow features. Subscribers' key pairs can be stored encrypted in the TOE.

This function covers the following SFRs: FDP_STG_EXT.1 Extended, FCS_CKM.4, FCS_CKM_EXT.1, FCS_CKM_EXT.2, FPT_SKP_EXT.1, FCS_STG_EXT.1, FCS_COP.1(1), FCS_COP.1(5), FDP_RIP.2

6.7 Log generation and audit

The TOE generates an audit record of the main security events. All the events defined in FAU_GEN.1 are audited in the default configuration of the TOE. Audit records can be stored:

- In database,
- In files,
- In the HSS components with cryptographic protection mechanisms (signature and chaining).

This multiple storage capability assures a high level of availability of the audit records. The export of the logs is possible by a simple script (database dump and/or secure copy of files to an external system), allowing administrators to backup audit trails when storage capacity is low. An alarm is provided to administrators when such event is detected by the TOE

This function covers the following SFRs: FAU_GEN.1, FAU_GEN.2, FAU_SAR.1, FAU_SAR.3, FAU_SEL.1, FAU_STG.1, FAU_STG.3, FDP_RIP.2

6.8 Management functions

The TOE offers all the necessary functions to configure and to manage the TOE features. Access control is enforced on these functions to restrict management functions to administrators and CA operation staff.

This function covers the following SFRs: FMT_SMF.1, FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MTD.1, FDP_RIP.2, FTA_SSL.4

6.9 Trusted update

The TOE does not support any automatic update. The TOE is distributed through software packages delivered with a delivery note indicating the hash of all files. The hash can be checked before installation of the update.

This function covers the SFR FPT_TUD_EXT.1, FDP_RIP.2.

7 EXTENDED COMPONENTS DEFINITION

7.1 Extended classes

No extended classes are defined.

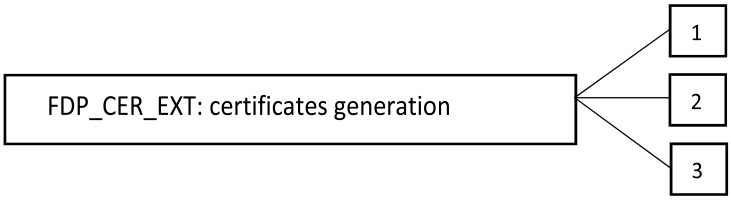
7.2 Extended famillies

7.2.1 Certificates generation (FDP_CER_EXT)

Family Behaviour

This family defines requirements for certificates generation by a Certification Autority.

Component levelling



FDP_CER_EXT.1 defines the content of the certificates.

FDP_CER_EXT.2 requires the link between the certificate requests and the issued certificates.

FDP_CER_EXT.3 requires the conformity of the certificates with a configured policy.

Management: FDP_CER_EXT.1, FDP_CER_EXT.2, FDP_CER_EXT.3

There are no management activities foreseen.

Audit: FDP_CER_EXT.1, FDP_CER_EXT.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Unsuccessful generation of validity evidence

Audit: FDP_CER_EXT.2

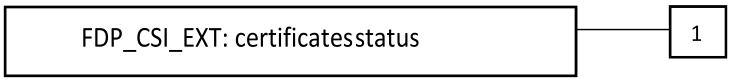
There are no auditable events foreseen.

7.2.2 Certificates status (FDP_CSI_EXT)

Family Behaviour

This family identify requirements for certificates status information.

Component levelling



FDP_CSI_EXT.1 defines the format of the certificate status information.

Management: FDP_CSI_EXT.1

There are no management activities foreseen.

Audit: FDP_CSI_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

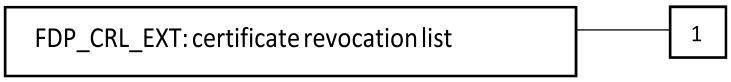
a) Minimal: Unsuccessful generation of validity evidence

7.2.3 Certificate revocation list (FDP_CRL_EXT)

Family Behaviour

This family defines requirements for the format of the certificate revocation list.

Component levelling



FDP_CRL_EXT.1 defines the format of the certificate revocation list.

Management: FDP_CRL_EXT.1

There are no management activities foreseen.

Audit: FDP_CRL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

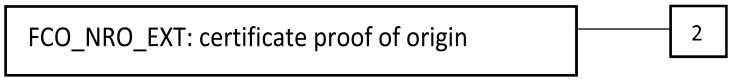
a) Minimal: Unsuccessful generation of validity evidence

7.2.4 Certificate proof of origin (FCO_NRO_EXT)

Family Behaviour

This family defines requirements for verifying the proof of origin for certificate request.

Component levelling



FCO_NRO_EXT.2 defines the proof of origin for the certificate requests.

Management: FCO_NRO_EXT.2

There are no management activities foreseen.

Audit: FCO_NRO_EXT.2

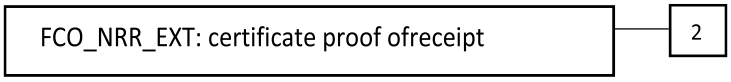
There are no auditable events foreseen.

7.2.5 Certificate proof of receipt (FCO_NRR_EXT)

Family Behaviour

This family identify requirements for verifying the proof of receipt for certificate requests.

Component levelling



FCO_NRR_EXT.2 defines the proof of receipt for certificate requests.

Management: FCO_NRR_EXT.2

There are no management activities foreseen.

Audit: FCO_NRR_EXT.2

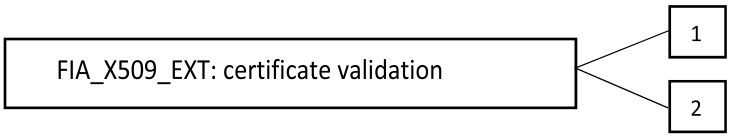
There are no auditable events foreseen.

7.2.6 Certificates usage (FIA_X509_EXT)

Family Behaviour

This family defines requirements for certificates validation.

Component levelling



FIA_X509_EXT.1 defines the rules used to validate certificates. FIA_X509_EXT.2 defines the validation policy.

Management: FIA_X509_EXT.1, FIA_X509_EXT.2 There are no management activities foreseen.

Audit: FIA_X509_EXT.1, FIA_X509_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

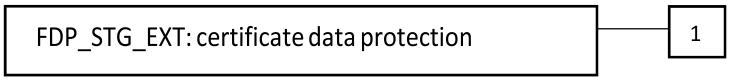
a) Minimal: Unsuccessful certificate validation

7.2.7 Certificate data protection (FDP_STG_EXT)

Family Behaviour

This family provides requirements that address protection of certificate data.

Component levelling



FDP_STG_EXT.1 identify the need to control access to the Trust Anchor Database.

Management: FDP_STG_EXT.1

There are no management activities foreseen.

Audit: FDP_STG_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

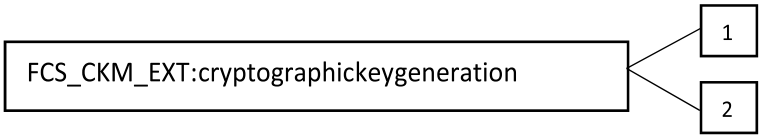
a) Minimal: Unsuccessful generation of validity evidence

7.2.8 Cryptographic key generation (FCS_CKM_EXT)

Family Behaviour

This family intends to define requirements for cryptographic key generation.

Component levelling



FCS_CKM_EXT.1 Cryptographic key generation, requires cryptographic keys used for key establishment to be generated from a specified method.

FCS_CKM_EXT.2 Cryptographic key generation, requires cryptographic keys used for key archival to be generated from a specified method.

Management: FCS_CKM_EXT.1, FCS_CKM_EXT.2 There are no management activities foreseen.

Audit: FCS_CKM_EXT.1, FCS_CKM_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Unsuccessful generation of cryptographic key

7.2.9 Key protection (FPT_SKP_EXT)

Family Behaviour

This family defines requirements for access control of cryptographic keys.

Component levelling

FPT_SKP_EXT.1 defines requirement to forbid read access of cryptographic key.

Management: FPT_SKP_EXT.1

There are no management activities foreseen.

Audit: FPT_SKP_EXT.1

There are no auditable events foreseen.

7.2.10 Key protection (FCS_STG_EXT)

Family Behaviour

This family provides requirements that address protection of private and secret keys.

Component levelling

FCS_STG_EXT.1 identify the need to control access to the private and secret keys.

Management: FCS_STG_EXT.1

There are no management activities foreseen.

Audit: FCS_STG_EXT.1

There are no auditable events foreseen.

7.2.11 Trusted update (FPT_TUD_EXT)

Family Behaviour

This family defines requirements to perform updates in a secure way.

Component levelling

FPT_TUD_EXT.1 defines the update functionality and the integrity verification of updates.

Management: FPT_TUD_EXT.1

There are no management activities foreseen.

Audit: FPT_TUD_EXT.1

There are no auditable events foreseen.

7.3 Extended components

7.3.1 Component FDP_CER_EXT.1

FDP_CER_EXT.1 defines the content of the certificates.

The component FDP_CER_EXT.1 is part of the FDP_CER_EXT family.

Hierarchical to: No other components.

Dependencies: FCP_COP.1.

FDP_CER_EXT.1.1 The TSF shall implement a certificate profile function and shall ensure that issued certificates are consistent with configured profiles.

FDP_CER_EXT.1.2 The TSF shall generate certificates using profiles that comply with requirements for certificates as specified in IETF RFC 5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”. At a minimum, the TSF shall ensure that:

a) The version field shall contain the integer 2.

b) The issuerUniqueID or subjectUniqueID fields are not populated.

c) The serialNumber shall be unique with respect to the issuing Certification Authority.

d) 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.

e) The issuer field is not empty.

f) The signature field and the algorithm in the subjectPublicKeyInfo field shall contain the OID for a signature algorithm specified in FCS_COP.1(2).

g) The following extensions are supported:

- subjectKeyIdentifier
- authorityKeyIdentifier
- basicConstraints
- keyUsage
- extendedKeyUsage
- certificatePolicy

h) A subject field containing a null Name (e.g., a sequence of zero relative distinguished names) is accompanied by an populated critical subjectAltName extension.

i) The subjectKeyIdentifier extension is populated with a value unique for each public key contained in a certificate issued by the TSF.

j) The authorityKeyIdentifier extension in any certificate issued by the TOE must be populated and must be the same as the subjectKeyIdentifier extension contained in the TOE's signing certificate.

k) Populated keyUsage and extendedKeyUsage fields in the same certificate contain consistent values.

FDP_CER_EXT.1.3 The TSF shall be able to generate at least 20 bits of random for use in issued certificates to be included in [selection: serialNumber, notBefore, notAfter] fields.

7.3.2 Component FDP_CER_EXT.2

FDP_CER_EXT.2 requires the link between the certificate requests and the issued certificates.

The component FDP_CER_EXT.2 is part of the FDP_CER_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_CER_EXT.2.1 The TSF shall establish a linkage from certificate requests to issued certificates.

7.3.3 Component FDP_CER_EXT.3

FDP_CER_EXT.3 requires the conformity of the certificates with a configured policy.

The component FDP_CER_EXT.3 is part of the FDP_CER_EXT family.

Hierarchical to: No other components.

Dependencies: FDP_CER_EXT.1

FDP_CER_EXT.3.1 The TSF shall support the approval of certificates issued according to a configured certificate profile.

7.3.4 Component FDP_CSI_EXT.1

FDP_CSI_EXT.1 defines the format of the certificates status information.

The component FDP_CSI_EXT.1 is part of the FDP_CSI_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_CSI_EXT.1.1 The TSF shall provide certificate status information whose format complies with [selection: ITU-T Recommendation X.509v1 CRL, ITU-T Recommendation X.509v2 CRL, the OCSP standard as defined by [selection: RFC 6960, other OCSP standard]].

FDP_CSI_EXT.1.2 The TSF shall support the approval of changes to the status of a certificate.

7.3.5 Component FDP_CRL_EXT.1

FDP_CRL_EXT.1 defines the format of the certificate revocation list.

The component FDP_CRL_EXT.1 is part of the FDP_CRL_EXT family.

Hierarchical to: No other components.

Dependencies: FCS_COP.1

FDP_CRL_EXT.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:

- If the version field is present, then it shall contain a 1.
- If the CRL contains any critical extensions, then the version field shall be present and contain the integer 1.
- 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.
- The signature and signatureAlgorithm fields shall contain the OID for a digital signature algorithm in accordance with FCS_COP.1.
- The thisUpdate field shall indicate the issue date of the CRL.
- The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.

7.3.6 Component FCO_NRO_EXT.2

FCO_NRO_EXT.2 defines requirements for verifying the proof of origin for certificate request.

The component FCO_NRO_EXT.2 is part of the FCO_NRO_EXT family.

Hierarchical to: No other components.

Dependencies: FCS_COP.1

FCO_NRO_EXT.2.1 The TSF shall provide proof of origin for certificates it issues in accordance with the digital signature requirements using a mechanism in accordance with RFC 5280 and FCS_COP.1.

FCO_NRO_EXT.2.2 The TSF shall provide proof of origin for certificate status information it issues in accordance with the digital signature requirements in [selection: CRLs (RFC 5280), OCSP (RFC 6960), [assignment: other OCSP standards]], no other certificate status information] and FCS_COP.1.

FCO_NRO_EXT.2.3 The TSF shall require and verify proof of origin for certificate requests it receives by authentication of the source.

FCO_NRO_EXT.2.4 The TSF shall require and verify proof of origin for public keys contained in certificate requests it receives via proof-of-possession mechanisms.

FCO_NRO_EXT.2.5 The TSF shall require and verify proof of origin for revocation requests it receives in accordance with authentication of the request origin (RA).

7.3.7 Component FCO_NRR_EXT.2

FCO_NRR_EXT.2 defines the proof of receipt for certificate requests.

The component FCO_NRR_EXT.2 is part of the FCO_NRR_EXT family.

Hierarchical to: No other components.

Dependencies: FCS_COP.1 or FIA_CMC_EXT.1 or FIA_EST_EXT.1

FCO_NRR_EXT.2.1 The TSF shall provide proof of receipt for certificate requests by providing signed responses using mechanisms in accordance with [selection: FCS_COP.1, FIA_CMC_EXT.1, FIA_EST_EXT.1].

7.3.8 Component FIA_X509_EXT.1

FIA_X509_EXT.1 defines the rules used to validate certificates.

The component FIA_X509_EXT.1 is part of the FIA_X509_EXT family.

Hierarchical to: No other components.

Dependencies: FDP_CSI_EXT.1

FIA_X509_EXT.1.1 The TSF shall [selection: validate, interface with the Operational Environment to validate] certificates in accordance with the following rules:

- IETF RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a certificate in the Trust Anchor Database.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the cA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [selection: the Online Certificate Status Protocol (OCSP) as specified in FDP_CSI_EXT.1, a Certificate Revocation List (CRL) as specified in FDP_CSI_EXT.1].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3),

- Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field,
- Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.

FIA_X509_EXT.1.2 The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

7.3.9 Component FIA_X509_EXT.2

FIA_X509_EXT.2 defines the validation policy.

The component FIA_X509_EXT.2 is part of the FIA_X509_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_X509_EXT.2.1 The TSF shall [selection: use, interface with the Operational Environment to use] X.509v3 certificates as defined by RFC 5280 to support authentication for [selection: IPsec, TLS, HTTPS, SSH].

FIA_X509_EXT.2.2 When the TSF cannot determine determine the validity of a certificate, the TSF shall [selection: allow the administrator to choose whether to accept the certificate, accept the certificate, not accept the certificate].

FIA_X509_EXT.2.3 The TSF shall not establish a trusted communication channel if the peer certificate is deemed invalid.

7.3.10 Component FDP_STG_EXT.1

FDP_STG_EXT.1 identify the need to control access to the Trust Anchor Database.

The component FDP_STG_EXT.1 is part of the FDP_STG_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_STG_EXT.1.1 The TSF shall provide access controlled storage for the Trust Anchor Database.

7.3.11 Component FCS_CKM_EXT.1

FCS_CKM_EXT.1 requires cryptographic keys used for key establishment to be generated from a specified method.

The component FCS_CKM_EXT.1 is part of the FCS_CKM_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_CKM.EXT.1.1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with [selection:

- NIST Special Publication 800-56A, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” for finite field- based key establishment schemes;

- NIST Special Publication 800-56A, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” for elliptic curve- based key establishment schemes and implementing “NIST curves” P256, P-384 and [selection: P-521, no other curves] (as defined in FIPS PUB 186-4, “Digital Signature Standard”)
- NIST Special Publication 800-56B, “Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography” for RSA-based key establishment schemes]

and specified cryptographic key sizes [assignment: at least 2048 bits].

7.3.12 Component FCS_CKM_EXT.2

FCS_CKM_EXT.2 requires cryptographic keys used for key archival to be generated from a specified method.

The component FCS_CKM_EXT.2 is part of the FCS_CKM_EXT family.

Hierarchical to: No other components.

Dependencies: [FCS_CKM.1 or none, depending of the selection chosen].

FCS_CKM_EXT.2.1 The TSF shall be able to generate [selection: asymmetric KEKs of [assignment: security strength greater than or equal to 112 bits] security strength in accordance with FCS_CKM_EXT.1.1, symmetric KEKs of [selection: 128-bit, 256-bit] key size] for the archival of TOE keys from two or more shares according to a key sharing mechanism.

7.3.13 Component FPT_SKP_EXT.1

FPT_SKP_EXT.1 defines requirement to forbid read access of cryptographic key.

The component FPT_SKP_EXT.1 is part of the FPT_SKP_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_SKP_EXT.1.1 The TSF shall [selection: implement, interface with the Operational Environment to implement] the ability to prevent reading of all pre-shared keys, private, and secret keys (e.g., KEKs, DEKs, session keys).

7.3.14 Component FCS_STG_EXT.1

FCS_STG_EXT.1 identify the need to control access to the private and secret keys.

The component FCS_STG_EXT.1 is part of the FCS_STG_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_STG_EXT.1.1 Persistent private and secret keys shall be stored within the TSF in an hardware cryptographic module.

7.3.15 Component FPT_TUD_EXT.1

FPT_TUD_EXT.1 defines the update functionality and the integrity verification of updates.

The component FPT_TUD_EXT.1 is part of the FPT_TUD_EXT family.

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TUD_EXT.1.1 The TSF shall [selection: implement, interface with the Operational Environment to implement] the ability to provide Administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF shall [selection: implement, interface with the Operational Environment to implement] the ability to verify firmware/software updates to the TOE using a digital signature prior to installing those updates.

FPT_TUD_EXT.1.3 The TSF shall [selection: implement, interface with the Operational Environment to implement] the ability to verify the digital signature whenever the software or firmware is externally loaded into the TOE and if verification fails, the TSF shall [assignment: action to be taken if the verification fails].

8 REFERENCES

CC Part 1	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012
CC Part 2	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components;CCMB-2012-09-002,Version3.1,Revision4,September 2012
CC Part 3	CommonCriteriaforInformationTechnologySecurityEvaluation,Part3:Security Assurance Requirements; CCMB-2012-09-003, Version 3.1, Revision 4, September 2012
PP CA	Protection Profile Certification Authorities, version 1.0, NIAP, 16 May 2014