KEYFACTOR

Security Target for EJBCA Enterprise 9.3.3

Version 1.4

January 2026

Contents

Revision History

The table below shows the revision history of this document.

VERSION	AUTHOR	DESCRIPTION	DATE
0.1	Keyfactor EJBCA	Initial Offering	06/28/2024
0.2	Keyfactor EJBCA	Updated version	10/07/2024
0.3	Keyfactor EJBCA	Minor fixes	10/16/2024
0.4	Keyfactor EJBCA	Update version	11/19/2024
0.5	Keyfactor EJBCA	Minor fixes	12/04/2024
0.6	Keyfactor EJBCA	Updated TOE	12/18/2024
0.7	Keyfactor EJBCA	Updated TOE	03/18/2025
0.8	Keyfactor EJBCA	Updated text per feedback from NIAP	04/07/2025
0.9	Keyfactor EJBCA	Updated text	04/30/2025
1.0	Keyfactor EJBCA	Updated text	05/13/2025
1.1	Keyfactor EJBCA	Updated text	07/02/2025
1.2	Keyfactor EJBCA	Updated text	07/10/2025
1.3	Keyfactor EJBCA	Updated text	10/30/2025
1.4	Keyfactor EJBCA	Updated text	01/14/2026

1 Introduction

1.1 Description of EJBCA

EJBCA is a PKI[1] Certification authority built on JEE technology, allowing the issuance and life cycle management of public key certificates of the type specified in the X.509 v3 [4], C-ITS Enrollment CA IEEE 1609.2, and CVC BSI TR-03110 [3] standards. Additionally, EJBCA can also be set up as a high performance, highly available OCSP responder service, Verification Authority (VA).

EJBCA PKI is an open-source enterprise PKI that can be used stand-alone or integrated in other JEE applications.

Functionalities offered by EJBCA can be used through web interfaces (by end users or TOE users). More information can be found at the project website [6].

The rest of this document describes the EJBCA Target of Evaluation (TOE) that is in the scope of this Common Criteria evaluation and the corresponding Security Target (ST).

1.2 ST Reference

ST Title	Security Target for EJBCA Enterprise 9.3.3
ST Version	1.4
ST Author	Keyfactor
ST Date	01/14/2026

1.3 TOE Reference

TABLE 1.TOE REFERENCE

TOE Developer	Keyfactor
Evaluation Sponsor	Keyfactor
TOE Identification	EJBCA Enterprise 9.3.3
CC Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1 (revision 5)
PP Conformance	Protection Profile for Certification Authorities, Version 2.1, 2017-12-01, National Information Assurance Partnership

1.4 TOE Overview

[1] Public Key Infrastructure.

The usage of Public-key Cryptography relies on the usage of digital certificates, to authenticate relying parties. However, given the complex nature of the issuance and management of the digital certificates lifecycle, organizations that want to carry out those types of operations usually need to use Certification authority applications.

The TOE is a Certification authority (CA). As PKI Certification authority, EJBCA issues and manages the life cycle of public key certificates of the type specified in the X.509 v3 [4], C-ITS Enrollment CA IEEE 1609.2, and CVC BSI TR-03110 [3]standards, allowing the issuance of public key certificates for different purposes, such as:

- Strong authentication for users accessing intranet/extranet/internet resources;
- Secure communication with TLS servers and TLS clients
- Smart card logon to Windows and/or Linux;
- Signing and encrypting email;
- VPN connections by issuing certificates to your VPN routers such as OpenVPN, Cisco, Juniper etc.;
- Client VPN access with certificates in users VPN clients;
- Single sign-on by using a single certificate to secure logon to Web applications;
- Creating signed documents;
- Issue citizen certificates for access to government resources, used in passports etc.;
- Create CVCAs and DVs and issue CV certificates (CVC) to Document Verifiers and Inspection Systems for EU EAC ePassports.
- Issue SSH device and host certificates for strong authentication using the SSH protocol
- Act as the Root or Enrollment CA issing explicit certificates for the C-ITS

Additionally, the TOE also can act as a CA independent OCSP responder service.

Technology wise, given that it is composed of a set of Jakarta EE modules, EJBCA is platform independent.

Regarding its usage, EJBCA is deployed as a regular JEE application, making most of its functionalities available through a set of Web interfaces.

Creation of electronic signatures is a vital part of PKI applications. Electronic signatures can be created in several ways, low level and high level. The TOE will provide means to obtain a private key reference that can be used by relying applications for signing of specific document types. Signatures can be created in cryptographic modules, either using software or hardware (such as HSMs and smart cards).

PKI management systems need to be able to create, process, and manage certificates throughout their lifecycle, including certificate revocation. Revocation information can be distributed and consumed through multiple mechanisms. The TOE supports certificate revocation using standards-based approaches, including generating and processing Certificate Revocation Lists (CRLs), as well as parsing OCSP requests and generating OCSP responses to provide near real-time certificate status information.

The functions for data integrity protection are used to ensure that data, in transit or in storage, cannot be tampered without detection. Integrity protection can be ensured using several techniques, where the most common are message authentication codes and digital signatures.

One very common requirement on sensitive systems is to provide secure audit records. Though creating audit records is simple, ensuring that they are not tampered with is much more difficult. By using the security audit functions of the TOE, an application will be able to create audit trails that meet ETSI EN 319 401 and ETSI EN 319 411 requirements for secure audit.

Authentication and authorization are the most basic security functions needed for an application to provide services to TOE users.

Authentication is the process of identifying the TOE users. Authentication can be performed in many ways, and the TOE provides a framework that can be extended by relying applications in order to meet their specific authentication needs.

Authorization approves or rejects a request for accessing a specific resource. In order to control authorization, the TOE also keeps a database of access rules. The access rules are connected to the authorization system so that TOE user's access to resources can be controlled. Some access rules are already built-in in the TOE, but they can be changed by the relying application.

Additionally, access control is also enforced through role separation, based on a combination of access rules.

The private keys used by the TOE to perform cryptographic operations are kept inside tokens, which can be activated/deactivated to allow/prevent using the keys they hold.

The TOE can generate key pairs for its own usage, kept inside a cryptographic module.

The various security functions of the TOE manage different types of data, including configuration data and recoverable key pairs. Disaster recovery procedures require that it must be possible to restore a security system in a determined state recovered from existing backups. Therefore, the backup functions of the TOE make it possible not only to perform secure backup operations, but also to restore the contents of those backups at another installation. The security functions of the backup make it possible to ensure that the backup, and thus the restored system, cannot be compromised and that confidential data is not revealed.

EJBCA allows the configuration of several CAs in the same TOE instance.

Since, according to [1], the contents of the X.509 certificates and CRLs can be extended to include additional relevant information, the TOE supports the configuration of profiles that define the fields and default values that should be included in the issued certificates and CRLs. For each existing CA, it is possible to configure one CRL profile and one or more certificate profiles.

Issued digital certificates are associated to users, created during the enrollment process. In addition to downloading their certificate(s), authenticated users can regain access to their key pairs kept by the TOE for key recovery purposes (after approval by a TOE user).

Additionally, certain users can be assigned one or more roles that grant them access to specific features of the TOE, like certificate suspension/revocation/activation, key recovery approval, configuration, administration, or user management.

To make them widely available to external users and applications, the TOE supports the configuration of domain-specific publishers that are responsible to relay issued digital certificates and CRLs to third-party repositories where they can be accessed or used.

Besides being able to publish them in the relevant repositories, the TOE also allows the lookup and retrieval of specific certificates and CRLs.

1.4.1 Non-TOE hardware/software/firmware

The rationale for excluding components from the TOE is elaborated in the following sections.

1.4.1.1 Docker

EJBCA is platform-independent, and it can be deployed in Docker containers if the container runs a compliant Java VM and the necessary dependencies for the application server are met. The TOE security functions do not rely on Docker's security features or the container runtime itself.

Deploying EJBCA in a Docker container provides several advantages, such as:

- Simplified deployment and environment consistency across different platforms;
- Isolation of application components, improving resource allocation and scalability;

1.4.1.2 Java Virtual Machine

EJBCA is developed in the Java programming language and, as such, runs in a Java Virtual Machine (JVM). Additionally, since the JVM specifications are public, it can be implemented by independent vendors.

By running on an application server inside a JVM, the TOE is independent of the underlying hardware and software platforms. Therefore, if a fully compliant JVM is available and may be used on such a platform, it should be possible to use the TOE.

Since there are several versions of the JVM specification, Oracle OpenJDK 17.0.16 for Linux has been explicitly chosen for the evaluated configuration.

1.4.1.3 Database

Data persisted by EJBCA is handled by a standard relational database, where the following information is kept:

- Publisher configuration;
- End user registration data, along with their respective enrollment codes;
- Roles and access rules for administrative users;
- Service configuration;
- Approval information (events waiting for approval by TOE users);
- Information about Approval Profiles;
- CA configuration;
- System configuration;
- Configuration pertaining to various protocols and APIs;
- Information about configured downstream peers, e.g. VAs and Ras.
- Certificates and CRLs;
- Basic CA configuration;
- Audit logs of all security relevant operations;
- Authorization data, such as which TOE user is authorized to which resources.

EJBCA enforces access control and maintains integrity of the data for which it is required.

All connections to the database are performed using the appropriate JDBC drivers. Given that it is located in the same platform as the TOE, no specific mechanisms are needed to ensure the integrity and confidentiality of the information transferred to/from the database by the TOE.

Any SQL compliant database can be used. The TOE works, at least, with the following components:

- MariaDB 11.5.2

See section 1.5.2.3 for the evaluated configuration.

1.4.1.4 Hardware Security Module

All cryptographic operations performed at the request of the TOE involving secret and private keys should take place in a cryptographic module, either in software or in hardware. The interaction with the cryptographic module is performed through a standard PKCS#11 library provided by the respective vendor.

Using the PKCS#11 interface makes it possible to use virtually any of the HSMs available on the market. A FIPS 140-2 or -3 validated HSM is recommended.

The TOE works, at least, with the following components:

- SafeNet Luna SA
- Thales DPoD Luna 7 FIPS 140-2 validated (CMVP #4327)

1.4.1.5 Configuration Artifacts

Configuration artifacts are basic TOE configuration items provided by the TOE users. The configuration artifacts define details on how the specific instance of the TOE works and consist of key-value pairs, stored in a configuration file or in a database. Examples of configuration artifacts are PKCS#11 library path for the hardware security module (HSM), key labels for cryptographic keys and modes for secure audit.

However, to run in a CC-certified configuration certain restrictions on the configuration artifacts may apply. Those restrictions are defined in ref [7].

1.4.1.6 Hardware and Operating System

Although the TOE is functionally independent of specific hardware and operating systems and is expected to operate on any platform that provides a reliable time source and can run a JVM, the TOE has a specific dependency on the underlying platform for entropy. In particular, the TOE relies on the operational environment to obtain entropy/seed material used to initialize and reseed the random number generation mechanisms that support the TOE's cryptographic security functions.

The hardware platform is limited to a generic x86 64-bit server for the evaluated configuration.

The TOE has been verified on Red Hat Enterprise Linux 9.4 in the host machine and Alma Linux in the docker container.

1.5 TOE Description

1.5.1 TOE Physical Scope

The TOE is obtained as a Docker container image from customer-specific HTTPS repositories operated by Keyfactor, at the private registry repo.keyfactor.com/images/ejbca-ee.

The EJBCA documentation is available for download to support customers via the Keyfactor Support Portal, which is currently hosted on Zendesk.

The Administrator Guide can also be consulted on the official Keyfactor documentation site at: https://docs.keyfactor.com/ejbca/9.3.3/.

Both documents, the CC Supplementary Guidance and the Administrator Guide, will be available on the NIAP web page associated with this TOE.

Item	Identifier	Format
CC supplementary Guidance	EJBCA Enterprise 9.3.3 Common Criteria Guidance Supplement v1.0	PDF
Administrator operational guidance	EJBCA 9.3.3 Administrator Guide v1.0	PDF
TOE	EJBCA Enterprise 9.3.3	Docker container

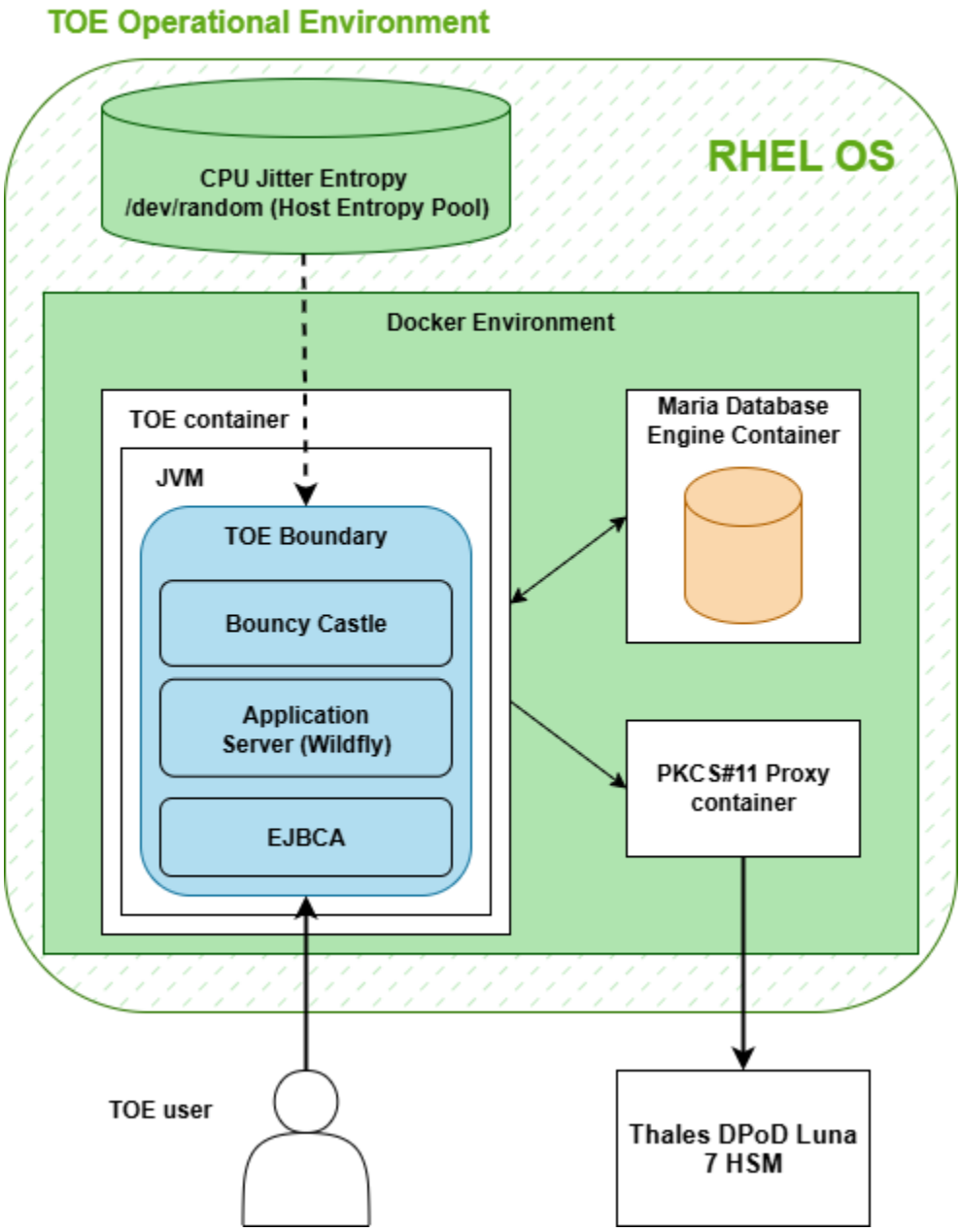
As illustrated by Figure 1, the TOE includes:

- The EJBCA component;
- Application server (WildFly)

Excluded from the TOE is:

- Hardware and operating system platform (abstract machine);
- Execution environment (Java Virtual Machine);
- Hardware security module (HSM); and
- Database engine

FIGURE 1 TOE BOUNDARY



As illustrated in Figure 1, the TOE establishes the following communications:

- The TOE acts as a server for TOE users, who interact with it via web interfaces using HTTPS/TLS channels.
- The TOE establishes a JDBC connection with a MariaDB database instance. This connection is protected within the confines of the operating system and container environment.
- The TOE interacts with the PKCS#11 proxy module via standard PKCS#11 API calls. This connection is protected within the confines of the operating system and container environment.

Excluded from the TOE scope:

- The PKCS#11 proxy container establishes a PKCS#11 protocol channel with the Thales DPoD Luna 7 HSM, which is considered part of the operational environment.

1.5.2 TOE Components

1.5.2.1 EJBCA

The EJBCA component consists of a set of Java classes that provide such functionalities as:

- Create digital certificates and CRLs;
- OCSP support;
- Certification authority management;
- Profile management;
- User registration and management;
- Certificate and CRL publishing;
- Approval profiles for enrollment, key recovery, and revocation;
- Backup of TOE data:
- Bouncy Castle for cryptographic operations

1.5.2.2 Application Server - WildFly

EJBCA is independent of the application server and execution environment where it is used, as long as the execution environment is a compliant Java VM and the application server implements the Enterprise Java Beans (EJB) standard.

EJBCA is deployed on a JEE 10 compliant application server, which provides a number of resources and services to EJBCA, namely:

- Database connectivity services (e.g. object mappings and connection pooling);
- Component creation and management (e.g. session bean pooling and life-cycle management);
- Communication interfaces (e.g. HTTPS and JEE).

These resources and services not only make development and maintenance more efficient, but also enable high performance, scalability and availability.

EJBCA should run on any JEE 10 certified application server. The TOE works, at least, with the following components:

- JBoss EAP
- WildFly

See section 1.5.2.3 for the evaluated configuration.

1.5.2.3 Evaluated Configuration

TABLE 2. EVALUATED CONFIGURATION

Part	Description
Abstract Machine	Dell EMC PowerEdge R440 with Xeon Silver 4216
Abstract Machine Operating System	Red Hat Enterprise Linux 9.4 (kernel: 5.14.0-427.13.1.el9_4) RHEL Kernel CPU Time Jitter RNG Entropy Source version 2.2.0 (ESV E#54)
Container	Docker version 27.1.2
Container Operating System	Alma Linux 9.6
JEE 10 compliant application server	Wildfly 35.0.1
Java Virtual Machine (JVM)	Oracle OpenJDK 17.0.16
Relational Database	MariaDB 11.5.2
HSM	HSM Thales DPoD Luna 7 FIPS 140-2 validated (CMVP #4327 and ESV E#98), using the SafeNet Accelerated Cryptographic Library validated under ACVP (SHS 4533, A1170, A11712, RSA 3042, ECDSA 1526)

1.5.3 TOE Logical Scope

The EJBCA TOE comprises all the security functions required by a Certification Authority, allowing the issuance of public key certificates and CRLs, the lifecycle management of those certificates and capability to provide real-time information about their revocation status, according to the OCSP protocol. Additionally, the TOE depends on several external components for its operation.

Though the security functions can be used independently of each other, the implementation of some functions depends on others. For example, the secure audit security function depends on data integrity protection and electronic signatures creation.

1.5.3.1 Security Audit

One very common requirement on sensitive systems is to provide secure audit records. Though creating audit records is simple, ensuring that they are not tampered with is much more difficult. By using the security audit functions of the TOE, an application will be able to create audit trails that meet ETSI EN 319 401 and ETSI EN 319 411 requirements for secure audit.

The TOE generates the audit records listed in Table 12, Table 13 and Table 14. The event formatting is described in the guidance documentation. The TOE can associate each auditable event with the identity of the user that caused the event. It provides the ability to apply searches of audit data based on serial

numbers associated with the event. The TOE provides an 'Auditors' group and stipulates that an auditor can view the signed audit logs.

The TOE provides secure storage of audit events and further provides separate audit storage for certificate-related events. The TOE provides no administrator or auditor method for deletion or removal of events, and the TOE generates an error message in the event of an error that prevents the TOE from creating new audit records.

The certificates and CRLs repository is provided by the database in the operational environment. PKCS#11 is used as interface. Audit events are stored in both the operating system and the database on the TOE platform.

1.5.3.2 Communication

The TOE provides certificate-based proof of origin for the certificates it issues by digitally signing each certificate in accordance with RFC 5280. In addition, the TOE provides proof of origin for the certificate status information it issues by digitally signing CRLs (RFC 5280) and OCSP responses (RFC 6960). The TOE uses the HSM to perform the digital signature operations for certificates, CRLs, and OCSP responses using the signature algorithm configured for the issuing CA key (e.g., RSA or ECDSA with the corresponding hash).

1.5.3.3 Cryptographic Support

The TOE relies upon the HSM in the operational environment for cryptographic operations involving persistent private keys, such as key-pair generation, certificate signing, OCSP responses, and protocol response messages (e.g., EST), performed through the PKCS#11 API using calls like C_GenerateKeyPair and C_Sign.

Secure communications are managed by the TOE's WildFly application server, which establishes HTTPS/TLS channels by integrating the Bouncy Castle cryptographic module as its JSSE provider. This architecture employs a hybrid cryptographic model to optimize both performance and security.

- The Bouncy Castle module handles the high-volume TLS session operations in software. This includes the negotiation of cipher suites, symmetric encryption (e.g., AES-GCM), hashing for message integrity (HMAC), and the generation of ephemeral keys for key establishment. The Bouncy Castle HMAC-DRBG(SHA-512) is used for this purpose, and in the evaluated configuration, it is seeded with entropy obtained from the RHEL host operating system's /dev/random pool. This pool is continuously supplied by the Kernel CPU Time Jitter RNG, which is the entropy source validated under ESV Certificate #E54.

- In contrast, all cryptographic operations requiring the server's persistent private key, such as the digital signature performed during the TLS handshake to authenticate the server, are delegated to the Thales DPoD Luna 7 HSM (CMVP #4327 and ESV E#98) using the SafeNet Accelerated Cryptographic Library validated under ACVP (SHS 4533, A1170, A11712, RSA 3042, ECDSA 1526) via a PKCS#11 provider. The PKCS#11 provider is the Java software component that connects EJBCA to the HSM's PKCS#11 interface, so EJBCA's key generation and signing requests are executed in the HSM via PKCS#11 calls such as C_GenerateKeyPair and C_Sign. The server's persistent TLS authentication key is generated and stored securely within the HSM and never leaves its boundary.

1.5.3.4 User data protection

The TOE provides certificate profile functionality and certificate generation services conforming to IETF RFC 5280. A CA account can have one or more profiles which are configured by an administrator using the TOE's web interface. Using that interface the administrator can assign a name (Certificate Profile ID), extensions, and default properties to the profile.

The TOE supports the approval of certificates issued according to a configured certificate profile through the web interface, the RA management interface or EST. Only user roles given 'Approve End Entity Actions' permission can approve certificates via the web interface (which is the only interface through which manual issuance occurs).

The TOE provides certificate status information through CRLs and OCSP responses. The TOE clears sensitive data from buffers before releasing the buffers.

The TOE does not store any personally identifiable information, that does not also appear in a certificate. The TOE handles the TLS session objects.

The TSF, in conjunction with the HSM, ensures that sensitive data is securely protected or erased when it is no longer needed, so that any previous content cannot be recovered. This applies to secret and ephemeral keys.

1.5.3.5 Identification and authentication

The TOE supports Enrollment over Secure Transport (EST) protocol as described in RFC 7030 to receive and act upon certificate enrollment requests using the simple enrollment method described in RFC 7030 Section 4.2.

Certificate enrollment requests are authenticated using an existing certificate and corresponding private key as specified by RFC 7030 Section 3.3.2.

The TOE provides a certificate-based mechanism via web interface. For certificate authentication the TOE uses the HSM to validate certificates. The TOE uses X.509v3 certificates, as defined by RFC 5280, to authenticate privileged users, subscribers, RAs, and DBAccess clients over HTTPS. Certificates can also be used for EST access authentication.
When the TSF cannot determine the current revocation status of a certificate the administrator is allowed to choose whether to accept the certificate or not.

The TSF allows the following actions prior to requiring a non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- Obtain certificate status information;
- Public information retrieval requests.

1.5.3.6 Security Management

The TOE provides all the interfaces necessary to manage the security functions identified throughout this Security Target as well as other functions commonly found in certificate authorities.

Administrative roles are fully configurable. The TOE provides default role templates corresponding to the roles defined in the Protection Profile, ref. 8. The corresponding TOE role names are:

- Administrator – Super Administrator
- Auditor – Auditor
- CA Operations Staff – CA Administrators

- RA Staff - RA Administrators
- Authorized Organizational Representative - Supervisor

1.5.3.7 Protection of the TSF

The TOE protects itself after a secure component failure that will result in an error message giving the Administrator possibility to perform a secure restart. The TOE utilizes a HSM and relies upon the HSM to secure and protect the keys stored by the TOE in the HSM, and to offer services to allow operations using the HSM protected certificates.

The TOE obtains the current time from its operational environment.

1.5.3.8 TOE Access

The TOE allows remote users to terminate their interactive session. The TOE also has the capability to display an advisory message (banner) when users access the TOE for use.

1.5.3.9 Trusted path/channels

The TOE protects interactive communication with administrators on the HTTPS (WebUI) interface and both integrity and disclosure protection are ensured. If the negotiation of an encrypted session fails or if the user does not have authorization for remote administration, the attempted connection will not be established.

1.5.4 Excluded Functionality

The TOE shall be deployed in a general computing platform. Deployment models that include hosting on, or delivering service from cloud environments are excluded from the evaluated configurations.

2 Conformance Claims

2.1 CC Conformance Claim

The TOE conforms to:

- Common Criteria for Information Technology Security Evaluation, Version 3.1 (revision 5), part 2 extended;
- Common Criteria for Information Technology Security Evaluation, Version 3.1 (revision 5), part 3 conformant

2.2 PP Conformance Claim

The ST demonstrates exact conformance to the following Protection Profile (PP):

- Protection Profile for Certification Authorities, Version 2.1, 2017-12-01, National Information Assurance Partnership, ref. 8.

2.3 Applicable Technical Details

The following technical decisions are applicable to the PP_CA_V2.1:

- TD0946 - Adding FIPS 186-5 in PP_CA_V2.1. Applicable
- TD0932 - Clarification when CTR_DRBG is Selected for FCS_RBG_EXT.1.2 in PP_CA_V2.1. Not applicable, the TOE does not claim CTR-DRBGs.
- TD0896 - Applicability of Containers to FPT_TUD_EXT.1. Applicable
- TD0866 - Removal of Obsolete Parts of TLSS.1.1 Test 4. Applicable.
- TD0845 - Addition of File-Based Protocols to FIA_CMCS_EXT.1.3. Not applicable, FIA_CMCS_EXT.1 is not claimed in this security target.
- TD0844 – Addition of Assurance Package for Flaw Remediation v1.0 Conformance Claim. Applicable.
- TD0796 – Removal of SHA-1 from Various Selections. Applicable.
- TD0782 – Terminology Change in CAPP: Extended to Functional Package. Applicable.
- TD0599 – Corrections to SAR Section in CAPP. Applicable
- TD0522 – Updates to Certificate Revocation (FIA_X509_EXT.1). Applicable.
- TD0500 – Cryptographic selections and updates for CAPP. Applicable.
- TD0415 – Trusted Update Test 4 Conditional. Applicable.
- TD0375 – FMT_MOF.1(4) selection. Applicable
- TD0353 – Guidance for Certificate Profiles. Applicable.
- TD0348 – FCS_TLSS_EXT.2.4 for TLS 1.2 or higher. Applicable.
- TD0328 – Split Knowledge Procedures distinction. Not applicable, the TOE does not claim the SFRs.
- TD0294 – Correction of TLS SFRs in CA PP ver 2.1. Applicable
- TD0287 – FAU_STG.4 Testing. Applicable.
- TD0286 – Audit Events for FPT_RCV.1. Applicable.
- TD0278 – Clarification of Role for Managing Manual Certificate Requests. Applicable

- TD0276 – X.509 Code Signing on TOE Updates. Not applicable, the TSF does not support authentication for code signing for TOE updates.

2.4 Package claims

Assurance Package for Flaw Remediation, Version 1.0, June 28, 2024 – conformant (TD0844 applied).

2.5 Conformance Rationale

All the assumptions, threats, policies, objectives and security requirements defined for Protection Profile for Certification Authorities have been reproduced in this ST. No additional assumption, threat, policy, objective or security requirement has been used.

Table 3 lists all security function requirements that have or have not been included. M – Mandatory, O – Optional, S – Selectable.

TABLE 3.SECURITY FUNCTIONAL REQUIREMENTS

Security Functional Requirement	M	O	S	Incl.	Rationale
FAU_ADP_EXT.1 Audit Dependencies	X			Y	Mandatory
FAU_GCR_EXT.1 Generation of Certificate Repository	X			Y	Mandatory
FAU_GEN.1 Audit Data Generation	X			Y	Mandatory
FAU_GEN.2 User Identity Association	X			Y	Mandatory
FAU_SAR.1 Audit Review			X	Y	Audit review is performed by an auditor through an interface provided by the TSF.
FAU_SAR.3 Selectable Audit Review			X	Y	Audit review is performed by an auditor through an interface provided by the TSF.
FAU_SCR_EXT.1 Certificate Repository Review			X	N	The ability to search on certificate fields provided entirely by the OE OE.CERTIFICATE_REPOSITORY_SEARCH objective is included instead.
FAU_SEL.1 Selective Audit			X	N	The TOE interfaces do not provide the capability to pre-select the audit records.
FAU_STG.1(1) Protected Audit Trail Storage			X	N	Audit records are not stored within the TOE boundary but in the database and the underlying operating system on the TOE platform.

FAU_STG.1(2) Protected Audit Trail Storage (Archive Data)			X	N	No audit data stored within the TOE boundary is expected to persist intact beyond the validity of certificates issued by the CA.
FAU_STG.4 Prevention of Audit Data Loss	X			Y	Applies to the TOE regardless of whether the audit trail is stored within the TOE boundary (e.g. the audit trail is full) or on an external system in the Operational Environment (e.g. the connection to a remote audit repository is broken).
FAU_STG_EXT.1 External Audit Trail Storage			X	Y	The TSF initiates the storage of the audit data (that is, it generates audit data that will be stored by the OE).
FAU_STG_EXT.2 Audit Data Retention			X	N	The Operational Environment provides mechanisms for retention of audit records.
FCO_NRO_EXT.2 Certificate-Based Proof of Origin	X			Y	Mandatory
FCO_NRR_EXT.2 Certificate-Based Proof of Receipt			X	Y	EST is claimed.
FCS_CDP_EXT.1 Cryptographic Dependencies	X			Y	Mandatory
FCS_CKM.1 Cryptographic Key Generation			X	Y	The Operational Environment is used to generate the persistent keys. The TOE uses its Bouncy Castle cryptographic library during TLS key exchange.
FCS_CKM.2 Cryptographic Key Establishment			X	Y	The TOE perform key establishment.
FCS_CKM_EXT.1(1) Symmetric Key Generation for DEKs			X	N	FDP_SDP_EXT.1 is not included.
FCS_CKM_EXT.1(2) Key Generation Key Encryption Keys			X	N	Keys are stored in a hardware cryptographic module.
FCS_CKM_EXT.1(3) Key Generation for Key Encryption Keys (TOE Key Archival)			X	N	FPT_SKY_EXT.1 is not included.
FCS_CKM_EXT.1(4) Generation of Key Shares			X	N	Used by FPT_SKY_EXT.1 is not included.
FCS_CKM_EXT.4 Cryptographic Key Destruction			X	Y	The Java Virtual Machine in the TOE platform and the external HSM destruct key material and CSP.
FCS_CKM_EXT.5 Public Key Integrity			X	N	FDP_STG_EXT.1 is not included.

FCS_CKM_EXT.6 TOE Key Archival			X	N	Key sharing mechanism is included in FPT_SKY_EXT.1 Split Knowledge Procedures
FCS_CKM_EXT.7 Key Generation for KEKs			X	N	The Key Encryption Key generation is performed in a hardware cryptographic module.
FCS_CKM_EXT.8 Key Hierarchy Entropy			X	N	The entropy is provided by the operational environment.
FCS_COP.1(1) Cryptographic Operation (AES Encryption/Decryption)			X	Y	AES is used for HTTPS/TLS.
FCS_COP.1(2) Cryptographic Operation (Cryptographic Signature)			X	Y	Digital signatures are used (e.g. HTTPS/TLS) and CA operations.
FCS_COP.1(3) Cryptographic Operation (Cryptographic Hashing)			X	Y	Hashing is used (e.g. HTTPS/TLS).
FCS_COP.1(4) Cryptographic Operation (Keyed-Hash Message Authentication)			X	Y	Keyed mac is used (e.g. HTTPS/TLS).
FCS_COP.1(5) Cryptographic Operation (Password-Based Key Derivation Function)		X		N	FPT_SKY_EXT.2 is not included.
FCS_HTTPS_EXT.1 HTTPS Protocol			X	Y	HTTPS/TLS is used for administration GUI and RA GUI.
FCS_IPSEC_EXT.1 IPsec Protocol			X	N	IPSec is not used.
FCS_RBG_EXT.1 Cryptographic Random Bit Generation			X	Y	The TOE invokes the entropy source of the environment to generate random numbers for TLS communications and uses the external HSM for CA operations.
FCS_STG_EXT.1 Cryptographic Key Storage	X			Y	Mandatory
FCS_TLSC_EXT.2 TLS Client Protocol with Mutual Authentication			X	N	The connection with the HSM is out of the scope.
FCS_TLSS_EXT.1 TLS Server Protocol			X	Y	Public information retrieval requests
FCS_TLSS_EXT.2 TLS Server Protocol with Mutual Authentication			X	Y	Used for administration GUI communication.
FDP_CER_EXT.1 Certificate Profiles	X			Y	Mandatory

FDP_CER_EXT.2 Certificate Request Matching	X			Y	Mandatory
FDP_CER_EXT.3 Certificate Issuance Approval	X			Y	Mandatory
FDP_CER_EXT.4 Non-X.509v3 Certificate Generation		X		N	Only X.509 v3 needs to be claimed.
FDP_CRL_EXT.1 Certificate Revocation List Validation			X	Y	CRL validation is supported.
FDP_CSI_EXT.1 Certificate Status Information	X			Y	Mandatory
FDP_ITT.1 Basic Internal Transfer Protection			X	N	The TOE is not a distributed TOE
FDP_OCSPG_EXT.1 OCSP Basic Response Generation			X	Y	OCSP is supported.
FDP_RIP.1 Subset Residual Information Protection	X			Y	Mandatory
FDP_SDP_EXT.1 User Sensitive Data Protection		X		N	User Sensitive Data is not encrypted.
FDP_STG_EXT.1 Public Key Protection		X		N	Public key protection is performed entirely by the DB in the Operational Environment.
FIA_AFL.1 Authentication Failure Handling			X	N	Unsuccessful authentication attempts are not detected.
FIA_CMCC_EXT.1 Certificate Management over CMS (CMC) Client			X	N	CMC is not supported.
FIA_CMCS_EXT.1 Certificate Management over CMS (CMC) Server			X	N	CMC is not supported.
FIA_ESTC_EXT.1 Enrollment over Secure Transport (EST) Client			X	N	Only EST Server not client is included.
FIA_ESTS_EXT.1 Enrollment over Secure Transport (EST) Server			X	Y	EST Server is included.
FIA_PMG_EXT.1 Password Management			X	Y	Password management is supported for end entities.
FIA_PSK_EXT.1 Pre-Shared Key Composition			X	N	Pre-Shared Keys are not supported.
FIA_X509_EXT.1 Certificate Validation	X			Y	Mandatory
FIA_X509_EXT.2 Certificate-Based Authentication	X			Y	Mandatory
FIA_X509_EXT.3 X509 Certificate Request			X	Y	CSR is supported.

FIA_UAU.7 Protected Authentication Feedback			X	N	The TOE implements certificate-based authentication.
FIA_UAU_EXT.1 Authentication Mechanism	X			Y	Mandatory
FIA_UIA_EXT.1 User Identification and Authentication	X			Y	Mandatory
FMT_MOF.1(1) Management of Security Functions Behavior (Administrator Functions)	X			Y	Mandatory
FMT_MOF.1(2) Management of Security Functions Behavior (CA/RA Functions)	X			Y	Mandatory
FMT_MOF.1(3) Management of Security Functions Behavior (CA Operations Functions)	X			Y	Mandatory
FMT_MOF.1(4) Management of Security Functions Behavior (Admin/Officer Functions)	X			Y	Mandatory
FMT_MOF.1(5) Management of Security Functions Behavior (Auditor Functions)	X			Y	Mandatory
FMT_MTD.1 Management of TSF Data	X			Y	Mandatory
FMT_SMF.1 Specification of Management Functions	X			Y	Mandatory
FMT_SMR.2 Restrictions on Security Roles	X			Y	Mandatory
FPT_APW_EXT.1 Protection of Privileged User Passwords			X	N	The TOE implements certificate-based authentication.
FPT_FLS.1 Failure with Preservation of Secure State	X			Y	Mandatory
FPT_ITT.1 Basic Internal TSF Data Transfer Protection			X	N	The TOE is not a distributed TOE
FPT_KST_EXT.1 No Plaintext Key Export	X			Y	Mandatory
FPT_KST_EXT.2 TSF Key Protection	X			Y	Mandatory
FPT_NPE_EXT.1 NPE Constraints		X		N	Not mandatory
FPT_RCV.1 Manual Trusted Recovery	X			Y	Mandatory
FPT_SKP_EXT.1 Protection of Keys	X			Y	Mandatory

FPT_SKY_EXT.1 Split Knowledge Procedures		X		N	Split knowledge procedures are performed entirely in the operational environment.
FPT_SKY_EXT.2 Key Share Access			X	N	Key share access is performed entirely in the operational environment.
FPT_STM.1 Reliable Time Stamps	X			Y	Mandatory
FPT_TST_EXT.1 TOE Integrity Test		X		N	TSF is a virtual application.
FPT_TST_EXT.2 Integrity Test		X		N	The operational environment verifies the integrity of the TOE databases.
FPT_TUD_EXT.1 Trusted Update	X			Y	Mandatory
FPT_TUD_EXT. 2 Integrity for Installation and Update			X	Y	The TOE is distributed as a container image.
FTA_SSL.3 TSF-Initiated Termination		X		N	The TOE does not terminate remote interactive sessions. Session is managed by the browser, which will automatically ended after a period of inactivity.
FTA_SSL.4 User-Initiated Termination	X			Y	Mandatory
FTA_SSL_EXT.1 TSF-Initiated Session Locking		X		N	The TOE does not lock or terminate local interactive sessions.
FTA_TAB.1 Default TOE Access Banners	X			Y	Mandatory
FTP_ITC.1 Inter-TSF Trusted Channel			X	N	No external entities are considered in the evaluated configuration.
FTP_TRP.1 Trusted Path	X			Y	Mandatory

All operations performed on the IT security requirements are within the bounds set by the Protection Profile for Certification Authorities. Assignment and selection operations on security requirements are indicated in chapter 6 .

3 Security Problem Definition

The security problem definition has been taken from the Protection Profile [8] and is reproduced here for the convenience of the reader, it includes the following parts:

- Threats;
- Organizational Security Policies; and
- Assumptions.

3.1 Threats

TABLE 4.THREATS

Threat	Description
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.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
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 nonrepudiation.
T.UNAUTHORIZED_ACCESS	A malicious user, process, or external IT entity intentionally circumvents TOE security mechanisms.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.UNDETECTED_ACTIONS	Remote users or external IT entities may take actions that adversely affect the security of the TOE.
T.USER_DATA_REUSE	A malicious user, process, or external IT entity may gain access to user data that is not cleared when resources are reallocated.
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

3.2 Organizational Security Policies

TABLE 5. ORGANIZATIONAL SECURITY POLICY

Threat	Description

P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

3.3 Assumptions

TABLE 6. ASSUMPTIONS

Assumption	Description
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.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment
A.TRUSTED_ADMIN	TOE Administrators are assumed to follow and apply all administrator guidance in a trusted manner

4 Security Objectives

The security objectives have been taken from the Protection Profile, ref. □, and is reproduced here for the convenience of the reader, it includes the following parts:

- Security objectives for the TOE;
- Security objectives for the operational environment; and
- Security objectives rationale

4.1 Security Objectives for the TOE

TABLE 7. SECURITY OBJECTIVES

Security Objectives for the TOE	Description
O.AUDIT_LOSS_RESPONSE	The TOE will respond to possible loss of audit records when audit trail storage is full or nearly full by restricting auditable events.
O.AUDIT_PROTECTION	The TOE will protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.
O.CERTIFICATES	The TSF must ensure that certificates, certificate revocation lists, and certificate status information are valid.
O.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.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.INTEGRITY_PROTECTION	The TOE will provide appropriate integrity protection for TSF data and software and any user data stored by the TOE.
O.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.
O.PROTECTED_ COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities. The TOE will protect data assets when they are being transmitted

	to and from the TOE, including through intervening untrusted components.
O.RECOVERY	The TOE will have the capability to store and recover to a previous state at the direction of the administrator (e.g., provide support for archival and recovery capabilities).
O.RESIDUAL_INFORMATION_ CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE will provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data. The TOE will record in audit records: date and time of action and the entity responsible for the action.
O.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 ensure that administrative responsibilities are separated across different roles in order to mitigate the impact of improper administrative activities or unauthorized administrative access.
O.TSF_SELF_TEST	The TOE will provide integrity protection to detect modifications to firmware, software, and archived data.
O.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 from a trusted source.

4.2 Security Objectives for the Operational Environment

TABLE 8.SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIORNMENT

Security Objectives for the Operational Environment	Description
OE.AUDIT_GENERATION	The Operational Environment provides a mechanism for the generation of portions of the audit data.

OE.CERT_REPOSITORY	The Operational Environment provides a certificate repository for storage of certificates (and optionally CRLs) issued by the TSF.
OE.CERT_REPOSITORY_ SEARCH	The Operational Environment provides the ability to search a certificate repository for specific certificate fields in certificates issued by the TSF and return the certificate and an identifier for the certificate that can be used to search the audit trail for events related to that certificate.
OE.AUDIT_RETENTION	The Operational Environment provides mechanisms for retention of audit records for both normal and extended retention periods.
OE.AUDIT_REVIEW	The Operational Environment provides a mechanism for the review of specified audit data.
OE.AUDIT_STORAGE	The Operational Environment provides a mechanism for the storage of specified audit data.
OE.CRYPTOGRAPHY	The Operational Environment provides cryptographic services that can be invoked by the TSF in order to perform security functionality.
OE.KEY_ARCHIVAL	The Operational Environment provides the ability to use split knowledge procedures to enforce two party control to export keys necessary to resume CA functionality if the TSF should fail.
OE.NO_GENERAL_PURPOSE	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.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.PUBLIC_KEY_PROTECTION	The Operational Environment provides protection for specified public keys associated with CA functions.
OE.SESSION_PROTECTION_ LOCAL	The Operational Environment provides the ability to lock or terminate local administrative sessions.
OE.SESSION_PROTECTION_ REMOTE	The Operational Environment provides the ability to lock or terminate remote administrative sessions.
OE.TOE_ADMINISTRATION	The Operational Environment provides specified management capabilities required for the overall operation of a Certificate

	Authority, and the ability to restrict access to a subset of the capabilities as specified in the ST.
OE.TRUSTED_ADMIN	The administrator of the TOE is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.
OE.TRUSTED_PLATFORM	The operating system on which the TOE has been installed is securely configured, regularly patched, and not subject to unauthorized access.

4.3 Security Objectives Rationale

The security objectives rationale is copied from the Protection Profile for Certification Authorities.

TABLE 9. SECURITY OBJECTIVES RATIONALE

SPD Element	Objective	Requirements
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.	OE.NO_GENERAL_PURPOSE 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.	N/A
A.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.	OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.	N/A
A.TRUSTED_ADMIN TOE Administrators are assumed to follow and apply all administrator guidance in a trusted manner.	OE.TRUSTED_ADMIN The administrator of the TOE is not careless, willfully negligent or hostile, and administers the software within compliance of the applied enterprise security policy.	N/A
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.	O.AUDIT_LOSS_RESPONSE The TOE will respond to possible loss of audit records when audit trail storage is full or nearly full by restricting auditable events.	FAU_ADP_EXT.1, FAU_STG.4
	O.AUDIT_PROTECTION The TOE will protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.	FAU_ADP_EXT.1
	O.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.	FIA_UAU_EXT.1, FIA_UIA_EXT.1, FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3),

	The TOE will ensure that administrative responsibilities are separated across different roles in order to mitigate the impact of improper administrative activities or unauthorized administrative access.	FMT_MOF.1(4), FMT_MOF.1(5), FMT_MTD.1, FMT_SMF.1, FMT_SMR.2, FTA_SSL.4
	OE.AUDIT_GENERATION The Operational Environment provides a mechanism for the generation of portions of the audit data.	N/A
	OE.AUDIT_STORAGE The Operational Environment provides a mechanism for the storage of specified audit data.	N/A
	OE.AUDIT_REVIEW The Operational Environment provides a mechanism for the review of specified audit data.	N/A
	OE.AUDIT_RETENTION The Operational Environment provides mechanisms for retention of audit records for both normal and extended retention periods.	N/A
	OE.SESSION_PROTECTION_LOCAL The Operational Environment provides the ability to lock or terminate local administrative sessions.	N/A
	OE.SESSION_PROTECTION_REMOTE The Operational Environment provides the ability to lock or terminate remote administrative sessions.	N/A
	OE.TOE_ADMINISTRATION The Operational Environment provides specified management capabilities required for the overall operation of a Certificate Authority, and the ability to restrict access to a subset of the capabilities as specified in the ST	N/A
	OE.TRUSTED_PLATFORM The operating system on which the TOE has been installed is securely configured, regularly patched, and not subject to unauthorized access.	N/A
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.	O.CERTIFICATES he TSF must ensure that certificates, ertificate revocation lists, and certificate status nformation are valid.	FDP_CER_EXT.1, FDP_CER_EXT.2, FDP_CER_EXT.3, FDP_CRL_EXT.1, FDP_CSI_EXT.1, FDP_OCSPG_EXT.1, FIA_ESTS_EXT.1, FIA_X509_EXT.1, FIA_X509_EXT.2,
	O.CONFIGURATION_MANAGEMENT The TOE will conduct configuration management to assure identification of system connectivity (software, hardware,	FDP_CER_EXT.1, FDP_CRL_EXT.1, FDP_OCSPG_EXT.1, FMT_MOF.1(1),

	and firmware), and components (software, hardware, and firmware), auditing of configuration data, and controlling changes to configuration items.	FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MTD.1,
	O.INTEGRITY_PROTECTION The TOE will provide appropriate integrity protection for TSF data and software and any user data stored by the TOE.	FCS_CDP_EXT.1,
	O.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.	FCO_NRO_EXT.2, FCO_NRR_EXT.2
	OE.PUBLIC_KEY_PROTECTION The Operational Environment provides protection for specified public keys associated with CA functions	N/A
	OE.TOE_ADMINISTRATION The Operational Environment provides specified management capabilities required for the overall operation of a Certificate Authority, and the ability to restrict access to a subset of the capabilities as specified in the ST	N/A
T.UNAUTHORIZED_ACCESS A malicious user, process, or external IT entity intentionally circumvents TOE security mechanisms.	O.PROTECTED_COMMUNICATIONS The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities. The TOE will protect data assets when they are being transmitted to and from the TOE, including through intervening untrusted components.	FCS_CDP_EXT.1, FCS_CKM.1, FCS_CKM.2, FCS_CKM_EXT.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP.1(5), FCS_HTTPS_EXT.1, FCS_RBG_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2, FPT_KST_EXT.1, FPT_KST_EXT.2, FPT_SKP_EXT.1, FTP_TRP.1
	O.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 ensure that administrative responsibilities are separated across different roles in order to mitigate the impact of improper administrative activities or unauthorized administrative access.	FIA_UAU_EXT.1, FIA_UIA_EXT.1, FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MTD.1, FMT_SMF.1, FMT_SMR.2, FTA_SSL.4
	OE.CRYPTOGRAPHY The Operational Environment provides cryptographic services that can be invoked by the TSF in order to perform security functionality.	N/A
	OE.KEY_ARCHIVAL	N/A

	The Operational Environment provides the ability to use split knowledge procedures to enforce two-party control to export keys necessary to resume CA functionality if the TSF should fail.	
	OE.SESSION_PROTECTION_LOCAL The Operational Environment provides the ability to lock or terminate local administrative sessions.	N/A
	OE.SESSION_PROTECTION_REMOTE The Operational Environment provides the ability to lock or terminate remote administrative sessions.	N/A
	OE.TOE_ADMINISTRATION The Operational Environment provides specified management capabilities required for the overall operation of a Certificate Authority, and the ability to restrict access to a subset of the capabilities as specified in the ST	N/A
T.UNAUTHORIZED_UPDATE A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.	O.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 from a trusted source.	FCS_CDP_EXT.1, FCS_COP.1(2), FIA_X509_EXT.2, FPT_TUD_EXT.1 FPT_TUD_EXT.2
T.UNDETECTED_ACTIONS Remote users or external IT entities may take actions that adversely affect the security of the TOE.	O.AUDIT_LOSS_RESPONSE The TOE will respond to possible loss of audit records when audit trail storage is full or nearly full by restricting auditable events.	FAU_ADP_EXT.1, FAU_STG.4
	O.AUDIT_PROTECTION The TOE will protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.	FAU_ADP_EXT.1
	O.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.	FAU_ADP_EXT.1, FAU_GEN.1, FAU_GEN.2 FAU_SAR.1 FAU_SAR.3 FAU_GCR_EXT.1, FAU_STG_EXT.1, FIA_UIA_EXT.1, FPT_STM.1
	OE.AUDIT_GENERATION The Operational Environment provides a mechanism for the generation of portions of the audit data.	N/A
	OE.AUDIT_STORAGE The Operational Environment provides a mechanism for the storage of specified audit data.	N/A
	OE.AUDIT_REVIEW The Operational Environment provides a mechanism for the review of specified audit data.	N/A
	OE.AUDIT_RETENTION	N/A

	The Operational Environment provides mechanisms for retention of audit records for both normal and extended retention periods.	
	OE.CERT_REPOSITORY The Operational Environment provides a certificate repository for storage of certificates (and optionally CRLs) issued by the TSF.	N/A
	OE.CERT_REPOSITORY_SEARCH The Operational Environment provides the ability to search a certificate repository for specific certificate fields in certificates issued by the TSF and return the certificate and an identifier for the certificate that can be used to search the audit trail for events related to that certificate.	N/A
T.USER_DATA_REUSE A malicious user, process, or external IT entity may gain access to user data that is not cleared when resources are reallocated.	O.RESIDUAL_INFORMATION_CLEARING The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.	FDP_RIP.1
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.	O.PROTECTED_COMMUNICATIONS The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities. The TOE will protect data assets when they are being transmitted to and from the TOE, including through intervening untrusted components.	FCS_CDP_EXT.1, FCS_CKM.1, FCS_CKM.2, FCS_CKM_EXT.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP.1(5), FCS_HTTPS_EXT.1, FCS_RBG_EXT.1, FCS_STG_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2, FPT_KST_EXT.1, FPT_KST_EXT.2, FPT_SKP_EXT.1, FTP_TRP.1
	O.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 from a trusted source.	FCS_CDP_EXT.1, FCS_COP.1(2), FIA_X509_EXT.2, FPT_TUD_EXT.1 FPT_TUD_EXT.2
	OE.CRYPTOGRAPHY The Operational Environment provides cryptographic services that can be invoked by the TSF in order to perform security functionality.	
	OE.KEY_ARCHIVAL The Operational Environment provides the ability to use split knowledge procedures to enforce two-party control to export keys necessary to resume CA functionality if the TSF should fail.	
P.ACCESS_BANNER	O.DISPLAY_BANNER	FTA_TAB.1

The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.	The TOE will display an advisory warning regarding use of the TOE.	

5 Extended Component Definition

Extended components have been defined in the Protection Profile for Certification Authorities.

Extended security requirements are explicitly identified in [8] and thoroughly described in the PP.

TABLE 10.EXTENDED COMPONENTS DEFINITION

Extended Components
FAU_ADP_EXT.1 Audit Dependencies
FAU_GCR_EXT.1 Generation of Certificate Repository
FAU_STG_EXT.1 External Audit Trail Storage
FCO_NRO_EXT.2 Certificate-Based Proof of Origin
FCO_NRR_EXT.2 Certificate-Based Proof of Receipt
FCS_CDP_EXT.1 Cryptographic Dependencies
FCS_CKM_EXT.4 Cryptographic Key Destruction
FCS_HTTPS_EXT.1 HTTPS Protocol
FCS_RBG_EXT.1 Cryptographic Random Bit Generation
FCS_STG_EXT.1 Cryptographic Key Storage
FCS_TLSS_EXT.1 TLS Server Protocol
FCS_TLSS_EXT.2 TLS Server Protocol with Mutual Authentication
FDP_CER_EXT.1 Certificate Profiles
FDP_CER_EXT.2 Certificate Request Matching
FDP_CER_EXT.3 Certificate Issuance Approval
FDP_CRL_EXT.1 Certificate Revocation List Validation
FDP_CSI_EXT.1 Certificate Status Information
FDP_OCSPG_EXT.1 OCSP Basic Response Generation
FIA_ESTS_EXT.1 Enrollment over Secure Transport (EST) Server
FIA_X509_EXT.1 Certificate Validation
FIA_X509_EXT.2 Certificate-Based Authentication
FIA_X509_EXT.3 X509 Certificate Request
FIA_UAU_EXT.1 Authentication Mechanism
FIA_UIA_EXT.1 User Identification and Authentication
FPT_KST_EXT.1 No Plaintext Key Export
FPT_KST_EXT.2 TSF Key Protection
FPT_SKP_EXT.1 Protection of Keys
FPT_TUD_EXT.1 Trusted Update

6 Security Requirements

The security requirements are based on the Protection Profile, ref. □, and include the following parts:

- Security functional requirements (SFRs);
- Security assurance requirements (SARs); and
- Security requirements rationale.

6.1 Security Functional Requirements

The following conventions have been applied in this document. Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.

- Iteration: Allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a number in parenthesis placed at the end of the component. For example, FCS_COP.1 (1) and FCS_COP.1(2) indicate that the ST includes two iterations of the FCS_COP.1 requirement, “1” and “2”.

- Assignment: Allows the specification of an identified parameter. Assignments performed in the Protection Profile are copied into this ST in plain text surrounded by brackets (e.g., [assignment]). Assignments performed in this ST are indicated using ***bold italics*** and are surrounded by brackets (e.g., [***assignment***]).

- Selection: Allows the specification of one or more elements from a list. Selections performed in the Protection Profile are copied into this ST in plain text surrounded by brackets (e.g., [selection]). Selections performed in this ST are indicated using **bold** and are surrounded by brackets (e.g., [**selection**]).

- Refinements performed in the Protection Profile are copied into this ST in plain text and are identified with "**Refinement**:" right after the short name. Additions to the CC text are specified in ***italicized bold and underlined text***.

6.1.1 Security Audit

Describes the TOE SFRs related to security audit.

TABLE 11. SECURITY AUDIT

FAU_ADP_EXT.1	**Audit Dependencies**
FAU_ADP_EXT.1.1	The TSF shall implement audit functionality and [**no additional audit functionality**] in order to perform audit operations on the following audit data: [***auditable events that require persistent storage***].
FAU_GCR_EXT.1	**Generation of Certificate Repository**
FAU_GCR_EXT.1.1	The TSF shall [**invoke the Operational Environment to store]** certificates and [**CRLs**] issued by the TSF.
FAU_GEN.1	**Audit data generation**
FAU_GEN.1.1	Refinement: The TSF shall generate and [**invoke the Operational Environment to generate**] an audit record of the following auditable events: a) Start-up of the TSF audit functions; b) All auditable events for the [**not specified**] level of audit; and c) All administrative actions invoked through the TSF interface; d) [Specifically defined auditable events listed in Table 12, Table 13 and Table 14].
FAU_GEN.1.2	Refinement: The TSF shall [**include**] within each audit record at least the following information: a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [information specified in column three of Table 12, Table 13 and Table 14].
FAU_GEN.2	**User identity association**
FAU_GEN.2.1	Refinement: 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	Refinement: The TSF shall provide the audit records in a manner suitable for the Auditor 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 [***certificate serial number***] associated with the event.
FAU_STG.4	**Prevention of audit data loss**
FAU_STG.4.1	Refinement: The TSF shall prevent audited events, except those taken by the Auditor and [**generate an error message according to FPT_FLS.1**] if the audit trail cannot be written to.

FAU_STG_EXT.1	External Audit Trail Storage
FAU_STG_EXT.1.1	The TSF shall maintain availability and integrity of audit data by storing it [**locally on the TOE platform**].

TABLE 12. AUDIT EVENTS AND RESPECTIVE DATA

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/ Extended	Responsible TSF or OE Component
FAU_ADP_EXT.1	None.	None.	N/A	None
FAU_GCR_EXT.1	None.	None.	N/A	None
FAU_GEN.1	None.	None.	N/A	None
FAU_GEN.2	None.	None.	N/A	None
FAU_STG.4	None.	None.	N/A	None
FCO_NRO_EXT.2	None.	None.	N/A	None
FCS_CDP_EXT.1	None.	None.	N/A	None
FCS_STG_EXT.1	None.	None.	N/A	None
FDP_CER_EXT.1	Certificate generation.	Success: [*Certificate object identifier*]. Note: The certificate object identifier is a hyperlink to the certificate value.	Extended	TSF
FDP_CER_EXT.2	Linking of certificate to certificate request	Success: [*Certificate object identifier*], [*Certificate request*]. Failure: Reason for failure, [*Certificate request*].	Extended	TSF
FDP_CER_EXT.3	Failed certificate approvals.	Reason for failure. [*Certificate request*].	Normal	TSF
FDP_CSI_EXT.1	None.	None.	N/A	None
FDP_RIP.1	None.	None.	N/A	None

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/ Extended	Responsible TSF or OE Component
FIA_X509_EXT.1	Failed certificate validations.	None.	Normal	TSF
FIA_X509_EXT.2	Failed authentications.	None.	Normal	TSF
FIA_UAU_EXT.1	All uses of the authentication mechanism used for access to TOE related functions.	Origin of the attempt (e.g., IP address).	Normal	TSF
FIA_UIA_EXT.1	All use of the identification and authentication mechanism used for TOE related roles.	Provided user identity. Origin of the attempt (e.g., IP address).	Normal	TSF
FMT_MOF.1(1)	None.	None.	N/A	None
FMT_MOF.1(2)	None.	None.	N/A	None
FMT_MOF.1(3)	None.	None.	N/A	None
FMT_MOF.1(4)	None.	None.	N/A	None
FMT_MOF.1(5)	None.	None.	N/A	None
FMT_MTD.1	None.	None.	N/A	None
FMT_SMF.1	None.	None.	N/A	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.	Extended	TSF
FPT_FLS.1	Invocation of failures under this requirement.	Indication that the TSF has failed with the type of failure that occurred.	Normal	TSF
FPT_KST_EXT.1	None.	None.	N/A	None
FPT_KST_EXT.2	All unauthorized attempts to use TOE secret and private keys.	Identifier of user or process that attempted access.	Normal	OE (the TOE secret and private keys are stored in the HSM)

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/ Extended	Responsible TSF or OE Component
FPT_RCV.1	The fact that a failure or service discontinuity occurred. Resumption of the regular operation.	TSF failure types that are available on recovery.	Extended	TSF
FPT_SKP_EXT.1	None.	None.	N/A	None
FPT_STM.1	Changes to the time.	The old and new values for the time.	Normal	OE (The TOE uses the timestamps from the underlying platform)
FPT_TUD_EXT.1	Initiation of update.	Version number.	Extended	OE (The TOE is distributed as a container)
FTA_SSL.4	The termination of an interactive session.	None.	Normal	TSF
FTA_TAB.1	None.	None.	N/A	None
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.	Identification of the claimed user identity.	Normal	TSF

TABLE 13. AUDITABLE EVENTS AND RESPECTIVE DATA FOR OPTIONAL REQUIREMENTS

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
FCS_COP.1(5)	None.	None.	N/A	N/A – Not claimed in this ST
FPT_SKY_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
FPT_TST_EXT.1	Execution of this set of TSF integrity tests. Detected integrity violations.	For integrity violations, the identity of the object that caused the integrity violation.	Normal	N/A – Not claimed in this ST
FPT_TST_EXT.2	Execution of this set of TSF integrity tests. Detected integrity violations.	For integrity violations, the identity of the object that caused the integrity violation.	Normal	N/A – Not claimed in this ST
FDP_CER_EXT.4	Certificate generation.	Name/identifier of certificate, value of certificate generated.	Extended	N/A – Not claimed in this ST
FDP_SDP_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST
FDP_STG_EXT.1	Changes to the trusted public keys and certificates relevant to TOE functions, including additions and deletions.	The public key and all context information associated with the key.	Normal	N/A – Not claimed in this ST
FPT_NPE_EXT.1	All changes to NPE rule sets and NPE associations.	The changes made to the NPE rule sets and associations.	Extended	N/A – Not claimed in this ST
FTA_SSL.3	The termination of a remote session by the session termination mechanism.	None.	Normal	N/A – Not claimed in this ST
FTA_SSL_EXT.1	Any attempts at unlocking or termination of an interactive session.	None.	Normal	N/A – Not claimed in this ST

TABLE 14. AUDITABLE EVENTS AND RESPECTIVE DATA FOR SELECTABLE REQUIREMENTS

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
FAU_SCR_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST
FAU_SAR.1	None.	None.	N/A	None
FAU_SAR.3	None.	None.	N/A	None
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	None.	Normal	N/A – Not claimed in this ST
FAU_STG.1(1)	None	None.	N/A	N/A – Not claimed in this ST
FAU_STG.1(2)	None	None	N/A	N/A – Not claimed in this ST
FAU_STG_EXT.1	None.	None.	N/A	None
FAU_STG_EXT.2	None,	None.	N/A	N/A – Not claimed in this ST
FCO_NRR_EXT.2	None.	None.	N/A	None
FCS_CKM.1	All occurrences of non- ephemeral and [no other] key generation for TOE related functions.	Success: public key generated	Normal	OE (non-ephemeral keys are generated in the HSM)
FCS_CKM.2	All occurrences of nonephemeral and [*no other*] key establishment for TOE related functions.	Success: key established	Normal	OE (non-ephemeral keys are generated in the HSM)
FCS_CKM_EXT.1(1)	None.	None.	N/A	N/A – Not claimed in this ST
FCS_CKM_EXT.1(2)	None.	None.	N/A	N/A – Not claimed in this ST

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
FCS_CKM_EXT.1(3)	None	None.	N/A	N/A – Not claimed in this ST
FCS_CKM_EXT.1(4)	None.	None.	N/A	N/A – Not claimed in this ST
FCS_CKM_EXT.4	Failure of the key destruction process for TOE related keys.	Identity of object or entity being cleared.	Normal	OE (non-ephemeral keys are generated in the HSM and the TLS secrets are handled by the JVM)
FCS_CKM_EXT.5	Detection of integrity violation for stored TSF data.	None.	Normal	N/A – Not claimed in this ST
FCS_CKM_EXT.6	All key archival actions.	None.	Extended	N/A – Not claimed in this ST
FCS_CKM_EXT.7	None.	None.	N/A	N/A – Not claimed in this ST
FCS_CKM_EXT.8	None.	None.	N/A	N/A – Not claimed in this ST
FCS_COP.1(1)	None.	None.	N/A	None
FCS_COP.1(2)	All occurrences of signature generation using a CA signing key. Failure in signature generation	Name/identifier of object being signed Identifier of key used for signing. None.	Extended Normal	TSF and OE. The signature operations are performed in the HSM. The TOE is able to record when the operation cannot be performed.
FCS_COP.1(3)	None.	None.	N/A	None
FCS_COP.1(4)	None.	None.	N/A	None

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session. Establishment/ Termination of a HTTPS	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and	Normal	TSF
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA. Establishment/ Termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.	Normal	N/A – Not claimed in this ST
FCS_RBG_EXT.1	None.	None.	N/A	None
FCS_TLSS_EXT.1	Failure to establish a TLS session. Establishment/ Termination of a TLS session.	Reason for failure. None.	Normal	TSF
FCS_TLSS_EXT.2	Failure to establish a TLS session. Establishment/ Termination of a TLS session.	Reason for failure. None.	Normal	TSF
FDP_CRL_EXT.1	Failure to generate CRL.	None.	Normal	TSF
FDP_ITT.1	None.	None.	N/A	N/A – Not claimed in this ST
FDP_OCSPG_EXT.1	Failure to generate certificate status information.	None.	Extended	TSF
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts. The action taken. The	None.	Normal	N/A – Not claimed in this ST

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
	reenablement of disabled nonadministrative accounts.			
FIA_CMCS_EXT.1	CMC requests (generated or received) containing certificate requests or revocation requests. CMC responses issued.	Identifiers for all entities authenticating the request, including the entity providing client authentication for the CMC transport (if any). The submitted request. Any signed response	Extended	N/A – Not claimed in this ST
FIA_CMCC_EXT.1	CMC requests (generated or received) containing certificate requests or revocation requests. CMC responses issued.	Identifiers for all entities authenticating the request, including the entity providing client authentication for the CMC transport (if any). The submitted request. Any signed response.	Extended	N/A – Not claimed in this ST
FIA_ESTC_EXT.1	EST requests (generated or received) containing certificate requests or revocation requests.	Identifiers for all entities authenticating the request, including the entity providing client authentication for the EST transport (if any). The submitted request. Any signed response.	Extended	N/A – Not claimed in this ST

Requirement	Auditable Events	Additional Audit Record Contents	Retention Normal/Extended	Responsible TSF or OE Component
	EST responses issued.			
FIA_ESTS_EXT.1	EST requests (generated or received) containing certificate requests or revocation requests. EST responses issued.	Identifiers for all entities authenticating the request, including the entity providing client authentication for the EST transport (if any). The submitted request. Any signed response.	Extended	TSF
FIA_PMG_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST
FIA_PSK_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST
FIA_UAU.7	None.	None.	N/A	N/A – Not claimed in this ST
FPT_APW_EXT.1	None.	None.	N/A	N/A – Not claimed in this ST
FPT_ITT.1	None.	None.	N/A	N/A – Not claimed in this ST
FPT_SKY_EXT.2	Access control violations for users involved in key share establishment or control.	None.	Extended	N/A – Not claimed in this ST
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.	Normal	N/A – Not claimed in this ST

6.1.2 Communication (FCO)

TABLE 15. COMMUNICATION

FCO_NRO_EXT.2	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 a 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), OCSP (RFC 6960)**] and FCS_COP.1(2).
FCO_NRO_EXT.2.3	The TSF shall require and verify proof of origin for certificate requests it receives [**EST using mechanisms in accordance with FIA_ESTS_EXT.1**].
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 in EST in accordance with FIA_ESTS_EXT.1**].
FCO_NRO_EXT.2.5	The TSF shall [require and verify proof of origin for revocation requests it receives via [**role authorization].**
FCO_NRR_EXT.2	**Certificate-Based Proof of Receipt**
FCO_NRR_EXT.2.1	The TSF shall provide proof of receipt for [**EST**] by providing signed responses using mechanisms in accordance with [**FIA_ESTS_EXT.1**].

6.1.3 Cryptographic Support (FCS)

TABLE 16. CRYPTOGRAPHIC

FCS_CDP_EXT.1(TSF)	Cryptographic Dependencies
FCS_CDP_EXT.1.1	The TSF shall [**implement cryptographic functionality**] in order to perform [• **FCS_CKM.1(TSF)** • **FCS_CKM_EXT.4** • **FCS_CKM.2** • **FCS_COP.1(1)** • **FCS_COP.1(2)(TSF)** • **FCS_COP.1(3/TSF)** • **FCS_COP.1(4)** • **FCS_RBG_EXT.1**] cryptographic operations.
FCS_CDP_EXT.1(OE)	**Cryptographic Dependencies**
FCS_CDP_EXT.1.1	The TSF shall [**invoke interfaces provided by the Operational Environment**] in order to perform [• **FCS_CKM.1(OE)** • **FCS_COP.1(2)(OE)** • **FCS_COP.1(3/OE)**] cryptographic operations.

FCS_CKM.1(TSF)	Cryptographic Key Generation
FCS_CKM.1.1	Refinement: The TSF shall **[generate]** asymmetric cryptographic keys in accordance with the specified key generation algorithm: **[** • **ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-5, Digital Signature Standard (DSS), Appendix B.4]** and specified cryptographic key sizes **[*256-, 384- and 521-bit ECC*]**.
FCS_CKM.1(OE)	**Cryptographic Key Generation**
FCS_CKM.1.1	Refinement: The TSF shall **[invoke interfaces provided by the Operational Environment to generate]** asymmetric cryptographic keys in accordance with the specified key generation algorithm: **[** • **RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following FIPS PUB 186-4, "Digital Signature Standard (DSS), Appendix B.3";** • **ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, Digital Signature Standard (DSS), Appendix B.4]** and specified cryptographic key sizes **[*2048 bits, 3072 bits, 4096 bits and 256, 384, 521 bit ECC*]**.
FCS_CKM.2	**Cryptographic Key Establishment**
FCS_CKM.2.1	Refinement: The TSF shall **[perform]** key establishment in accordance with a specified cryptographic key establishment algorithm **[** • **Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair- Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"** **].**
FCS_CKM_EXT.4	**Cryptographic Key Destruction**
FCS_CKM_EXT.4.1	The TSF shall [**invoke interfaces provided by the Operational Environment to destroy**] all cryptographic keys and critical security parameters which are not permanently protected from export by hardware when no longer required, in accordance with the specified cryptographic key destruction method [• **for volatile memory, the destruction shall be executed by a [** ○ **destruction of reference to the key directly followed by a request for garbage collection],**
FCS_CKM_EXT.4.2	The TSF shall [**invoke interfaces provided by the Operational Environment to destroy**] all plaintext keying material cryptographic security parameters when no longer needed.
FCS_COP.1(1)	**Cryptographic Operation (AES Encryption/Decryption)**

FCS_COP.1.1(1)	**Refinement:** The TSF shall **[perform]** [encryption and decryption] in accordance with a specified cryptographic algorithm: **[** • **AES-CBC (as defined in NIST SP 800-38A) mode,** • **AES-GCM (as defined in NIST SP 800-38D) mode,** and cryptographic key size **[128-bit, 256-bit]**.
FCS_COP.1(2)(TSF)	**Cryptographic Operation (Cryptographic Signature)**
FCS_COP.1.1(2)(TSF)	Refinement: The TSF shall **[perform]** [cryptographic signature services] in accordance with the following specified cryptographic algorithms **[** • **RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of [2048 bits, 3072 bits, 4096 bits] that meets FIPS-PUB 186-4, "Digital Signature Standard",** • **Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256 bits or greater that meets 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")]**
FCS_COP.1(2)(OE)	**Cryptographic Operation (Cryptographic Signature)**
FCS_COP.1.1(2)(OE)	Refinement: The TSF shall **[invoke interfaces in the operational environment to perform]** [cryptographic signature services] in accordance with the following specified cryptographic algorithms **[** • **RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of [2048 bits, 3072 bits, 4096 bits] that meets FIPS-PUB 186-4, "Digital Signature Standard",** • **Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256 bits or greater that meets 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")]**
FCS_COP.1(3)(TSF)	**Cryptographic Operation (Cryptographic Hashing)**
FCS_COP.1.1(3)(TSF)	Refinement: 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(3)(OE)	**Cryptographic Operation (Cryptographic Hashing)**
FCS_COP.1.1(3)(OE)	Refinement: The TSF shall [**invoke interfaces in the operational environment to 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(4)	**Cryptographic Operation (Keyed-Hash Message Authentication)**
FCS_COP.1.1(4)	Refinement: The TSF shall [**perform**] [keyed hash message authentication] in accordance with a specified cryptographic algorithm HMAC-[**SHA-256, SHA-384, SHA-512**], key size [**256, 384, 512**] and message digest sizes [**256, 384,**

	512] bits that meet the following: [FIPS Pub 198-1,"The Keyed Hash Message Authentication Code"; FIPS Pub 180-4, "Secure Hash Standard"].
FCS_RBG_EXT.1	**Cryptographic Random Bit Generation**
FCS_RBG_EXT.1.1	The TSF shall [**perform**] all deterministic random bit generation (RBG) services in accordance with NIST Special Publication 800-90A using [**HMAC-DRBG(SHA-512)**].
FCS_RBG_EXT.1.2	The deterministic RBG shall be seeded by an entropy source that accumulates entropy from [**an Operational Environment-based noise source**] with a minimum of [**256 bits**] of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and authorization factors that it will generate.
FCS_HTTPS_EXT.1	**HTTPS Protocol**
FCS_HTTPS_EXT.1.1	The TSF shall implement the HTTPS protocol that complies with RFC 2818.
FCS_HTTPS_EXT.1.2	The TSF shall implement HTTPS using TLS.
FCS_STG_EXT.1	**Cryptographic Key Storage**
FCS_STG_EXT.1.1	Persistent private and secret keys shall be stored within the [**Operational Environment**] [**in a hardware cryptographic module**].
FCS_TLSS_EXT.1	**TLS Server Protocol**
FCS_TLSS_EXT.1.1	The TSF shall implement [**TLS 1.2 (RFC 5246)**] supporting the following ciphersuites: [• **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289]** and no other ciphersuite.
FCS_TLSS_EXT.1.2	The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [**TLS 1.1**].
FCS_TLSS_EXT.1.3	The TSF shall [**generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves]**.
FCS_TLSS_EXT.2	**TLS Server Protocol with Mutual Authentication**
FCS_TLSS_EXT.2.1	The TSF shall implement [**TLS 1.2 (RFC 5246)**] supporting the following ciphersuites: [

	• **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289]** and no other ciphersuite.
FCS_TLSS_EXT.2.2	The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [**TLS 1.1**].
FCS_TLSS_EXT.2.3	The TSF shall [**generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves]**.
FCS_TLSS_EXT.2.4	The TSF shall support mutual authentication of TLS clients using X.509 certificates.
FCS_TLSS_EXT.2.5	For communications configured to require TLS with mutual authentication, the shall not establish a trusted channel if the client certificate is invalid.
FCS_TLSS_EXT.2.6	The TSF shall respond with a fatal TLS error if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate presented for client authentication does not match the expected identifier for the client.

6.1.4 User Data Protection (FDP)

TABLE 17. USER DATA PROTECTION

FDP_CER_EXT.1	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 IETF RFC 5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”, while ensuring that the following conditions are met: 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: a. subjectKeyIdentifier b. authorityKeyIdentifier c. basicConstraints d. keyUsage e. extendedKeyUsage f. 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 issuer'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 [**serialNumber**] fields, where the random values are generated in accordance with FCS_RBG_EXT.1.
FDP_CER_EXT.2	**Certificate Request Matching**
FDP_CER_EXT.2.1	The TSF shall establish a linkage from certificate requests to issued certificates.
FDP_CER_EXT.3	**Certificate Issuance Approval**
FDP_CER_EXT.3.1	The TSF shall support the approval of certificates by [**CA Operations Staff**] issued according to a configured certificate profile.
FDP_CRL_EXT.1	**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: a) If the version field is present, then it shall contain a 1. b) If the CRL contains any critical extensions, then the version field shall be present and contain the integer 1. c) 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. d) The signature and signatureAlgorithm fields shall contain the OID for a digital signature algorithm in accordance with FCS_COP.1(2).

	e) The thisUpdate field shall indicate the issue date of the CRL. f) The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.
FDP_CSI_EXT.1	**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, the OCSP standard as defined by [RFC 6960]**].
FDP_CSI_EXT.1.2	The TSF shall support the approval of changes to the status of a certificate by [**CA Operations Staff**].
FDP_OCSPG_EXT.1	**OCSP Basic Response Generation**
FDP_OCSPG_EXT.1.1	The TSF shall ensure that all mandatory fields in the OCSP response contain values in accordance with the standards specified in FDP_CSI_EXT.1. At a minimum, the following items shall be enforced: a) The version field shall indicate a current version. b) The signatureAlgorithm field shall contain the object identifier (OID) for a digital signature algorithm in accordance with FCS_COP.1(2). c) The thisUpdate field shall indicate the time at which the status being indicated is known to be correct. d) The producedAt field shall indicate the time at which the OCSP responder signed the response. e) The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.
FDP_RIP.1	**Subset Residual Information Protection**
FDP_RIP.1.1	Refinement**:** The TSF and [**Operational Environment**] shall ensure that any previous information content of a resource is made unavailable upon the [**deallocation of the resource from**] the following objects: [***TLS data buffer and ephemeral keys***].

6.1.5 Identification and Authentication (FIA)

TABLE 18. IDENTIFICATION AND AUTHENTICATION

FIA_ESTS_EXT.1	**Enrollment over Secure Transport (EST) Server**
FIA_ESTS_EXT.1.1	The TSF shall use the Enrollment over Secure Transport (EST) protocol as specified in RFC 7030 to receive, process, and respond to certificate simple enrollment requests from authorized clients.
FIA_ESTS_EXT.1.2	The TSF shall authenticate EST clients for re-enrollment via TLS certificate-based mutual authentication in accordance with RFC 7030 Section 3.3.2 and FCS_TLSS_EXT.2.

FIA_ESTS_EXT.1.3	The TSF shall authenticate EST clients for initial enrollment and for supplemental authentication via [**TLS certificate-based mutual authentication in accordance with RFC 7030 section 3.3.2 and FCS_TLSS_EXT.2**].
FIA_ESTS_EXT.1.4	The TSF shall authorize EST clients based on [**role authorization**].
FIA_X509_EXT.1	**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 flag is set to TRUE for all CA certificates, and that any path constraints are met. • The TSF shall validate the revocation status of the certificate using [**OCSP as specified in RFC 6960**]. • The TSF shall validate the extendedKeyUsage (EKU) field according to the following rules: o 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) in the extendedKeyUsage field. o Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the EKU field. o 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 EKU field. o Delegated OCSP signer's certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the EKU field. o Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the EKU 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	**Certificate-Based Authentication**
FIA_X509_EXT.2.1	The TSF shall [**use**] X.509v3 certificates as defined by RFC 5280 to support [**TLS, HTTPS**], and [**[privileged user's access, access for EST**]].
FIA_X509_EXT.2.2	When the TSF cannot determine the current revocation status 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.
FIA_X509_EXT.3	**X509 Certificate Request**

FIA_X509_EXT.3.1	The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key, CA's distinguished name, [**no other information**].
FIA_X509_EXT.3.2	The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.
FIA_UAU_EXT.1	**Authentication Mechanism (UI)**
FIA_UAU_EXT.1.1	The TSF shall [**provide**] a [**[certificate-based authentication mechanism]**] to perform privileged user authentication.
FIA_UIA_EXT.1	**User Identification and Authentication**
FIA_UIA_EXT.1.1	The TSF shall allow the following actions prior to requiring a non-TOE entity to initiate the identification and authentication process: • Display the warning banner in accordance with FTA_TAB.1; • Obtain certificate status information; • [[**Public information retrieval requests**]].
FIA_UIA_EXT.1.2	The TSF shall require each user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that user, including subscriber certificate renewal, subscriber revocation requests, privileged user access, [**no other actions**].
FIA_UIA_EXT.1.3	For subscriber actions, the TSF shall verify that the DN of the certificate presented by the subscriber for authentication matches that of the certificate being affected by the subscriber's actions.

6.1.6 Security Management (FMT)

TABLE 19. IDENTIFICATION AND AUTHENTICATION

FMT_MOF.1(1)	**Management of Security Functions Behavior (Administrator Functions)**
FMT_MOF.1(1)	Refinement: The [**TSF, Operational Environment**] shall restrict the ability to: 1) manage the TOE locally (OE) and remotely (TSF) 2) configure the audit mechanism (OE); 3) configure and manage certificate profiles (TSF); 4) modify revocation configuration (TSF); 5) perform updates to the TOE (OE); 6) perform on-demand integrity tests (OE); 7) import and remove X.509v3 certificates into/from the Trust Anchor Database (TSF); [**8) configure certificate revocation list function (TSF);** **9) configure OCSP function (TSF);**

	10) disable deprecated algorithms (TSF); **11) accept certificates whose validity cannot be determined (TSF);** **12) export PKCS#10 certificate request;** **13) import CA certificate;** [**14) request activation and use of component private keys maintained within the cryptographic module;** **15) request the generation of persistent Component keys (TSF);**]] to Administrators.
FMT_MOF.1(2)	**Management of Security Functions Behavior (CA/RA Functions)**
FMT_MOF.1.1(2)	Refinement: The [**TSF**] shall restrict the ability to 1) approve and execute the issuance of certificates; 2) configure subscriber self-service request constraints; [3) configure automated certificate approval management;] to [**CA Operations Staff**].
FMT_MOF.1(3)	**Management of Security Functions Behavior (CA Operations Functions)**
FMT_MOF.1.1(3)	Refinement: The [**TSF**] shall restrict the ability to 1) approve certificate revocation; [2) no other function;] to [**CA Operations Staff**].
FMT_MOF.1(4)	**Management of Security Functions Behavior (Admin/Officer Functions)**
FMT_MOF.1.1(4)	Refinement: The [**Operational Environment**] shall restrict the ability to 1) perform destruction of sensitive data when no longer needed (OE); [2) no other function;] to [**Administrators**].
FMT_MOF.1(5)	**Management of Security Functions Behavior (Auditor Functions)**
FMT_MOF.1.1(5)	Refinement: The [**TSF, Operational Environment**] shall restrict the ability to 1) Delete entries from the audit trail (OE); [

	2) Search the audit trail;] to auditors.
FMT_MTD.1	**Management of TSF Data**
FMT_MTD.1.1	The TSF shall restrict the ability to manage the TSF data to privileged users.
FMT_SMF.1	**Specification of Management Functions**
FMT_SMF.1.1	Refinement: The [**TSF, Operational Environment**] shall be capable of performing the following management functions: [1) Ability to manage the TOE locally and remotely; 2) Ability to perform updates to the TOE (OE); 3) Ability to perform archival and recovery (OE); 4) Ability to manage the audit mechanism (OE); 5) Ability to configure and manage certificate profiles (TSF); 6) Ability to approve and execute the issuance of certificates (TSF); 7) Ability to approve certificate revocation (TSF); 8) Ability to modify revocation configuration (TSF); 9) Ability to configure subscriber self-service request constraints (TSF); 10) Ability to perform on-demand integrity tests (OE); 11) Ability to destroy sensitive user data when no longer needed (OE); 12) Ability to import and remove X.509v3 certificates into/from the Trust Anchor Database (TSF); **[** 13) **Ability to modify the CRL configuration (TSF);** 14) **Ability to modify the OCSP configuration (TSF);** **]** 15) **Ability to configure the cryptographic functionality (TSF);** 16) **Ability to disable deprecated algorithms (TSF);** 17) **Ability to accept certificates whose revocation status cannot be determined (TSF).**]].
FMT_SMR.2	**Restrictions on Security Roles**

FMT_SMR.2.1	Refinement: The TSF and [**no other component**] shall maintain the roles: [• Administrator, • Auditor, • CA Operations Staff, • **[RA Staff,** • **Authorized Organizational Representative]]** **Application note:** The corresponding TOE role names are: Administrator – Super Administrator Auditor – Auditor CA Operations Staff – CA Administrators RA Staff - RA Administrators Authorized Organizational Representative - Supervisor
FMT_SMR.2.2	Refinement: The TSF and [**no other component**] shall be able to associate users with roles.
FMT_SMR.2.3	Refinement: The TSF and [**no other component**] 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.

6.1.7 Protection of the TSF (FPT)

TABLE 20. PROTECTION OF THE TSF

FPT_FLS.1	**Failure with Preservation of Secure State**
FPT_FLS.1.1	The TSF shall preserve a secure state when the following types of failures occur: [[**PKCS#11 Cryptographic Module failure, Database inaccessible**]].
FPT_KST_EXT.1	**No Plaintext Key Export**
FPT_KST_EXT.1.1	The TSF and [**Operational Environment**] shall prevent the plaintext export of ***[all TOE secret and private keys and user secret and private keys***].
FPT_KST_EXT.2	**TSF Key Protection**
FPT_KST_EXT.2.1	The TSF and [**Operational Environment**] shall prevent unauthorized use of all TSF private and secret keys.
FPT_RCV.1	**Manual Trusted Recovery**
FPT_RCV.1.1	After [***PKCS#11 Cryptographic Module failure, Database becomes inaccessible***] the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.
FPT_SKP_EXT.1	**Protection of Keys**

FPT_SKP_EXT.1.1	The TSF shall [**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).
FPT_STM.1	**Reliable Time Stamps**
FPT_STM.1.1	Refinement: The TSF shall [**interface with the Operational Environment to provide**] reliable time stamps.
FPT_TUD_EXT.1	**Trusted Update**
FPT_TUD_EXT.1.1	The TSF shall [**interface with the Operational Environment to implement**] the ability to check for updates and patches to the TOE.
FPT_TUD_EXT.1.2	The TSF shall [**interface with the Operational Environment to implement**] the ability to provide Administrators the ability to initiate updates to TOE firmware/software.
FPT_TUD_EXT.1.3	The TSF shall [**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.1.4	The TSF shall [**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 [***inform the Administrator***].
FPT_TUD_EXT.1.5	The TSF is distributed as [**an application**], [**as an additional software package to the platform OS**].
FPT_TUD_EXT.2	**Integrity for Installation and Update**
FPT_TUD_EXT.2.1	The software package containing the TSF shall be distributed using [**a container image**].
FPT_TUD_EXT.2.2	The software package shall be packaged such that its removal results in the deletion of all traces of the application, with the exception of configuration settings, output files, and audit/log events.
FPT_TUD_EXT.2.3	The software installation package shall be digitally signed such that its platform can cryptographically verify them prior to installation.

6.1.8 TOE Access (FTA)

TABLE 21. TOE ACCESS

FTA_SSL.4	**User-Initiated Termination**
FTA_SSL.4.1	Refinement: The TSF shall [**implement**] the ability to allow privileged user-initiated termination of the privileged user's own interactive session
FTA_TAB.1	**Default TOE Access Banners**
FTA_TAB.1.1	Refinement: Before establishing a privileged user session the TSF shall display an Administrator-configured advisory notice and consent warning message regarding unauthorized use of the TOE.

6.1.9 Trusted Path/Channels (FTP)

TABLE 22. FTP TRUSTED PATHS/CHANNELS

FTP_TRP.1	Trusted Path
FTP_TRP.1.1	Refinement: The TSF shall use [**HTTPS, TLS**] to provide a trusted communication path between itself and remote subscribers and privileged users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification, disclosure.
FTP_TRP.1.2	Refinement: The TSF shall permit remote subscribers and privileged users to initiate communication via the trusted path.
FTP_TRP.1.3	The TSF shall require the use of the trusted path for [***initial subscriber and privileged user authentication and all remote administration actions***].

6.2 Security Assurance Requirements

6.2.1 Development (ADV)

TABLE 23. DEVELOPMENT

ADV_FSP.1	Basic Functional Specification
	Developer action elements:
ADV_FSP.1.1D	The developer shall provide a functional specification.
ADV_FSP.1.2D	The developer shall provide a tracing from the functional specification to the SFRs.
Developer Note:	*As indicated in the introduction to this section, the functional specification is comprised of the information contained in the AGD_OPE and AGD_PRE documentation, coupled with the information provided in the TSS of the ST. The assurance activities in the functional requirements point to evidence that should exist in the documentation and TSS section; since these are directly associated with the SFRs, the tracing in element ADV_FSP.1.2D is implicitly already done and no additional documentation is necessary.*
	Content and presentation elements:
ADV_FSP.1.1C	The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.2C	The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.3C	The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-noninterfering.
ADV_FSP.1.4C	The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

	Evaluator action elements:
ADV_FSP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.1.1E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

6.2.2 Guidance Documentation (ADV)

TABLE 24. GUIDANCE DOCUMENTATION ASSURANCE REQUIREMENTS

AGD_OPE.1	**Operational User Guidance**
	Developer action elements:
AGD_OPE.1.1D	The developer shall provide operational user guidance.
Developer Note:	*Rather than repeat information here, the developer should review the assurance activities for this component to ascertain the specifics of the guidance that the evaluator will be checking for. This will provide the necessary information for the preparation of acceptable guidance.*
	Content and presentation elements:
AGD_OPE.1.1C	The operational user guidance shall describe, for each privileged user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.
AGD_OPE.1.2C	The operational user guidance shall describe, for each privileged user role, how to use the available interfaces provided by the TOE in a secure manner.
AGD_OPE.1.3C	The operational user guidance shall describe, for each privileged user role, the available functions and interfaces, in particular all security parameters under the control of the privileged user, indicating secure values as appropriate.
AGD_OPE.1.4C	The operational user guidance shall, for each privileged user role, clearly present each type of security-relevant event relative to the privileged user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_OPE.1.5C	The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.
AGD_OPE.1.6C	The operational user guidance shall, for each privileged user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.
AGD_OPE.1.7C	The operational user guidance shall be clear and reasonable.
	Evaluator action elements:

AGD_OPE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_PRE.1	**Preparative Procedures**
	Developer action elements:
AGD_PRE.1.1D	The developer shall provide the TOE, including its preparative procedures.
Developer Note:	*As with the operational guidance, the developer should look to the assurance activities to determine the required content with respect to preparative procedures.*
	Content and presentation elements:
AGD_PRE.1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_PRE.1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.
	Evaluator action elements:
AGD_PRE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_PRE.1.2E	The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

6.2.3 Life-Cycle Support (ALC)

TABLE 25. ALC LIFE CYCLE SUPPORT

ALC_CMC.1	**Labeling of the TOE**
	Developer action elements:
ALC_CMC.1.1D	The developer shall provide the TOE and a reference for the TOE.
	Content and presentation elements:
ALC_CMC.1.1C	The TOE shall be labeled with its unique reference.
	Evaluator action elements:
ALC_CMC.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ALC_CMS.1	**TOE CM Coverage**
	Developer action elements:

ALC_CMS.1.1D	The developer shall provide a configuration list for the TOE.
	Content and presentation elements:
ALC_CMS.1.1C	The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.
ALC_CMS.1.2C	The configuration list shall uniquely identify the configuration items.
	Evaluator action elements:
ALC_CMS.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6.2.4 Tests (ATE)

TABLE 26. TEST ASSURANCE REQUIREMENTS

ATE_IND.1	**Independent Testing – Conformance**
	Developer action elements:
ATE_IND.1.1D	The developer shall provide the TOE for testing.
	Content and presentation elements:
ATE_IND.1.1C	The TOE shall be suitable for testing.
	Evaluator action elements:
ATE_IND.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ATE_IND.1.2E	The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

6.2.5 Vulnerability Analysis (AVA)

TABLE 27. VULNERABILITY ANALYSIS ASSURANCE REQUIREMENTS

AVA_VAN.1	**Vulnerability Survey**
	Developer action elements:
AVA_VAN.1.1D	The developer shall provide the TOE for testing.
	Content and presentation elements:
AVA_VAN.1.1C	The TOE shall be suitable for testing.

	Evaluator action elements:
AVA_VAN.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VAN.1.2E	The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.
AVA_VAN.1.3E	The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6.3 Security Requirements Rationale

The SFRs and SARs which are claimed in chapter 6 are consistent with the SFRs that are defined in the claimed Protection Profile.

7 TOE Summary Specifications

7.1 Security Audit (FAU)

TABLE 28. SECURITY AUDIT/TSS DESCRIPTION

FAU_ADP_EXT.1	Audit Dependencies
FAU_ADP_EXT.1.1	The TOE implements two complementary logging mechanisms: a Security Audit Log and a System Log, both of which are described in the operational guidance together with the set of auditable events. The Security Audit Log is generated by the EJBCA application within the TOE and records security-relevant such as certificate issuance and revocation operations, CA management actions, administrative logins, configuration changes, and failures of these operations. Each audit record is stored in a structured format containing, at a minimum, the time of the event, the event type, the outcome, the authenticated administrator identity, the functional module, the affected Certification Authority, identifiers for the affected certificate or user account when applicable, the TOE node on which the event occurred, and a free-text details field for additional diagnostic information. These audit records are persistently stored in the platform database (for example, MariaDB) in the operational environment, while their generation, structure, and integrity protection are under the control of the TOE, and they can be viewed and exported via the TOE's administrative interfaces or forwarded via configured log devices. The System Log is produced by the application server and underlying platform in the operational environment and is written to log files; it records operational and diagnostic events such as rejected invalid requests, profile accesses, and detailed runtime messages from the TLS and cryptographic providers (for example, information about fatal alerts and handshake failures), using the standard application server logging format that includes timestamp, severity level, logger identifier, thread, and message text. To see which component is responsible for generating the log, refer to tables 12, 13, and 14.
FAU_GCR_EXT.1	Generation of Certificate Repository
FAU_GCR_EXT.1.1	The certificate and CRL repository is provided by the TOE database in the operational environment. PKCS#11 is used as interface.
FAU_GEN.1	Audit data generation
FAU_GEN.1.1	The TOE minimally generates the events listed in Tables 12, 13 and 14 above and includes the date, time, event type, subject, success or failure, as well as any additional content listed in those tables.
FAU_GEN.1.2	

	Generation of ephemeral keys is not audited for FCS_CKM.1(TSF)
FAU_GEN.2	User identity association
FAU_GEN.2.1	The TOE records the responsible user in the contents of each audit record. The user identity is the target user for failed authentication attempts, or the user authenticated for the session causing the event.
FAU_SAR.1	Audit Review
FAU_SAR.1.1	The TOE provides an 'Auditors' group and stipulates that an auditor can view the signed Security Audit Log and is created to audit the operation of the TOE. The System Log is reviewed through environment interfaces.
FAU_SAR.1.2	
FAU_SAR.3	Selectable Audit Review
FAU_SAR.3.1	The TOE provides an 'Auditors' group and an auditor can view the signed Security Audit Log and is created to audit the operation of the TOE. The TSF shall provide the capability to search audit records based on the certificate serial number.
FAU_STG_EXT.1	External Audit Trail Storage
FAU_STG_EXT.1.1	The TOE generates Security Audit Log records and stores them persistently in an external relational database located on the same platform as the TOE but defined as part of the operational environment. From the TOE perspective, the database is used solely as the storage mechanism, while the TOE defines the structure and content of each audit record and uses an authenticated application-to-database connection to write the data. The TOE also produces System Log messages through the application server and container logging facilities, which are configured to send these messages to the host operating system's syslog service on the Docker host. In this way, security-relevant events are preserved in the external database while platform and runtime events are consolidated in the host syslog outside the TOE boundary but still associated with the TOE instance.
FAU_STG.4	Prevention of audit data loss
FAU_STG.4.1	In accordance with FPT_FLS.1, the TOE depends on both its external database and the host syslog service that are located in the TOE platform. Loss of either storage facility causes the TOE to enter a safe failure state. If the database reaches its storage limit or the TOE loses

	connectivity to it, the TOE detects the inability to read or write required operational and audit data, ceases certificate management and administrative processing, and presents an error message to the administrator through the management interface. Likewise, if the TOE cannot write System Log messages to the host's syslog service, the TOE is configured such that the container hosting the TOE is automatically shut down, terminating TOE services rather than allowing operation without logging. In all these cases, the TOE becomes inoperable until the underlying storage condition is corrected and the TOE is restarted, ensuring a controlled, administrator-visible failure consistent with FPT_FLS.1.

7.2 Communication (FCO)

TABLE 29. FCO COMMUNICATION, TSS DESCRIPTION

<table>
<tr><td>FCO_NRO_EXT.2</td><td>Certificate-Based Proof of Origin</td></tr>
<tr><td>FCO_NRO_EXT.2.1
FCO_NRO_EXT.2.2
FCO_NRO_EXT.2.3
FCO_NRO_EXT.2.4
FCO_NRO_EXT.2.5</td><td>The TOE uses the HSM to digitally sign certificates, EST certificate request responses, CRLs, and OCSP responses it creates using one of the algorithms in the table below as determined by the issuing certificate's key type. The TOE also uses the HSM to digitally sign the database table. The choice of key type and size is made when the crypto tokens are assigned to a CA. The table differentiates between key sizes that can be generated vs those that it can support (i.e., the PKCS#11 Cryptographic Module can be used to generate a public/private key pair using its own tools that may support different key sizes than the TOE itself can generate on the PKCS#11 Cryptographic Module).

<table>
<tr><td>Key</td><td>Supported Key Sizes</td><td>Generated Key Sizes</td></tr>
<tr><td>RSA</td><td>2048, 3072, 4096, 8192 bits</td><td>2048, 3072, 4096</td></tr>
<tr><td>ECDSA</td><td>All NIST defined B, K, and P curves with key length 256, 384, 512 bits</td><td>NIST curves: P-256, P-384, P-521</td></tr>
</table>

The TSF shall provide proof of receipt for EST by providing signed responses using mechanisms in accordance with FIA_ESTS_EXT.1.</td></tr>
<tr><td>FCO_NRR_EXT.2</td><td>Certificate-Based Proof of Receipt</td></tr>
<tr><td>FCO_NRR_EXT.2.1</td><td>The TSF shall provide proof of receipt for EST by providing signed responses using mechanisms in accordance with FIA_ESTS_EXT.1.</td></tr>
</table>

7.3 Cryptographic Support (FCS)

TABLE 30. CRYPTOGRAPHIC SUPPORT, TSS DESCRIPTION

FCS_CDP_EXT.1(TSF) and FCS_CDP_EXT.1(OE)	Cryptographic Dependencies

FCS_CDP_EXT.1.1	The TOE's WildFly application server ensures that only authenticated TLS sessions and certificates are available to the application as per the Jakarta EE standard specification. Wildfly uses the Bouncy Castle cryptographic provider for TLS/HTTPS operations via the Java Secure Socket Extension (JSSE). WildFly integrates Bouncy Castle as a JSSE security provider to perform cryptographic functions such as TLS handshakes, cipher suite negotiation, and certificate validation.
	
	Bouncy Castle provides the HMAC-DRBG(SHA-512) deterministic random number generator used in the TLS communications. In the evaluated configuration, Bouncy Castle is seeded with entropy obtained from the RHEL host operating system's /dev/random pool. This pool is continuously supplied by the Kernel CPU Time Jitter RNG, which is the entropy source validated under ESV Certificate #E54.
	
	The TOE uses its Bouncy Castle cryptographic library during TLS key exchange to generate ephemeral asymmetric keys during negotiation of TLS_ECDHE_* cipher suites.
	
	Cryptographic operations that are digital signatures, i.e. key-pair generation, signing certificates, OCSP response or protocol response messages (EST) are performed on the HSM using the standard PKCS#11 API. PKCS#11 API calls used are C_GenerateKeyPair and C_Sign.
	
	Keys to TLS authentication are generated and used in the HSM through the Java PKCS#11 provider. The application server is configured to use Java PKCS#11 keystores for the TLS keys, and all private keys are kept and used inside the HSM.
	
	• Bouncy Castle
	  ○ Cryptography supporting TLS/HTTPS
	  ○ DRBG
	• Hardware Security Module
	  ○ Cryptography supporting CA operations
	  ○ Key Generation
	  ○ Digital Signatures
	  ○ Hashing
	
	For more information, see Table 37, which provides the mapping between the SFRs and the corresponding NIST certificates.
	
	The following table attempts to break this down further:

<table>
<tr><td></td><td>
<table>
<tr><th>Functionality/SFRs</th><th>Cryptographic Provider Performing the Function</th></tr>
<tr><td>TLS/HTTPS session establishment (includes ephemeral key gen, hashing, client authentication verification and symmetric encryption as required by TLS/HTTPS)
FCS_HTTPS_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2. FCS_CKM.1(TSF) FCS_CKM.2, FCS_COP.1(1), FCS_COP.1(2)(TSF) FCS_COP.1(3)(TSF), FCS_COP.1(4), FCS_RBG_EXT.1</td><td>Bouncy castle</td></tr>
<tr><td>Asymmetric key generation of TLS/HTTPS server credentials. Digital signatures.
FCS_CKM.1(OE), FCS_COP.1(2)(OE)</td><td>PKCS#11 Cryptographic Module</td></tr>
<tr><td>Digital signature over certificate, CRL, and OCSP responses
FCS_CKM.1(OE), FCS_COP.1(2)(OE), FCS_COP.1(3)(OE)</td><td>PKCS#11 Cryptographic Module</td></tr>
</table>
</td></tr>
<tr><td>FCS_CKM.1(TSF)</td><td>Cryptographic Key Generation</td></tr>
<tr><td>FCS_CKM.1.1</td><td>The TOE uses its Bouncy Castle cryptographic library during TLS key exchange to generate ephemeral asymmetric keys during negotiation of TLS_ECDHE_* cipher suites.
The following asymmetric key sizes are supported:
<table>
<tr><th>Scheme</th><th>Size</th></tr>
<tr><td>ECC NIST curves</td><td>P-256, P-384, P-521</td></tr>
</table>
</td></tr>
<tr><td>FCS_CKM.1(OE)</td><td>Cryptographic Key Generation</td></tr>
<tr><td>FCS_CKM.1.1</td><td>The TOE relies upon an external HSM within its Operational Environment to securely generate and store RSA and ECDSA key pairs related to certificate related functions (Signature keys for RSA and ECC for signing certificates, OCSP responses, etc) as well as TLS authentication key pairs.
The following asymmetric key sizes are supported:
<table>
<tr><th>Scheme</th><th>Size</th></tr>
<tr><td>RSA</td><td>2048, 3072, 4096</td></tr>
<tr><td>ECC NIST curves</td><td>P-256, P-384, P-521</td></tr>
</table>
</td></tr>
<tr><td>FCS_CKM.2</td><td>Cryptographic Key Establishment</td></tr>
<tr><td>FCS_CKM.2.1</td><td>Elliptic curve-based key establishment schemes are used for cipher suites used for TLS/HTTPS communication.</td></tr>
</table>

<table>
<tr><td>FCS_CKM_EXT.4</td><td>Cryptographic Key Destruction</td></tr>
<tr><td>FCS_CKM_EXT.4.1

FCS_CKM_EXT.4.2</td><td>All secret keying material used for TLS communications (except for private keys stored in the HSM) including ephemeral ECDHE keys, the negotiated pre-master and master secrets, and the derived symmetric encryption and MAC keys are instantiated only in the volatile memory of the Java Virtual Machine (JVM) and JSSE implementation, which are part of the operational environment and not of the TOE.

The TOE invokes the standard TLS APIs to create and use this keying material, but does not itself implement or control the low-level storage or destruction procedures; instead, the TOE relies on the JVM and underlying operating system to manage and reclaim the process memory where these values reside.

When a TLS session is closed or the TOE process is terminated, the JSSE/JVM components in the operational environment release references to this keying material so that it becomes inaccessible to TOE code and the associated memory regions are eligible for reclamation, ensuring that no persistent copies remain within the TOE.</td></tr>
<tr><td>FCS_COP.1(1)</td><td>Cryptographic Operation (AES Encryption/Decryption)</td></tr>
<tr><td>FCS_COP.1.1(1)</td><td>The following AES algorithms are supported:
• AES-CBC (as defined in NIST SP 800-38A) mode,
• AES-GCM (as defined in NIST SP 800-38D) mode,

Key sizes of 128 and 256 bits are supported.

The TOE utilizes AES encryption to protect TLS communications in which case the TOE’s Bouncy Castle library provides the AES implementations.</td></tr>
<tr><td>FCS_COP.1(2)(OE)</td><td>Cryptographic Operation (Cryptographic Signature)</td></tr>
<tr><td>FCS_COP.1.1(2)(OE)</td><td>The following digital signature algorithms and key sizes are supported:

• RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048, 3072, 4096 bits that meets FIPS-PUB 186-4, “Digital Signature Standard”,
• Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256, 384, 512 bits that meets 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”),

Signature using RSA and ECC for TOE purposes (signing certificates, creating CRLs, OCSP responses, protocol response and TLS sessions) are performed in the HSM, using PKCS#11 C_Sign calls.

<table><tr><th>Object Signed</th><th>Crypto Module/API</th><th>Details</th></tr></table></td></tr>
</table>

Certificate request	PKCS#11 Cryptographic Module	The TOE creates certificate requests for cross-certificate signing and subordinate issuer creation. The requests are signed using the PKCS#11 Cryptographic Module.
Issued certificate	PKCS#11 Cryptographic Module	The TOE signs certificates using the PKCS#11 Cryptographic Module.
CRL	PKCS#11 Cryptographic Module	The TOE signs CRLs using the PKCS#11 Cryptographic Module.
OCSP response	PKCS#11 Cryptographic Module	The TOE signs OCSP responses using the PKCS#11 Cryptographic Module.
TLS/HTTPS ServerKeyExchange message	PKCS#11 Cryptographic Module	The TLS/HTTPS ciphersuite uses ECDHE for key agreement and the TOE uses the PKCS#11 Cryptographic Module to sign a hash value as part of the TLS negotiation process.

FCS_COP.1(2/TSF)	Cryptographic Operation (Cryptographic Signature)
FCS_COP.1.1(2)(TSF)	The following digital signature algorithms and key sizes are supported: • RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048, 3072, 4096 bits that meets FIPS-PUB 186-4, “Digital Signature Standard”, • Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256, 384, 512 bits that meets 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”), Signature verification for TLS server communication is performed by Bouncy Castle in the application server.

Object Signed	Crypto Module/API	Details
TLS/HTTPS client authentication message verification	Bouncy castle	If the TLS/HTTPS connection requires client authentication, the signature sent by the client is verified by the TOE using BC.

FCS_COP.1(3/OE)	Cryptographic Operation (Cryptographic Hashing)

FCS_COP.1.1(3)	The hashing algorithms SHA-256, SHA-384, SHA-512 and message digest sizes 256, 384, and 512 bits that meet the following: FIPS Pub 180-4, “Secure Hash Standard” are supported. Hashing for CA purposes (signing certificates, OCSP responses, protocol response) are performed in the HSM as part of the signing operation, using PKCS#11 C_Sign calls.
FCS_COP.1(3/TSF)	Cryptographic Operation (Cryptographic Hashing)
FCS_COP.1.1(3)	The hashing algorithms SHA-256, SHA-384, and SHA-512, with message digest sizes of 256, 384, and 512 bits, are supported in accordance with FIPS Pub 180-4, “Secure Hash Standard”. The TOE uses the Bouncy Castle provider to perform all cryptographic hashing, including in the following contexts: • When establishing a TLS/HTTPS connection, the TOE uses SHA-256 (and SHA-384 where applicable) as the underlying hash for the TLS 1.2 PRF and to compute the hashes that are signed and verified during the TLS handshake, as required by the configured cipher suites. • The TOE instantiates a deterministic random bit generator based on HMAC-SHA-512 (HMAC_DRBG) provided by Bouncy Castle; in this DRBG, SHA-512 is used internally for hashing to derive and update the DRBG state, and the resulting random bits are used for TLS-related randomness (such as nonces and ephemeral keys) and for other TOE cryptographic operations that require random values.
FCS_COP.1(4)	Cryptographic Operation (Keyed-Hash Message Authentication)
FCS_COP.1.1(4)	The TOE supports the HMAC algorithms HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with key sizes of 256, 384 and 512 bits and message digest sizes of 256, 384, and 512 bits, respectively, in accordance with FIPS Pub 198-1 “The Keyed-Hash Message Authentication Code (HMAC)” and FIPS Pub 180-4 “Secure Hash Standard”. During TLS/HTTPS operations, the TOE uses the Bouncy Castle provider to calculate MAC integrity checksums as required by the configured cipher suites: HMAC-SHA-256 and HMAC-SHA-384 are used for the TLS 1.2 record protection and PRF according to the selected AES-CBC cipher suite. For random number generation, the TOE instantiates a deterministic random bit generator based on HMAC-SHA-512 (HMAC-DRBG) with a 512-bit key, and uses its output to obtain at least 256 bits of entropy for TLS-related randomness (e.g. nonces and ephemeral keys) and other TOE cryptographic operations.
FCS_RBG_EXT.1	Cryptographic Random Bit Generation
FCS_RBG_EXT.1.1 **FCS_RBG_EXT.1.2**	The HSM, operating within the TOE environment, provides a hardware-based random number generator. This generator is not accessed directly by the TOE; instead, it is invoked through high-level PKCS#11 interface calls.

<table>
<tr><td></td><td>The TOE's cryptographic library, Bouncy Castle, provides a HMAC-DRBG(SHA-512) based deterministic random bit generator and output of at least 256 bits of entropy according to the entropy documentation and assessment which is utilized during TLS communication establishment (e.g., ephemeral key generation, premaster secret generation, etc.).

In the evaluated configuration, Bouncy Castle is seeded with entropy obtained from the RHEL host operating system's /dev/random pool. This pool is continuously supplied by the Kernel CPU Time Jitter RNG, which is the entropy source validated under ESV Certificate #E54.The entropy documentation explains, in detail, the sources and quantities of entropy available to the TOE.</td></tr>
<tr><td>FCS_HTTPS_EXT.1</td><td>HTTPS Protocol</td></tr>
<tr><td>FCS_HTTPS_EXT.1.1

FCS_HTTPS_EXT.1.2</td><td>The TOE provides all users (including those in administrative roles of admin, agent or auditor) with an HTTPS interface to use and configure the TOE. The user authenticates over the HTTPS interface using an x509 certificate. The user’s browser sends the user’s X.509 certificate to the TOE during the TLS negotiation. If the TOE finds the user’s certificate trustworthy (i.e., valid and chaining to a trusted root), then the TOE establishes the TLS session and TOE permits further attempted user actions depending upon that user's authorization.</td></tr>
<tr><td>FCS_STG_EXT.1</td><td>Cryptographic Key Storage</td></tr>
<tr><td>FCS_STG_EXT.1.1</td><td>
<table>
<tr><th>Key</th><th>Purpose</th><th>Storage</th><th>Protection</th></tr>
<tr><td>CA Issuers (asymmetric)</td><td>Signing certificates, CRLs, and OCSP responses</td><td>HSM</td><td>Protected by HSM</td></tr>
<tr><td>TLS/HTTPS Server Key (asymmetric)</td><td>Server Authentication</td><td>HSM</td><td>Protected by HSM</td></tr>
<tr><td>TSF Credential (asymmetric)</td><td>Encryption of CA secrets</td><td>HSM</td><td>Protected by HSM</td></tr>
<tr><td>TLS/HTTPS Client Key (asymmetric)</td><td>Authentication</td><td>Web Browser key store</td><td>Protected by the web browser</td></tr>
</table>
</td></tr>
<tr><td>FCS_TLSS_EXT.1</td><td>TLS Server Protocol</td></tr>
<tr><td>FCS_TLSS_EXT.1.1

FCS_TLSS_EXT.1.2</td><td>The following cipher suites are supported:
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289</td></tr>
</table>

FCS_TLSS_EXT.1.3	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289] The TOE supports Transport Layer Security (TLS) version 1.2 as defined in RFC 5246 and does not support any other TLS or SSL protocol versions. For non-privileged functionality, the TOE operates TLS 1.2 with server authentication only (no TLS client authentication required), allowing external entities to establish protected connections to retrieval enrolment information and perform other unauthenticated functions as defined in the operational guidance. The TSF shall generate EC Diffie-Hellman parameters over NIST curves secp256r1, secp384r1 and secp521r1 curves.
FCS_TLSS_EXT.2	TLS Server Protocol with Mutual Authentication
FCS_TLSS_EXT.2.1 **FCS_TLSS_EXT.2.2** **FCS_TLSS_EXT.2.3** **FCS_TLSS_EXT.2.4** **FCS_TLSS_EXT.2.5** **FCS_TLSS_EXT.2.6**	The following cipher suites are supported: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289] The TOE also supports TLS version 1.2, and no other TLS or SSL protocol versions, for privileged administrative functionality that require mutual authentication. For these management connections and EST requests in RA mode, the TOE is configured to request TLS client

	authentication, requiring administrators to present valid x509 client certificates that are verified by the TOE before granting access to administrative functions. In this mode, the TLS 1.2 channel provides confidentiality and integrity protection, while the combination of server and client certificate authentication ensures that only authorized administrators can access the TOE's management services. The TSF shall generate EC Diffie-Hellman parameters over NIST curves secp256r1, secp384r1 and secp521r1 curves. For each authorized administrator or client, the TOE stores an "expected identifier" that is either the complete Subject DN or a specific Subject Alternative Name (for example, an rfc822Name or dNSName) as configured in the role or account definition. During the TLS handshake, after the certificate is validated, the TOE extracts the configured identifier field (DN or SAN) from the certificate and performs an exact equality comparison against the stored expected identifier. Authentication succeeds only if the DN or SAN value in the certificate matches the configured identifier; partial, wildcard, or substring matches are not accepted.

7.4 User Data Protection (FDP)

TABLE 31. USER DATA PROTECTION, TSS DESCRIPTION

FDP_CER_EXT.1	Certificate Profiles
FDP_CER_EXT.1.1	The TOE implements a certificate profile function and issues certificates consistent with each profile's configuration. In EJBCA 9.3.3, certificate profiles are managed in the Administration Web under the "Certificate Profiles" function, where an administrator defines the Certificate Profile ID, validity period, allowed key algorithms and sizes, key usage and extended key usage bits, basic constraints, name constraints, and other X.509v3 extensions and default properties. Certificate profiles are then referenced from End Entity Profiles, which bind a given profile to one or more CAs; when a certificate is issued (via the RA GUI, Administration Web, web services, or protocol interfaces such as EST), the TSF automatically selects the configured certificate profile for the end entity and populates the resulting X.509 certificate fields strictly according to that profile. Proof-of-possession of the private key corresponding to the certificate request is verified by the TSF as described in Section 7.2, by validating the signature over the PKCS#10 certificate request (or the corresponding protocol-specific proof-of-possession) using the public key contained in the request before issuing the certificate. Certificate serial numbers are generated by the TOE as random values and placed in the X.509 serialNumber field in accordance with FDP_CER_EXT.1.3. For each certificate profile, the administrator configures the serial-number length (in bytes) in the EJBCA 9.3.3 Administration Web up to a maximum of 20 bytes; for the CC configuration, this length is selected so that the serial number contains at least 20 bits of entropy. The TSF obtains the serial number from the Bouncy Castle random bit generator that satisfies FCS_RBG_EXT.1.
FDP_CER_EXT.1.2	
FDP_CER_EXT.1.3	
FDP_CER_EXT.2	**Certificate Request Matching**
FDP_CER_EXT.2.1	Each certificate request is identified by a unique request ID which is linked to the issued certificate and can be verified in the System Log under the field "task". Each certificate is identified by a unique issuer DN and serial number.
FDP_CER_EXT.3	**Certificate Issuance Approval**
FDP_CER_EXT.3.1	The TOE supports the approval of certificates issued if an approval profile has been associated with the certificate profile or the corresponding CA through the RA management interface. Only user roles given "Approve End Entity Actions" permission can approve certificates via the web interface (which is the only interface through which manual issuance occurs).

FDP_CRL_EXT.1	Certificate Revocation List Validation
FDP_CRL_EXT.1.1	The TOE supports X.509 v2 CRL generation for each issuing CA, either on demand or according to a schedule configured per CA in the Administration Web (for example, CRL issue interval and overlap time). An administrator can trigger CRL generation explicitly using the "Create CRL" function and the TSF will construct the CRL from the revocation information maintained for that CA and publish it through the configured CRL publishers. Issued CRLs conform to ITU-T Recommendation X.509 and include at least the fields and extensions required by FDP_CRL_EXT.1.1: a) If the version field is present, then it shall contain a 1. b) If the CRL contains any critical extensions, then the version field shall be present and contain the integer 1. c) 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. d) The signature and signatureAlgorithm fields shall contain the OID for a digital signature algorithm in accordance with FCS_COP.1(2). e) The thisUpdate field shall indicate the issue date of the CRL. f) The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.
FDP_CSI_EXT.1	**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, the OCSP standard as defined by RFC 6960. Users with 'Revoke End Entities' permission, or subscribers, can approve changes to the status of a certificate. This can be done via the TOE's web interface and the RA management interface. The process varies based on the interface.
FDP_CSI_EXT.1.2	
FDP_OCSPG_EXT.1	**OCSP Basic Response Generation**
FDP_OCSPG_EXT.1.1	The TOE supports OCSP and includes an OCSP responder that can be enabled per CA; when enabled, the TSF accepts OCSP requests over HTTPS and generates BasicOCSPResponse structures for the requested certificate identifiers in accordance with FDP_OCSPG_EXT.1.1 and the standards referenced by FDP_CSI_EXT.1. At a minimum, the following items are enforced*:* a) The version field shall indicate a current version. b) The signatureAlgorithm field shall contain the object identifier (OID) for a digital signature algorithm in accordance with FCS_COP.1(2). c) The thisUpdate field shall indicate the time at which the status being indicated is known to be correct. d) The producedAt field shall indicate the time at which the OCSP responder signed the response.

	e) The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.
FDP_RIP.1	**Subset Residual Information Protection**
FDP_RIP.1.1	The TOE does not store any personally identifiable information beyond what is contained in issued certificates.The cryptographic keying material used for TLS (session keys, ephemeral ECDHE keys, and related DRBG state) is maintained only in volatile process memory within Java objects and TLS session structures. When a TLS session completes, the TSF releases all references to the associated session objects and does not copy them into reusable buffers shared with other connections or into any persistent storage; at this point in the processing of the corresponding buffers, the previous information content is no longer accessible via any TOE interface. The Java Virtual Machine and underlying operating system in the operational environment then reclaim and overwrite the freed memory in the course of normal garbage collection and memory management. For key material resident in the HSM, residual information protection is provided by the HSM itself.

7.5 Identification and Authentication (FIA)

TABLE 32. IDENTIFICATION AND AUTHENTICATION, TSS DESCRIPTION

FIA_ESTS_EXT.1	**Enrollment over Secure Transport (EST) Server**
FIA_ESTS_EXT.1.1 FIA_ESTS_EXT.1.2 FIA_ESTS_EXT.1.3 FIA_ESTS_EXT.1.4	The TOE supports the Enrollment over Secure Transport (EST) protocol as described in RFC 7030 and implements the simple enrollment method in Section 4.2 to receive and act upon certificate enrollment requests. For each EST enrollment request, the TOE authenticates the requester either using a TLS client certificate and corresponding private key as specified in RFC 7030 Section 3.3.2, the authenticated identity is then mapped to a TOE account through the role-based access control mechanism. For initial issuance and authorization of certificates, the TOE supports a Registration Authority (RA) role: EST clients whose authenticated identity is mapped to a configured RA or administrative role are permitted to submit and approve enrollment requests on behalf of subscribers in accordance with the applicable End Entity and Certificate Profiles. The authorization for RA functions is determined by the TOE's internal roles and access rules associated with the authenticated EST client.
FIA_X509_EXT.1	**Certificate Validation**
FIA_X509_EXT.1.1	

FIA_X509_EXT.1.2	The TSF performs X.509 certificate and certificate path validation in software using the Java PKIX/JSSE/Bouncy Castle validation logic in the application server. For each certificate presented to the TOE (for example, TLS client certificates for administrator authentication, certificates used for EST, and OCSP responder certificates), the TSF constructs and validates a certification path in accordance with RFC 5280, starting from the end-entity certificate and terminating at a trust anchor stored in the TOE's configured trust store or Trust Anchor Database: The TSF verifies the following parameters: – 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 the Online Certificate Status Protocol (OCSP) as specified in RFC 6960. – The TSF shall validate the extendedKeyUsage field according to the following rules: o 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) in the extendedKeyUsage field. o Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the EKU field. o 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 EKU field. o OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the EKU field. o Server certificates presented for EST shall have the CMC Registration Authority (RA) purpose (id-kp-cmcRA with OID 1.3.6.1.5.5.7.3.28) in the EKU field.
FIA_X509_EXT.2	**Certificate-Based Authentication**
FIA_X509_EXT.2.1	The TOE uses X.509v3 certificates, as defined by RFC 5280, to authenticate privileged users over HTTPS. Certificates can also be used for EST access authentication. The TSF uses the certificate presented by the user. When the TSF cannot determine the current revocation status of a certificate the operation will be rejected and an error message will be displayed in the web portal.
FIA_X509_EXT.2.2	
FIA_X509_EXT.2.3	
FIA_X509_EXT.3	**X509 Certificate Request**
FIA_X509_EXT.3.1	

FIA_X509_EXT.3.2	The TSF generates CA certificate request messages in PKCS#10 format as specified in RFC 2986 via the “Create CA – Signed by external CA” workflow, where it creates an internal CA key pair and constructs a request that includes only the CA’s public key and distinguished name as the PKCS#10 subject, with no additional attributes.
FIA_UAU_EXT.1	**Authentication Mechanism**
FIA_UAU_EXT.1.1	NA
FIA_UIA_EXT.1	**User Identification and Authentication**
FIA_UIA_EXT.1.1 FIA_UIA_EXT.1.2 FIA_UIA_EXT.1.3	Identification and authentication are required for all the administrative actions of the TOE user roles. The TOE supports the following successful logon outcomes: – Web GUI access (Admin Web / RA Web) over HTTPS (TLS 1.2): Successful logon occurs when an HTTPS/TLS session is established with mutual TLS (mTLS), the client presents a valid X.509 certificate trusted by the TOE, and the TOE maps the authenticated identity to an authorized role, thereby granting access to the Admin Web or RA Web GUI. – EST access over HTTPS (TLS 1.2): Successful logon occurs when an EST client successfully authenticates to the TOE over HTTPS/TLS using mTLS (a valid client certificate trusted by the TOE). Prior to requiring a non-TOE entity to initiate identification and authentication, the TSF permits only non-privileged operations over HTTPS/TLS 1.2. Specifically, the TSF: – Displays the configured warning and consent banner in accordance with FTA_TAB.1 before access to any administrative or RA GUI function is granted. – Provides access to public PKI information such as CA certificates, CRLs, and certificate status information via the TOE interface (e.g. CRL download and status lookup pages) without requiring user authentication.

7.6 Security Management (FMT)

TABLE 33. SECURITY MANAGEMENT, TSS DESCRIPTION

FMT_MOF.1(1)	**Management of Security Functions Behavior (Administrator Functions)**

FMT_MOF.1.1(1)	NA
FMT_MOF.1(2)	**Management of Security Functions Behavior (CA/RA Functions)**
FMT_MOF.1.1(2)	NA
FMT_MOF.1(3)	**Management of Security Functions Behavior (CA Operations Functions)**
FMT_MOF.1.1(3)	NA
FMT_MOF.1(4)	**Management of Security Functions Behavior (Admin/Officer Functions)**
FMT_MOF.1.1(4)	The TOE enforces role-based access control on all management functions exposed through the Administration GUI and RA Web. Administrative access to the TSF requires mutual TLS authentication with an administrator certificate that is mapped to a EJBCA role (Super Administrator, CA Administrator, RA Administrator, Auditor). For each authenticated request, the TSF checks the caller's role(s) against an internal access rule set and enables, read-only exposes, or denies the requested management function (menu items, buttons, URLs). Functions allocated to the Operational Environment (OE) – such as local system management, software update, platform integrity checks, audit storage configuration, and secure deletion of data – are restricted by operating system, database, and application-server mechanisms and are only available to OS/platform administrators. Administrator / Super Administrator The Administrator/Super Administrator role is the global management role for the TSF. It can remotely manage all TSF configuration, including certificate and end-entity profiles, revocation and CRL settings, OCSP responder configuration and behaviour, allowed algorithms and key sizes, crypto tokens and key generation/activation in the cryptographic module, import and deletion of trust anchors (external CA certificates), export of CA PKCS#10 requests, approval of certificate issuance and revocation, subscriber self-service constraints, automated approval profiles, and viewing/searching the TSF audit log. It cannot perform OE-only actions (software installation/update, low-level audit mechanism configuration, file/database deletion, physical media destruction) unless the same person also holds OE administrator credentials, and it cannot delete stored audit records via any TSF interface. CA Administrator The CA Administrator role provides operational management of specific CAs. Within the scope of the CAs assigned to it, it can manage related certificate and end-entity profiles, adjust revocation and CRL configuration, approve certificate issuance and revocation,

	and configure subscriber self-service constraints and approval profiles for those CAs. It cannot create or manage crypto tokens or generate new keys, cannot configure or create OCSP responders or change global validation behaviour, cannot import or delete trust anchors or export CA PKCS#10 requests, cannot access the TSF audit log, and cannot perform any OE-allocated management functions. RA Administrator The RA Administrator role supports day-to-day registration and handling of end entities and certificate requests in RA Web, limited to the CAs and profiles assigned to it. It may view and process registration data in accordance with the configured policies. It cannot configure certificate or end-entity profiles, cannot configure approval profiles or automated approval behaviour, cannot approve certificate issuance or revocation when approval is required, cannot manage CAs, revocation, CRLs, services, crypto tokens, OCSP responders, trust anchors, or PKCS#10 requests, cannot view or search the TSF audit log, and cannot perform any OE-allocated functions. Auditor The Auditor role is a read-only supervision role. It can access and search the TSF audit log via the Audit Log supervision function and can view certain configuration information in read-only form (for example, CA, service or profile information) where permitted by the TSF. It cannot modify any TSF configuration, cannot approve or execute certificate issuance or revocation, cannot manage profiles, CAs, revocation, OCSP, crypto tokens, trust anchors or PKCS#10 requests, and cannot delete or modify stored audit records; deletion or truncation of audit data is an OE function. Operational Environment Administrators Operational Environment administrators (system, database and application-server administrators) manage functions allocated to the OE: local TOE management (deployment, start/stop, platform configuration), software and patch updates, configuration of underlying audit storage and rotation, execution of platform-level integrity checks, secure deletion of TOE data (databases, backups, log files, media), and deletion/truncation of stored audit records using OS/DB tools. These privileges are controlled by the OE and are not available through TSF interfaces. OE administrators do not automatically have TSF administrative rights; to manage the TSF they must also possess an appropriate administrator certificate and TSF role.
FMT_MOF.1(5)	**Management of Security Functions Behavior (Auditor Functions)**

FMT_MOF.1.1(5)	NA
FMT_MTD.1	**Management of TSF Data**
FMT_MTD.1.1	The TOE provides no access to administrative functions to unauthenticated users that have not presented a valid client certificate.
FMT_SMF.1	**Specification of Management Functions**
FMT_SMF.1.1	NA
FMT_SMR.2	**Restrictions on Security Roles**
FMT_SMR.2.1 FMT_SMR.2.2 FMT_SMR.2.3	The TSF, and no other component, maintains the following roles: – Administrator → Super Administrator – Auditor → Auditor – CA Operations Staff → CA Administrator – RA Staff → RA Administrator – Authorized Organizational Representative → Supervisor Each role is realized as an EJBCA administrative role, enforced by the TSF through role-based access control on the Administration GUI and RA Web. Access is over HTTPS with mutual TLS; the authenticated certificate is mapped to one TOE role, and the corresponding fixed privilege set is applied for the entire session. Administrator (Super Administrator): Full TSF management via the Admin GUI and RA Web (CAs, profiles, revocation/CRL/OCSP, crypto tokens and key generation, trust anchors, PKCS#10 CA requests, approvals, subscriber self-service and approval profiles, audit log viewing). CA Operations Staff (CA Administrator): Management of assigned CAs via the Admin GUI (certificate/end-entity profiles, revocation configuration, approval of issuance and revocation, subscriber self-service constraints and approval profiles for those CAs). No access to crypto token/key generation, OCSP/validation behaviour, trust anchor import/removal, CA PKCS#10 export, audit log viewing, or OE-only functions. RA Staff (RA Administrator): RA Web operations for registration and handling of end entities and requests within assigned CAs/profiles. No configuration of profiles, approvals, CAs, revocation/CRL/OCSP, services, crypto tokens, trust anchors or PKCS#10, no audit log access, and no OE-only functions. Auditor (Auditor): Read-only supervision via the Admin GUI, with access to view/search the TSF audit log and selected configuration data in read-only mode. No configuration changes, no issuance/revocation approvals, no management of profiles, CAs, revocation, OCSP, crypto tokens, trust anchors or PKCS#10, and no deletion or modification of stored audit records.

	Authorized Organizational Representative (Supervisor): Use of RA Web to submit and track certificate requests in accordance with configured end-entity profiles and approval workflows. No TSF management capabilities and no OE-only privileges. Role separation is enforced by: – TSF mechanisms that map each administrative certificate to exactly one TOE role with a fixed privilege set, without in-session role switching or elevation; and – guidance-mandated procedures requiring distinct certificates with different characteristics (for example, OU and profiles) per role, so that a single certificate is not assigned multiple roles.

7.7 Protection of the TSF (FPT)

TABLE 34. PROTECTION OF THE TSF, TSS DESCRIPTION

FPT_FLS.1	**Failure with Preservation of Secure State**
FPT_FLS.1.1	The TOE depends on both its external database and the external HSM and that loss of either causes the TOE to enter a secure failure state. If the database becomes unavailable (for example, connectivity loss or storage exhaustion) or the HSM cannot be reached or returns a cryptographic failure, the TOE detects the condition and stops all certificate management, enrollment, and TLS services instead of continuing with degraded security.
FPT_KST_EXT.1	**No Plaintext Key Export**
FPT_KST_EXT.1.1	The list of private keys used by the TOE and its processes comprises: – defaultKey (required alias): the primary private key selected by default for CA operations. – certSignKey: the private key used for certificate signing (issuance). – alternativeCertSignKey: the private key used for alternative certificate signing operations. – crlSignKey: the private key used for CRL signing. – keyEncryptKey (if Key Recovery is enabled): a private key used to protect/recover archived key material. – testKey: a key used for Crypto Token health checks (a quick verification that the token/HSM is usable). – OCSP signing key: the private key of the “OCSP Key Binding” stored in a Crypto Token (HSM) and used to sign OCSP responses. There is no way to export private keys used by the TSF or users from the TOE. The private keys are stored in the HSM. The HSM does not provide any user-accessible mechanism to retrieve private key

	material; private keys are generated and remain non-exportable within the HSM boundary and are only usable via HSM-mediated cryptographic operations.
FPT_KST_EXT.2	**TSF Key Protection**
FPT_KST_EXT.2.1	The TOE relies on the external HSM accessed via the PKCS#11 interface to protect all private keys stored in the HSM. The HSM does not provide any user-accessible mechanism to retrieve private key material; private keys are generated and remain non-exportable within the HSM boundary and are only usable via HSM-mediated cryptographic operations. Administrative access to the HSM is restricted to authorized HSM administrators through the HSM's management interfaces and is separated from application use of keys. For programmatic access, the TOE authenticates to the HSM by establishing a PKCS#11 session and performing token login using the configured HSM credential associated with the TOE's PKCS#11 token/slot. Only after successful authentication does the HSM permit cryptographic operations using the protected private keys. Within the TOE, use of HSM-protected keys is mediated by TOE role-based access control: only users assigned the appropriate authorized roles can invoke TOE functions that cause the TSF to use those keys (e.g., certificate/CRL signing), and no TOE user is provided an interface to export or otherwise access private key material.
FPT_RCV.1	**Manual Trusted Recovery**
FPT_RCV.1.1	In this state, following a PKCS#11 cryptographic token/HSM failure or loss of access to the database, no further administrative or subscriber operations are performed, an error condition is reported to the administrator, and all CA private keys remain protected within the HSM, so that key material and user data are not exposed until the underlying failure is corrected and the TOE is restarted.
FPT_SKP_EXT.1	**Protection of Keys**
FPT_SKP_EXT.1.1	The cryptographic keys used by the TOE in the evaluated configuration are: – CA and OCSP keys stored in the HSM: the CA key aliases used for CA operations and signing (defaultKey, certSignKey, alternativeCertSignKey, crlSignKey, keyEncryptKey (if enabled), testKey) and the OCSP Key Binding signing key. These keys are generated and used via the PKCS#11 Cryptographic Module and are non-exportable from the HSM. – TLS authentication key stored in the HSM: the TOE's TLS/HTTPS server private key used for server authentication. The application server is configured to use the PKCS#11 keystore so that the TLS private key is kept and used inside the HSM. – TLS session keying material (ephemeral): ephemeral ECDHE keying material and the derived TLS session secrets/record-

	protection keys are generated during TLS negotiation (e.g., TLS_ECDHE_* cipher suites) and exist only transiently in volatile memory within the container's JVM/JSSE implementation; they are not stored persistently and are destroyed when the session ends.
FPT_STM.1	**Reliable Time Stamps**
FPT_STM.1.1	The TOE obtains the current time from the underlying OS that hosts the Docker container. The current system time is used when: Generating audit records, issuing certificates, CRLs, and signing OCSP responses. The SFRs that use time are: FAU_GEN.1.2, FCO_NRO_EXT.2.2, FCS_HTTPS_EXT.1, FCS_TLSS_EXT.1, FDP_CER_EXT.3, FDP_CSI_EXT.1, FDP_CRL_EXT.1, FDP_OCSP_EXT.1.1, FIA_X509_EXT.1, and FIA_X509_EXT.2.
FPT_TUD_EXT.1	**Trusted Update**
FPT_TUD_EXT.1.1	In the evaluated configuration, the TOE is distributed and updated exclusively as a container image published in Keyfactor's vendor container registry. Updates are applied by deploying a newer container image release. Keyfactor is the authorized source for TOE updates; therefore, only images that carry a valid Keyfactor signature and pass the administrator's signature-verification checks are treated as acceptable update candidates. The update process is performed from the platform OS by an administrator. The administrator selects the candidate TOE image in the Keyfactor registry (by reference and immutable digest) and verifies the vendor signature against the registry-hosted image before proceeding. Only after a successful verification does the administrator retrieve and deploy the corresponding container image. If verification fails, the verification tool reports an error to the administrator, and the update is stopped; the candidate image is not retrieved for use and is not deployed. Signature verification is performed using an Operational Environment utility (Cosign). Cosign verifies that the signature corresponds to the container image digest, providing integrity for the specific image identified in the registry. For authorized-source assurance, Cosign uses the Keyfactor-provided signing certificate together with the associated certificate chain to perform certificate path validation. The signing certificate is accepted only if its certification path terminates at a trusted container-signing trust anchor (i.e., the trusted public key/certificate at the end of the validated chain), which is maintained under administrative control on the platform. This certificate and chain validation is performed consistent with FIA_X509_EXT.1. In addition, the signing certificate is required to assert the Code Signing purpose in the extendedKeyUsage extension. The signature scheme uses an RSA-4096 public key and SHA-384.
FPT_TUD_EXT.1.2	
FPT_TUD_EXT.1.3	
FPT_TUD_EXT.1.4	
FPT_TUD_EXT.1.5	
FPT_TUD_EXT.2	**Integrity for Installation and Update**
FPT_TUD_EXT.2.1	

FPT_TUD_EXT.2.2	Refer to FPT_TUD_EXT.1.
FPT_TUD_EXT.2.3	

7.8 TOE Access (FTA)

TABLE 35. FTA – TOE ACCESS, TSS DESCRIPTION

FTA_SSL.4	**User-Initiated Termination**
FTA_SSL.4.1	An administrator using a web browser can close their browser or click button “Logout” in order to close their TLS session.
FTA_TAB.1	**Default TOE Access Banners**
FTA_TAB.1.1	Before establishing a privileged user session over the administrative HTTPS/TLS interface the TSF displays an Administrator-configured advisory notice and consent warning message regarding unauthorized use of the TOE.

7.9 Trusted Path/Channels (FTP)

TABLE 36. FTP– TRUSED PATH/CHANNELS

FTP_TRP.1	**Trusted Path**
FTP_TRP.1.1	The TSF uses HTTPS/TLS 1.2 for the trusted communication path to the remote subscribers and privileged users.
FTP_TRP.1.2	
FTP_TRP.1.3	

7.10 CAVP algorithm

TABLE 37. CAVP ALGORITHM

Requirement Class	**Requirement Component**	**Implementation Details**	**Module**	**Certificate #**
FCS: Cryptographic Support	FCS_CKM.1 (TSF) Cryptographic Key Generation	Generating P-256, P-384 and P-521 ECDSA keypairs conforming to FIPS 186-5.	Bouncy Castle Java API	A7617

Requirement Class	Requirement Component	Implementation Details	Module	Certificate #
	FCS_CKM.1 (OE) Cryptographic Key Generation	Generating 2048, 3072 and 4096-bits RSA keypairs conforming to FIPS 186-4. Generating P-256, P-384 and P-521 ECDSA keypairs conforming to FIPS 186-4.	SafeNet Accelerated Cryptographic Library for Thales DPoD Luna 7	RSA 3042 A1171 ECDSA 1526
	FCS_CKM.2 Cryptographic Key Establishment	Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;	Bouncy Castle Java API	A7617
	FCS_COP.1(1) Cryptographic Operation (AES Data Encryption/Decryption)	AES encryption and decryption used in CBC, GCM mode with 128-bit, 256-bit key sizes validated conforming to FIPS PUB 197.	Bouncy Castle Java API	A7617
	FCS_COP.1(2)(OE)/ SigGen Cryptographic Operation (Signature Generation)	RSA signature generation and verification with 2048-bit, 3072-bit and 4096-bit key sizes according to FIPS 186-4. ECDSA signature generation and verification using P-256, P-384 and P-521 curves according to FIPS 186-4.	SafeNet Accelerated Cryptographic Library for Thales DPoD Luna 7	RSA 3042 A1170 A1171
	FCS_COP.1(2)(TSF)/ SigGen Cryptographic Operation (Signature Generation)	RSA signature generation and verification with 2048-bit, 3072-bit and 4096-bit key sizes according to FIPS 186-4. ECDSA signature generation and verification using P-256,	Bouncy Castle Java API	A7617

Requirement Class	Requirement Component	Implementation Details	Module	Certificate #
		P-384 and P-521 curves according to FIPS 186-4.		
	FCS_COP.1(3)(TSF)/ Cryptographic Operation (Hash Algorithm)	Hashing using SHA-256, SHA-384, SHA-512 validated conforming to FIPS 180-4, Secure Hash Standard (SHS).	Bouncy Castle Java API	A7617
	FCS_COP.1(3)(OE)/ Cryptographic Operation (Hash Algorithm)	Hashing using SHA-256, SHA-384 and SHA-512 validated conforming to FIPS 180-4, Secure Hash Standard (SHS).	SafeNet Accelerated Cryptographic Library for Thales DPoD Luna 7	SHS 4533
	FCS_COP.1(4)/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)	Keyed hash HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 validated conforming to FIPS 198, Keyed-Hash Message Authentication Code (HMAC). Supported message digest sizes: 256, 384 and 512 bits.	Bouncy Castle Java API	A7617
	FCS_RBG_EXT.1.1 Cryptographic Operation (Random Bit Generation)	HMAC-DRBG (SHA2-512) random bit generation validated conforming to NIST SP PUB 800-90A.	Bouncy Castle Java API	A7617
	FCS_RBG_EXT.1.2 Cryptographic Operation (Random Bit Generation)	The entropy source provides an output of 256 bits of entropy.	RHEL Kernel CPU Time Jitter RNG Entropy Source	E54
	FCS_TLSS_EXT.1 FCS_TLSS_EXT.2	KDF TLS 1.2	Bouncy Castle Java API	A7617

8 References

- [1] http://tools.ietf.org/pdf/rfc5280.pdf - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.
- [2] https://datatracker.ietf.org/doc/html/rfc6960: RFC 6960 - X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP.
- [3] BSI, Technical Guideline TR-03110*:* Advanced Security Mechanisms for Machine Readable Travel Documents – Extended Access Control (EAC), Password Authenticated Connection Establishment (PACE),and Restricted Identification (RI).
- [4] X.509 : Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks (itu.int). ITU-T X.509: Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks.
- [5] http://tools.ietf.org/pdf/rfc5019.pdf The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments.
- [6] http://www.ejbca.org/ EJBCA Project Website.
- [7] EJBCA 9.3.3 Enterprise Common Criteria Guidance Supplement, version 1.0, 2025-10-31
- [8] Protection Profile for Certification Authorities, version 2.1, 2017-12-01, National Information Assurance Partnership

9 Glossary

ACME	Automatic Certificate Management Environment
API	Application Programming Interface
CA	Certification Authority
CC	Common Criteria
CEM	Common Evaluation Methodology
CIMC	Certificate Issuing and Management Components Protection Profile
CIMS	Certificate Issuing Management System
CMP	Certificate Management Protocol
CP	Certificate Policy
CPS	Certification Practices Statement
CRL	Certificate Revocation List
CRMF	Certificate Request Message Format
CVC	Card Verifiable Certificates
CWA	CEN Workshop Agreement
EAC	Extended Access Control
EAL	Evaluation Assurance Level
EJB	Enterprise Java Bean
EST	Enrollment over Secure Transport
FIPS	Federal Information Processing Standard
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
IT	Information Technology
JDBC	Java Database Connectivity
JEE	Java Enterprise Edition
JVM	Java Virtual Machine
NIST	National Institute of Standards and Technology
OCSP	Online Certificate Status Protocol
PKI	Public Key Infrastructure
PKCS#10	Certification Request Syntax Standard
PKCS#11	Cryptographic Token Interface Standard

PP	Protection Profile
RA	Registration Authority
SAR	Security Assurance Requirement
SF	Security Functions
SFP	Security Functions Policy
SFR	Security Functional Requirement
SOF	Strength of TOE Security Functions CC components (deprecated since CC 3.1)
SPM	Security Policy Modelling CC components (deprecated since CC 3.1)
SQL	Structured Query Language
SSCD	Secure Signature Creation Device
TOE	Target of Evaluation
TSF	TOE Security Functionality
VA	Validation Authority
VAN	Vulnerability Analysis CC components
VM	Virtual Machine