SecuGATE Version 4.0 (NDcPP21) Security Target

Version 0.7
December 19, 2019

Prepared for:
BlackBerry Ltd.

Prepared By:

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is SecuGATE SIP Server provided by BlackBerry Ltd. The TOE is being evaluated as an Netwrok Device, specifically an Enterprise Session Controller.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

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 parenthetical number placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – SecuGATE Version 4.0 (NDcPP21) Security Target

ST Version – Version 0.7

ST Date December 19, 2019

1.2 TOE Reference

TOE Identification – BlackBerrySecuGATE SIP Server v4.0

TOE Developer – BlackBerry Ltd

Evaluation Sponsor – BlackBerry Ltd

1.3 TOE Overview

The Target of Evaluation (TOE) is SecuGATE SIP Server v4.0. The SecuGATE SIP Server v4.0 enables use of the Session Initiation Protocol (SIP) to establish secure connections between mobile devices. The SecuGATE SIP server runs on RHEL 7.6 OS within an ESXi version 6.5 virtualized environment using a physical platform which includes an Intel Xeon E3-1240, Xeon E3-1515 or Xeon Gold 5218 processor including

- the SUPERMICRO system with an Intel Xeon E3-1240,
- the SUPERMICRO system with an Intel Xeon Gold 5218. and
- the PacStar 451 system with an Intel Xeon E3-1515.

The SecuGATE SIP Server is the centerpiece in the SecuSUITE Security Solution. The SecuSUITE Security Solution includes the SecuGATE SIP server and client software[1] for mobile device platforms. Together these form a system that provides end-to-end secure mobile voice communication and instant messaging, using IP-based mobile data connections such as EDGE, UMTS/HSPA, LTE, and Wi-Fi.

This Security Target (ST) pertains to only the SecuGATE SIP Server v4.0 component.

1.4 TOE Description

The TOE is the SecuGATE SIP server version 4.0. The SecuGATE SIP Server enables use of the Session Initiation Protocol (SIP) to establish secure connections between mobile devices.

The SecuGATE SIP Server is an infrastructure component of the SecuSUITE Security Solution shown in Figure 1 below. The SIP Server does not work in isolation but relies on other infrastructure components to enable secure VoIP communications.

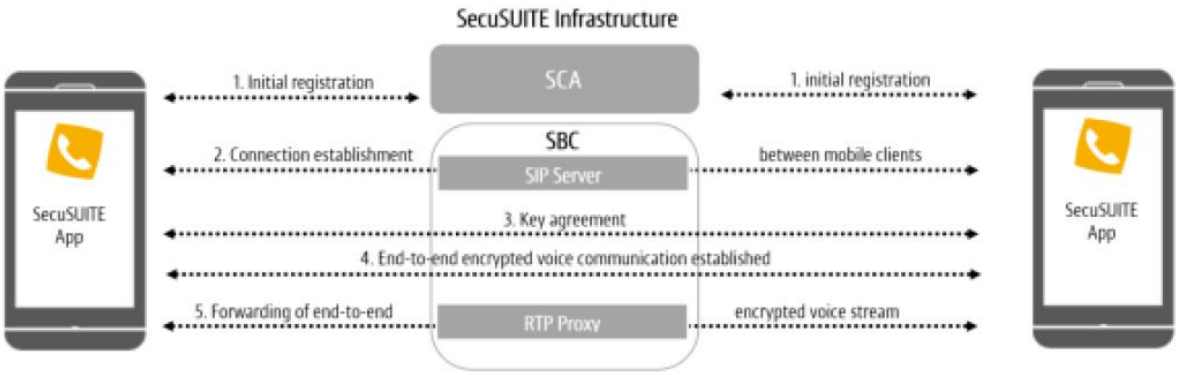

Figure 1-1 SecuSUITE Security Solution

As shown in Figure 1, the SecuSUITE VoIP process flow is as follows:

a) Step 1 Initial Registration. Every participating client has to register first to the Secure Client Authentication (SCA) server. The SCA server authenticates users and credentials to access services. Only clients that have been enrolled via the SCA service are able to connect to the SIP server and are allowed to establish end-to-end encrypted communication to other SecuSUITE clients. Note: Clients must also register to the SIP server using a SIP password. This is in addition to initial client registration with the SCA server.

b) Step 2 Connection establishment. The Session Initiation Protocol (SIP) together with TLS is used to establish a secure connection between mobile devices and the SIP server (aka SIP Calling). The use of a TLS connection, providing encryption and mutual authentication, ensures that the devices connect with authorized SIP servers and the dialled call numbers are transmitted encrypted.

[1] The client software is the target for another evaluation.

c) Step 3 Key agreement. When a call is placed and accepted, SecuSUITE clients exchange SIP messages that include digital certificates used to confirm caller identity and perform key agreement for SRTP encryption.

d) Step 4 End-to-end encrypted voice communication established. Clients utilize the SRTP protocol to exchange encrypted voice communications. The voice stream remains encrypted while traversing the SecuSUITE infrastructure and only the clients have access to the session keys.

e) Step 5 Forwarding of end-to-end encrypted voice stream. During connection signaling, the SIP server sets up the RTP/RTCP packet bridging in the Real-Time Transport Protocol (RTP) Proxy for this connection. The RTP Proxy relays / bridges the encrypted data stream between clients. The main purpose of the RTP Proxy is to make the communication between SIP user agents behind NAT/NAPT possible.

1.4.1 TOE Architecture

The SecuGATE SIP Server v4.0 is network appliance providing SIP server, RTP Proxy and SCA functionality as well as interfaces for management. The SecuGATE SIP Server TOE is composed of hardware, a hardened Red Hat Enterprise Linux OS (the TOE does not offer general purpose computer capabilities), and custom software. The custom software provides SIP server, RTP Proxy and SCA functionality. It runs on a Red Hat Enterprise Linux (RHEL 7.6) and utilizes the OpenSSL FIPS object module along with other supporting software.

Specifically, the TOE utilizes the OpenSSL 1.0.2 FIPS object module v2.0.16 which provides cryptographic functionality used by the TOE. The TOE's software executes on the RHEL 7.6 operating system on ESXi on a physical platform as specified in section 1.3 above.

1.4.1.1 Physical Boundaries

The TOE boundary is illustrated in Figure 2.

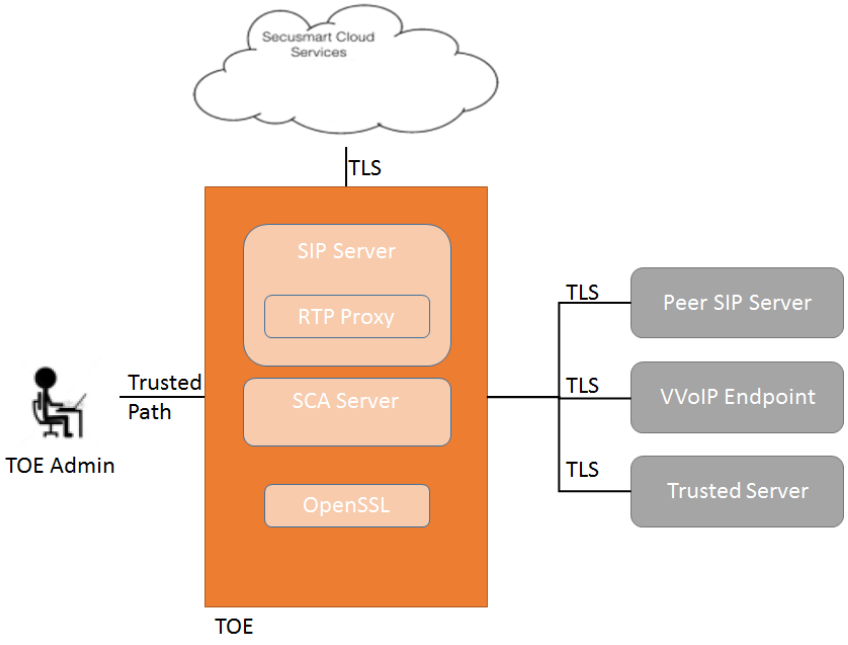

Figure 1-2 TOE Boundary

The TOE operates in a network environment mediating connections between VVoIP endpoints while utilizing services from other network entities.

SIP Server Functionality

The SIP Server interacts with the SecuSUITE VoIP client and provides registrar and proxy capabilities required for call-session management (e.g. establishing, processing, and terminating VoIP calls). As a SIP registrar, the SIP Server accepts REGISTER requests and places the information received into the location service on the SIP Server. As a SIP proxy server, the SIP Server is a stateful server that manages transactions to route SIP requests and responses. The SIP Server also provides a secure connection between mobile devices running the SecuSUITE app using TLS, providing encryption and mutual authentication.

RTP Proxy Functionality

The Real-time Transport Protocol (RTP) Proxy bridges media packets sent between clients. The TOE creates and deletes RTP and Real-time Transport Control Protocol (RTCP) bridging sessions in the RTP Proxy.

Secure Client Authentication Functionality

The SCA functionality authenticates users, facilitates VoIP client enrollment and pushes client SIP configuration to the client. Only clients which have been enrolled via the SCA service are able to connect to the SIP server. During SCA enrollment the SCA authorizes authenticated clients (via activation code) to use SIP service and provisions them with the SIP credentials and a TLS client certificate for the required trusted channel.

NON-TOE Components

The TOE is part of a broader system (SecuSUITE security solution) and requires the following components to be present in the environment:

a) Audit server. The TOE is able to send audit logs to a remote syslog server.

b) NTP Server. The TOE is able to obtain time from an NTP server over a TLS protected session.

c) Peer SIP server. The TOE can communicate with another SIP server (such as Asterisk SIP or similar) over TLS.

d) Push Server. The TOE can communicate with a push notification server that allows the VVoIP endpoint OS to execute deep sleep cycles and wake-up client applications for incoming events.

e) VVoIP Endpoints. The TOE mediates connections initiated by a VVoIP client enrolled through the SCA Server to another VVoIP endpoint.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by SecuGATE SIP Server:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE generates audit events for numerous activities including policy enforcement, system management, authentication and system status (i.e., system log records). The TOE also generates call detail records providing information about connections that are mediated by the TOE. A syslog server in the environment is relied on to store audit and system log records generated by the TOE. The TOE generates a complete audit record including the IP address of the TOE, the event details, and the time the event occurred. The time stamp is provided by the TOE appliance hardware.

1.4.1.2.2 Cryptographic support

The TOE contains CAVP-tested cryptographic implementations that provide key management, random bit generation, encryption/decryption, digital signature and secure hashing and key-hashing features in support of higher level cryptographic protocols including HTTPS, NTP, SSH and TLS.

1.4.1.2.3 Identification and authentication

The TOE authenticates administrative users. In order for an administrative user to access the TOE, a user account including a user name and password must be created for the user, and an administrative role must be assigned. The TOE performs the validation of the login credentials. The TOE also performs extensive X.509v3 certificate validation checks on certificates it receives as identification and authentication material.

1.4.1.2.4 Security management

The TOE also provides a Web UI (protected by HTTPS) and Command Line Interface (protected by SSH) to configure the TOE. Security management commands are limited to authorized users (i.e., administrators) and available only after they have provided acceptable user identification and authentication data to the TOE. The security management functions are controlled through the use of privileges associated with roles that can be assigned to TOE users. Among the available privileges, only the Authorized Administrator role can actually manage the security policies provided by the TOE and the TOE offers a complete set of functions to facilitate effective management.

1.4.1.2.5 Protection of the TSF

The TOE implements a number of features design to protect itself to ensure the reliability and integrity of its security features.

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. It also provides its own timing mechanism to ensure that reliable time information is available (e.g., for log accountability) and can obtain time from external time sources using NTP.

The TOE performs self-tests and integrity checks on TOE executables during system start-up as well as periodically during normal operation. The TOE also includes mechanisms (i.e., verification of the digital signature of each new image) so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE.

1.4.1.2.6 TOE access

The TOE can be configured to display a warning banner when an administrator establishes an interactive session and subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session (local or remote) will be terminated.

1.4.1.2.7 Trusted path/channels

The TOE protects interactive communication with administrators using SSHv2 for CLI access, ensuring both integrity and disclosure protection. The TOE also provides a Web UI API interface for security management that is protected with HTTPS/TLS. If the negotiation of an encrypted session (either SSH or TLS) fails or if the user does not have authorization for remote administration, an attempted connection is not be established.

The TOE protects communication with network peers, such as an NTP server, an audit server, VVoIP endpoints, ESC devices for trunking, and a VVoIP conferencing system using TLS connections to prevent unintended disclosure or modification of data.

1.4.2 TOE Documentation

BlackBerry Ltd offers documentation that describes the installation process for the TOE as well as guidance for subsequent use and administration of the applicable security features of the TOE. The following list of documents were examined as part of the evaluation.

SecuGATE Common Criteria Configuration Guide, SecuSUITE for Government 4.0, Document Version 1.3

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
  - Part 3 Conformant
- Package Claims:
  - collaborative Protection Profile for Network Devices, Version 2.1, 24 September 2018

    with the following technical decisions:

    - TD0453
    - TD0452
    - TD0451
    - TD0450
    - TD0449
    - TD0448
    - TD0447
    - TD0425
    - TD0424
    - TD0423
    - TD0412
    - TD0410
    - TD0409
    - TD0408
    - TD0407
    - TD0402
    - TD0401
    - TD0400
    - TD0399
    - TD0398
    - TD0397
    - TD0396
    - TD0395

2.1 Conformance Rationale

The ST conforms to the NDcPP21. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDcPP21 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP21 offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP21 should be consulted if there is interest in that material.

In general, the NDcPP21 has defined Security Objectives appropriate for a dedicated network appliance providing an enterprise session controller capability and as such are applicable to the SecuGATE SIP Server TOE.

3.1 Security Objectives for the Operational Environment

OE.ADMIN_CREDENTIALS_SECURE The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

OE.COMPONENTS_RUNNING (applies to distributed TOEs only) For distributed TOEs the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.

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.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

OE.RESIDUAL_INFORMATION The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

OE.UPDATES The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP21. The NDcPP21 defines the following extended requirements and since they are not redefined in this ST the NDcPP21 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP21:FCS_HTTPS_EXT.1: HTTPS Protocol
- NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP21:FCS_SSHS_EXT.1: SSH Server Protocol
- NDcPP21:FCS_TLSC_EXT.2: TLS Client Protocol with authentication
- NDcPP21:FCS_TLSS_EXT.1: TLS Server Protocol
- NDcPP21:FCS_TLSS_EXT.2: TLS Server Protocol with mutual authentication
- NDcPP21:FIA_PMG_EXT.1: Password Management
- NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests
- NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
- NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
- NDcPP21:FPT_TST_EXT.1: TSF testing
- NDcPP21:FPT_TUD_EXT.1: Trusted update
- NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the NDcPP21. The refinements and operations already performed in the NDcPP21 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP21 and any residual operations have been completed herein. Of particular note, the NDcPP21 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDcPP21 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the NDcPP21 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The NDcPP21 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by SecuGATE SIP Server TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP21:FAU_GEN.1: Audit Data Generation
	NDcPP21:FAU_GEN.2: User identity association
	NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP21:FCS_CKM.1: Cryptographic Key Generation
	NDcPP21:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP21:FCS_CKM.4: Cryptographic Key Destruction
	NDcPP21:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	NDcPP21:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP21:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	NDcPP21:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP21:FCS_HTTPS_EXT.1: HTTPS Protocol
	NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP21:FCS_SSHS_EXT.1: SSH Server Protocol
	NDcPP21:FCS_TLSC_EXT.2: TLS Client Protocol with authentication
	NDcPP21:FCS_TLSS_EXT.1: TLS Server Protocol
	NDcPP21:FCS_TLSS_EXT.2: TLS Server Protocol with mutual authentication
FIA: Identification and authentication	NDcPP21:FIA_AFL.1: Authentication Failure Management
	NDcPP21:FIA_PMG_EXT.1: Password Management
	NDcPP21:FIA_UAU.7: Protected Authentication Feedback
	NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	NDcPP21:FMT_MOF.1/ManualUpdate: Management of security functions behaviour

	NDcPP21:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP21:FMT_MTD.1/CryptoKeys: Management of TSF Data
	NDcPP21:FMT_SMF.1: Specification of Management Functions
	NDcPP21:FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
	NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
	NDcPP21:FPT_TST_EXT.1: TSF testing
	NDcPP21:FPT_TUD_EXT.1: Trusted update
FTA: TOE access	NDcPP21:FTA_SSL.3: TSF-initiated Termination
	NDcPP21:FTA_SSL.4: User-initiated Termination
	NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP21:FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	NDcPP21:FTP_ITC.1: Inter-TSF trusted channel
	NDcPP21:FTP_TRP.1/Admin: Trusted Path

Table 5-1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP21:FAU_GEN.1)

NDcPP21:FAU_GEN.1.1

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

a) Start-up and shut-down of the audit functions;

b) All auditable events for the not specified level of audit; and

c) All administrative actions comprising:

- Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [***no other actions***];

d) Specifically defined auditable events listed in **Table 5-2**.

Table 5-2 Audit Events

Requirement	Auditable Events	Additional Content
NDcPP21:FAU_GEN.1	None	None
NDcPP21:FAU_GEN.2	None	None
NDcPP21:FAU_STG_EXT.1	None	None
NDcPP21:FCS_CKM.1	None	None
NDcPP21:FCS_CKM.2	None	None
NDcPP21:FCS_CKM.4	None	None
NDcPP21:FCS_COP.1/DataEncryption	None	None
NDcPP21:FCS_COP.1/Hash	None	None
NDcPP21:FCS_COP.1/KeyedHash	None	None
NDcPP21:FCS_COP.1/SigGen	None	None

NDcPP21:FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
NDcPP21:FCS_RBG_EXT.1		
NDcPP21:FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
NDcPP21:FCS_TLSC_EXT.2	Failure to establish a TLS Session.	Reason for failure.
NDcPP21:FCS_TLSS_EXT.1	Failure to establish a TLS Session.	Reason for failure.
NDcPP21:FCS_TLSS_EXT.2	Failure to establish a TLS Session.	Reason for failure.
NDcPP21:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_PMG_EXT.1	None	None
NDcPP21:FIA_UAU.7	None	None
NDcPP21:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate.	Reason for failure.
NDcPP21:FIA_X509_EXT.2	None	None
NDcPP21:FIA_X509_EXT.3	None	None
NDcPP21:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	
NDcPP21:FMT_MTD.1/CoreData	All management activities of TSF data.	
NDcPP21:FMT_SMF.1	None	None
NDcPP21:FMT_SMR.2	None	None
NDcPP21:FPT_APW_EXT.1	None	None
NDcPP21:FPT_SKP_EXT.1	None	None
NDcPP21:FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
NDcPP21:FPT_TST_EXT.1	None	None
NDcPP21:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	
NDcPP21:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	
NDcPP21:FTA_SSL.4	The termination of an interactive session.	
NDcPP21:FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a	

	local session by the session locking mechanism.	
NDcPP21:FTA_TAB.1	None	None
NDcPP21: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.
NDcPP21:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	

NDcPP21:FAU_GEN.1.2

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

a) Date and time of the event, type of event, subject identity, 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 cPP/ST, information specified in column three of **Table 5-2**.

5.1.1.2 User identity association (NDcPP21:FAU_GEN.2)

NDcPP21:FAU_GEN.2.1

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

5.1.1.3 Protected Audit Event Storage (NDcPP21:FAU_STG_EXT.1)

NDcPP21:FAU_STG_EXT.1.1

The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

NDcPP21:FAU_STG_EXT.1.2

The TSF shall be able to store generated audit data on the TOE itself. [***TOE shall consist of a single standalone component that stores audit data locally***].

NDcPP21:FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [delete the oldest audit log file]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (NDcPP21:FCS_CKM.1)

NDcPP21:FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic 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] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4***].

5.1.2.2 Cryptographic Key Establishment (NDcPP21:FCS_CKM.2)

NDcPP21:FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography'].***

5.1.2.3 Cryptographic Key Destruction (NDcPP21:FCS_CKM.4)

NDcPP21:FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- For plaintext keys in volatile storage, the destruction shall be executed by a [***single overwrite consisting of [zeroes]***];
- For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [***logically addresses the storage location of the key and performs a [[Single]-pass] overwrite consisting of [zeroes]***]

that meets the following: No Standard.

5.1.2.4 Cryptographic Operation (AES Data Encryption/Decryption) (NDcPP21:FCS_COP.1/DataEncryption)

NDcPP21:FCS_COP.1.1/DataEncryption

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, GCM***] mode and cryptographic key sizes [***128, 256 bits***] that meet the following: AES as specified in ISO 18033-3, [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***].

5.1.2.5 Cryptographic Operation (Hash Algorithm) (NDcPP21:FCS_COP.1/Hash)

NDcPP21:FCS_COP.1.1/Hash

The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-384, SHA-512***] and message digest sizes [***160, 256, 384, 512***] that meet the following: ISO/IEC 10118-3:2004.

5.1.2.6 Cryptographic Operation (Keyed Hash Algorithm) (NDcPP21:FCS_COP.1/KeyedHash)

NDcPP21:FCS_COP.1.1/KeyedHash

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384***] and cryptographic key sizes [***160, 256, 384***] and message digest sizes [***160, 256, 384***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.7 Cryptographic Operation (Signature Generation and Verification) (NDcPP21:FCS_COP.1/SigGen)

NDcPP21:FCS_COP.1.1/SigGen

The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

- ***RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits],***
- ***Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 or 384 bits]***]

that meet the following: [

- ***For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***
- ***For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-256, P-384]; ISO/IEC 14888-3, Section 6.4***].

5.1.2.8 HTTPS Protocol (NDcPP21:FCS_HTTPS_EXT.1)

NDcPP21:FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC 2818.

NDcPP21:FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

NDcPP21:FCS_HTTPS_EXT.1.3
If a peer certificate is presented, the TSF shall [***not require client authentication***] if the peer certificate is deemed invalid.

5.1.2.9 NTP Protocol (NDcPP21:FCS_NTP_EXT.1)

NDcPP21:FCS_NTP_EXT.1.1
The TSF shall use only the following NTP version(s) [**NTP v4 (RFC 5905)**].

NDcPP21:FCS_NTP_EXT.1.2
The TSF shall update its system time using [**Authentication using [SHA256] as the message digest algorithm(s)**].

NDcPP21:FCS_NTP_EXT.1.3
The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.

NDcPP21:FCS_NTP_EXT.1.4
The TSF shall support configuration of at least three (3) NTP time sources.

5.1.2.10 Random Bit Generation (NDcPP21:FCS_RBG_EXT.1)

NDcPP21:FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].

NDcPP21:FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[1] hardware-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.11 SSH Server Protocol (NDcPP21:FCS_SSHS_EXT.1)

NDcPP21:FCS_SSHS_EXT.1.1
The TSF shall implement the SSH protocol that complies with RFC(s) [***4251, 4252, 4253, 4254, 5656, 6668***]. (TD0398 applied)

NDcPP21:FCS_SSHS_EXT.1.2
The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [***password-based***].

NDcPP21:FCS_SSHS_EXT.1.3
The TSF shall ensure that, as described in RFC 4253, packets greater than [**256k**] bytes in an SSH transport connection are dropped.

NDcPP21:FCS_SSHS_EXT.1.4
The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [***aes128-cbc, aes256-cbc***].

NDcPP21:FCS_SSHS_EXT.1.5
The TSF shall ensure that the SSH public-key based authentication implementation uses [***ecdsa-sha2-nistp256***] as its public key algorithm(s) and rejects all other public key algorithms. (TD0424 applied)

NDcPP21:FCS_SSHS_EXT.1.6
The TSF shall ensure that the SSH transport implementation uses [***hmac-sha2-256***] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

NDcPP21:FCS_SSHS_EXT.1.7
The TSF shall ensure that [***ecdh-sha2-nistp256***] and [***no other methods***] are the only allowed key exchange methods used for the SSH protocol.

NDcPP21:FCS_SSHS_EXT.1.8
The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

5.1.2.12 TLS Client Protocol with authentication (NDcPP21:FCS_TLSC_EXT.2)

NDcPP21:FCS_TLSC_EXT.2.1
The TSF shall implement [***TLS 1.2 (RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [
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 RFC5289].

NDcPP21:FCS_TLSC_EXT.2.2
The TSF shall verify that the presented identifiers of the following types [identifiers defined in RFC 6125, IPv4 address in SAN] are matched to reference identifiers. (TD0452 applied).

NDcPP21:FCS_TLSC_EXT.2.3
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [***not implement any administrator override mechanism;***].

NDcPP21:FCS_TLSC_EXT.2.4
The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1] and no other curves***] in the Client Hello.

NDcPP21:FCS_TLSC_EXT.2.5
The TSF shall support mutual authentication using X.509v3 certificates.

5.1.2.13 TLS Client Protocol with authentication (NDcPP21:FCS_TLSC_EXT.2/ExternalCA)

NDcPP21:FCS_TLSC_EXT.2/ExternalCA.1
The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
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_256_GCM_SHA384 as defined in RFC5289].

NDcPP21:FCS_TLSC_EXT.2.2
The TSF shall verify that the presented identifiers of the following types [***identifiers defined in RFC 6125, IPv4 address in SAN***] are matched to reference identifiers. (TD0452 applied).

NDcPP21:FCS_TLSC_EXT.2/ExternalCA.3
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [***not implement any administrator override mechanism;***].

NDcPP21:FCS_TLSC_EXT.2/ExternalCA.4
The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1] and no other curves***] in the Client Hello.

NDcPP21:FCS_TLSC_EXT.2/ExternalCA.5
The TSF shall support mutual authentication using X.509v3 certificates.

5.1.2.14 TLS Server Protocol (NDcPP21:FCS_TLSS_EXT.1)

NDcPP21:FCS_TLSS_EXT.1.1
The TSF shall implement [***TLS 1.2 (RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [
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 RFC5289].

NDcPP21: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***].

NDcPP21:FCS_TLSS_EXT.1.3
The TSF shall [***generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1] and no other curves***].

Application Note: This requirement applies to TLS interfaces offered by the TOE for remote administration over HTTPS and initial enrollment of VVoIP devices with Secure Client Authentication (i.e., SCA0).

5.1.2.15 TLS Server Protocol with mutual authentication (NDcPP21:FCS_TLSS_EXT.2)

NDcPP21:FCS_TLSS_EXT.2.1
The TSF shall implement [***TLS 1.2 (RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [
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 (Not Supported for SIP Calling),
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC5289 (Not Supported for SIP Calling)].

NDcPP21: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***].

NDcPP21:FCS_TLSS_EXT.2.3
The TSF shall [***generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1] and no other curves***].

NDcPP21:FCS_TLSS_EXT.2.4
The TSF shall support mutual authentication of TLS clients using X.509v3 certificates.

NDcPP21:FCS_TLSS_EXT.2.5
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also [***not implement any administrator override mechanism***].

NDcPP21:FCS_TLSS_EXT.2.6
The TSF shall not establish a trusted channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the client.

Application Note: This requirement applies to TLS interfaces offered by the TOE for communication with telecommunication devices such as an ESCor PBX, as well as for communication with VVoIP endpoints for SIP Calling or Secure Client Authentication services (i.e., SCA1).

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (NDcPP21:FIA_AFL.1)

NDcPP21:FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [***3 to 7***] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password. (TD0408 applied).

NDcPP21:FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until [an authorized administrator unlocks the locked user account] is taken by an Administrator, prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed***]. (TD0408 applied).

5.1.3.2 Password Management (NDcPP21:FIA_PMG_EXT.1)

NDcPP21:FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [***'!', '@', '#', '$', '%', '^', '&', '*', '(', ')'***];

b) Minimum password length shall be configurable to [***between 8***] and [***32***] characters.

5.1.3.3 Protected Authentication Feedback (NDcPP21:FIA_UAU.7)

NDcPP21:FIA_UAU.7.1

The TSF shall provide only obscured feedback to the administrative user while the authentication is in progress at the local console.

5.1.3.4 Password-based Authentication Mechanism (NDcPP21:FIA_UAU_EXT.2)

NDcPP21:FIA_UAU_EXT.2.1

The TSF shall provide a local [***password-based, SSH public key-based***] authentication mechanism to perform local administrative user authentication. (TD0408 applied)

5.1.3.5 User Identification and Authentication (NDcPP21:FIA_UIA_EXT.1)

NDcPP21:FIA_UIA_EXT.1.1

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

- Display the warning banner in accordance with FTA_TAB.1;
- [***no other actions***].

NDcPP21:FIA_UIA_EXT.1.2

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

5.1.3.6 X.509 Certificate Validation (NDcPP21:FIA_X509_EXT.1/Rev)

NDcPP21:FIA_X509_EXT.1.1/Rev

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

- RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates.

- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [***Certificate Revocation List (CRL) as specified in RFC 5759 Section 5***]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
  - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
  - 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 extendedKeyUsage field.
  - 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 extendedKeyUsage field.

NDcPP21:FIA_X509_EXT.1.2/Rev
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.

5.1.3.7 X.509 Certificate Authentication (NDcPP21:FIA_X509_EXT.2)

NDcPP21:FIA_X509_EXT.2.1
The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [***TLS***], and [***no additional uses***].

NDcPP21:FIA_X509_EXT.2.2
When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [***accept the certificate***].

5.1.3.8 X.509 Certificate Requests (NDcPP21:FIA_X509_EXT.3)

NDcPP21:FIA_X509_EXT.3.1
The TSF shall generate a Certification Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name***].

NDcPP21:FIA_X509_EXT.3.2
The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behaviour (NDcPP21:FMT_MOF.1/ManualUpdate)

NDcPP21:FMT_MOF.1.1/ManualUpdate
The TSF shall restrict the ability to enable the functions to perform manual update to Security Administrators.

5.1.4.2 Management of TSF Data (NDcPP21:FMT_MTD.1/CoreData)

NDcPP21:FMT_MTD.1.1/CoreData
The TSF shall restrict the ability to manage the TSF data to Security Administrators.

5.1.4.3 Management of TSF Data (NDcPP21:FMT_MTD.1/CryptoKeys)

NDcPP21:FMT_MTD.1.1/CryptoKeys
The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

5.1.4.4 Specification of Management Functions (NDcPP21:FMT_SMF.1)

NDcPP21:FMT_SMF.1.1

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

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [***digital signature***] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- [
  - ***Ability to configure audit behaviour;***
  - ***Ability to manage the cryptographic keys;***
  - ***Ability to configure the cryptographic functionality;***
  - ***Ability to set the time which is used for time-stamps;***
  - ***Ability to configure NTP;***
  - ***Ability to configure the reference identifier for the peer;***
  - ***Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;***
  - ***Ability to import X.509v3 certificates to the TOE's trust store***].

5.1.4.5 Restrictions on Security Roles (NDcPP21:FMT_SMR.2)

NDcPP21:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP21:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP21:FMT_SMR.2.3

The TSF shall ensure that the conditions

- The Security Administrator role shall be able to administer the TOE locally;
- The Security Administrator role shall be able to administer the TOE remotely are satisfied.

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Protection of Administrator Passwords (NDcPP21:FPT_APW_EXT.1)

NDcPP21:FPT_APW_EXT.1.1

The TSF shall store passwords in non-plaintext form.

NDcPP21:FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext passwords.

5.1.5.2 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (NDcPP21:FPT_SKP_EXT.1)

NDcPP21:FPT_SKP_EXT.1.1

The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.5.3 Reliable Time Stamps (NDcPP21:FPT_STM_EXT.1)

NDcPP21:FPT_STM_EXT.1.1

The TSF shall be able to provide reliable time stamps for its own use.

NDcPP21:FPT_STM_EXT.1.2

The TSF shall [***allow the Security Administrator to set the time, synchronise time with an NTP server***].

5.1.5.4 TSF testing (NDcPP21:FPT_TST_EXT.1)

NDcPP21:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)***] to demonstrate the correct operation of the TSF: [***Cryptographic Known Answer Test (KAT) and TOE software integrity checks***].

5.1.5.5 Trusted update (NDcPP21:FPT_TUD_EXT.1)

NDcPP21:FPT_TUD_EXT.1.1
The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [***no other TOE firmware/software version***].

NDcPP21:FPT_TUD_EXT.1.2
The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [***no other update mechanism***].

NDcPP21:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.6 TOE access (FTA)

5.1.6.1 TSF-initiated Termination (NDcPP21:FTA_SSL.3)

NDcPP21:FTA_SSL.3.1
The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.6.2 User-initiated Termination (NDcPP21:FTA_SSL.4)

NDcPP21:FTA_SSL.4.1
The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.6.3 TSF-initiated Session Locking (NDcPP21:FTA_SSL_EXT.1)

NDcPP21:FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [***terminate the session***] after a Security Administrator-specified time period of inactivity.

5.1.6.4 Default TOE Access Banners (NDcPP21:FTA_TAB.1)

NDcPP21:FTA_TAB.1.1
Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Inter-TSF Trusted Channel (NDcPP21:FTP_ITC.1)

NDcPP21:FTP_ITC.1.1
The TSF shall be capable of using **[TLS]** to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [***/push server, ESC endpoint, NTP server, certificate authority, VVoIP endpoint/***] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

NDcPP21:FTP_ITC.1.2
The TSF shall permit the TSF or the authorized IT entities to initiate communication via the trusted channel.

NDcPP21:FTP_ITC.1.3
The TSF shall initiate communication via the trusted channel for [audit server, push server, certificate authority, telecommunications device].

5.1.7.2 Trusted Path (NDcPP21:FTP_TRP.1/Admin)

NDcPP21:FTP_TRP.1.1/Admin
The TSF shall be capable of using [***SSH, HTTPS***] to provide a communication path between itself and authorized remote Administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and provides detection of modification of the channel data.

NDcPP21:FTP_TRP.1.2/Admin
The TSF shall permit remote Administrators to initiate communication via the trusted path.

NDcPP21:FTP_TRP.1.3/Admin
The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing â€“ Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 5-3 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

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.

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-non-interfering.

ADV_FSP.1.4c

The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e

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

ADV_FSP.1.2e

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d

The developer shall provide operational user guidance.

AGD_OPE.1.1c

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

AGD_OPE.1.2c

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

AGD_OPE.1.3c

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

AGD_OPE.1.4c

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

AGD_OPE.1.5c

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

AGD_OPE.1.6c

The operational user guidance shall, for each user role, describe the security measures to be followed in order to 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.

AGD_OPE.1.1e

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

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c

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

AGD_PRE.1.2c

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

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

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

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d
The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c
The TOE shall be labelled with its unique reference.

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

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d
The developer shall provide a configuration list for the TOE.

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.

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

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing â€“ Conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

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.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

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. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The TOE is a single standalone device that stores its audit data locally and can send audit data to an external audit server in real time. The TOE uses the Linux Audit System (auditd) to log the events and information identified by the 'Audit Events' table in section 5.1.1. Audits of administrative actions affecting cryptographic keys (i.e., generation, import, modification or deletion) include a keyword indicating the service using the key (e.g., est_tls, syslog_tls, pabx_sip_tls) which correspond with the naming of the service in the TOE Web UI interface for certificate assignment.

The Linux auditd daemon process receives audit data from applications and the kernel. The daemon runs as a root process and writes audit data to an audit log file on the local machine. Only the root and the security administrator have read and write access to the locally saved audit log. Audit logs are saved in a dedicated disk partition. The default maximum size of the audit logs is 1 GB.

Linux auditd can be configured by the security administrator to write audit logs additionally to the local rsyslogd that can forward the logs to an external syslog server via TLS.

The TOE generates system log records indicating the current IP connections, NTP status, CPU usage, memory usage, disk and file storage capacity, and audit storage capacity of the TOE. This data is periodically written into the TOE system log that can be forwarded to an external syslog. The TOE generate these system log recordsevery 60 seconds. The TOE connects "calls" between VVoIP endpoints in accordance with the policies defined by FDP_IFC.1 and FDP_IFF.1. The TOE also keeps a record of various call details for each connection it processes (not call contents). A call detail record for a communication attempt between two endpoints is known as a Call Detail Record (CDR). A CDR is recorded internally and contains the following meta-data:

- calling party number (i.e. call originator),
- called party number (i.e. call receiver or terminating number),
- unique transaction sequence number,
- call disposition (e.g. call connected, call terminated, call transferred),
- call type (e.g. voice only, voice and video, text),
- call start time,
- call end time,
- call duration,
- unique identifier of the TOE,
- call routing into TOE,
- call routing out of TOE,
- time zone, and
- call release cause, if applicable (i.e. reason for termination of call).

CDRs are stored in the system internal database that does not provide direct access to external administrators or users nor to IT entities. CDRs are only accessible as download via administrative interface functions.

The TOE implements an internal clock provided by the OS to keep reliable time. Linux auditd, rsyslogd and TOE Call detail mechanism make use of the internal clock for timestamps in audit records, system log records and CDR.

The TOE allows administrators to use TOE interfaces to view all records that have been sent to the system log.

The Security audit function satisfies the following security functional requirements:

- NDcPP21:FAU_GEN.1: The TOE uses the Linux Audit System (auditd) to log the events and information identified by the 'Audit Events' table in section 5.1.1. The TOE also audits administrator actions that are performed through TOE interfaces. The TOE includes in each audit records a date/time stamp, an event type, an identifier of the subject responsible for the activity, the outcome of the activity, and other data specific to each event type as defined by the 'Audit Events' table in section 5.1.1. Audit records stored by 'auditd' are transferred to a network peer using rsyslogd over a TLS protected connection.
- NDcPP21:FAU_GEN.2: The TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- NDcPP21:FAU_STG_EXT.1: The TOE stores audit data locally and uses rsyslogd to forward log data to an external syslog server via TLS. The TOE uses log rotation with 5 log files where the oldest is deleted when the sixth log file is started. The log rotation is triggered when the current audit log file exceeds 6 Mbyte. Thus the TOE can store 30 Mbytes of data locally.

6.2 Cryptographic support

The TOE is a network device that runs on a physical platform that is the SUPERMICRO SuperServer with an Intel Xeon E3-1240, Xeon E3-1515 or Xeon Gold 5218 processor supporting the RDRAND feature. The TOE utilizes the OpenSSL 1.0.2 FIPS Object Module Version 2.0.16 as the cryptographic provider for all cryptographic operations. These functions have been CAVP tested and received the certificates identified by Table 6-1 Cryptographic Functions.

Requirements	Functions	Standards	Cert
FCS_CKM.1	ECC key generation schemes using 'NIST curves' P-256, P-384	FIPS Pub 186-4	C1277
	RSA Key Generation	FIPS Pub 186-4	C1277
FCS_CKM.2	Elliptic curve-based key establishment schemes	NIST SP 800-56A	C1277
FCS_COP.1/DataEncryption	AES CBC (128 and 256 bits)	FIPS Pub 197	C1277
	AES GCM (128 and 256 bits)	NIST SP 800-38A	C1277
FCS_COP.1 / Hash	SHA-1, SHA-256, SHA-384, SHA-512	FIPS Pub 180-3	C1277
FCS_COP.1/KeyedHash	HMAC-SHA-1 HMAC-SHA-256, HMAC-SHA-384	FIPS Pub 198-1 FIPS Pub 180-3	C1277
FCS_COP.1/SigGen	RSA Digital Signature Algorithm (rDSA) (modulus 2048)	FIPS Pub 186-4	C1277
	Elliptic Curve Digital Signature Algorithm (ECDSA) with an elliptical curve size of 256 or 384 bits	FIPS Pub 186-4	C1277
FCS_RBG_EXT.1	AES-256 CTR_DRBG	FIPS SP 800-90A	C1277

Table 6-1 Cryptographic Functions

The TOE uses hashing for the following functions:

- SHA-1
  - TLS client authentication with RSA
  - Certificates: subjectKeyIdentifier
- SHA-256
  - TLS server authentication with ECDHE-RSA
  - TLS pseudorandom function (PRF) with AES128-GCM
  - TLS PRF with AES128-CBC
  - SSH server and client authentication with ecdsa-sha2
  - SSH key exchange with ecdh-sha2
  - Digest Access Authentication
  - AIDE file integrity
- SHA-384

  - Certificate signature
  - TLS PRF with AES256-GCM
- SHA-512
  - Admin password hashing
  - SSH password hashing

The TOE uses key-hash message authentication with TLS and SSH (HMAC-SHA-256).

HMAC	Key Length	Block Size	Output Length
SHA-1	160 bit	512 bit	160 bit
SHA-256	256 bit	512 bit	256 bit
SHA-384	384 bit	1024 bit	384 bit

Table 6-2 Keyed Hashing

The TOE supports SSHv2 using AES-CBC with 128 or 256 bit ciphers, in conjunction with HMAC-SHA2-256 for message integrity. The TOE supports the ecdsa-sha2-nistp256 public key authentication methods. The TOE offers the ecdh-sha2-nistp256 as the only supported key exchange method. No optional aspects of the protocol are supported.

The TOE allows users to perform SSHv2 authentication using password based authentication and allows users to upload a public key for SSHv2 public key client authentication. The TOE's SSHv2 implementation limits SSH packets to a size of 256k bytes.

Whenever the timeout period or authentication retry limit is reached, the TOE closes the applicable TCP connection and releases the SSH session resources. As SSH packets are being received, the TOE uses a buffer to build all packet information. Once complete, the packet is checked to ensure it can be appropriately decrypted. However, if it is not complete when the buffer becomes full (256k bytes) the packet will be dropped and the connection terminated. There is a TOE initiated rekey before 1 hour or before 1GB whichever comes first. The TOE implements default values roughly half of these values. The TOE does not offer a method for these default rekey values to be modified by the administrator.

The TOE supports the SSHv2 (compliant with RFCs 4251, 4252, 4253, 4254, 5656, 6668) and TLS v1.2 (RFC 5246) secure communication protocols.

The TOE provides support for TLS ciphersuites and versions for various services as shown in the following table.

Table 6-3 TLS Support by Service

Service	**TOE Role**	**TLS Version**	**TLS Ciphersuites**				
			*TLS_ECDHE_**ECDSA**_WITH_**AES_128**_**GCM**_SHA256*	*TLS_ECDHE_**ECDSA**_WITH_**AES_256**_**GCM**_SHA384*	*TLS_ECDHE_**RSA**_WITH_**AES_128**_**GCM**_SHA256*	*TLS_ECDHE_**RSA**_WITH_**AES_256**_**GCM**_SHA384*	*TLS_ECDHE_**RSA**_WITH_**AES_256**_**CBC**_**SHA***
Push Server	Client	TLSv1.2	Yes	Yes	Yes	Yes	No
Syslog	Client	TLSv1.2	Yes	Yes	Yes	Yes	No
External CA	Client	TLSv1.2 & TLSv1.1	Yes	Yes	No	Yes	Yes
Web UI	Server	TLSv1.2	Yes	Yes	Yes	Yes	No
Secure Client Authentication (SCA0 and SCA1)	Server	TLSv1.2	Yes	Yes	Yes	Yes	No
SIP Calling	Server	TLSv1.2	Yes	Yes	No	No	No

Trunking[2]	Server	TLSv1.2	Yes	Yes	Yes	Yes	No
	Client	TLSv1.2	Yes	Yes	Yes	Yes	No

- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC5289
- 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_256_CBC_SHA as defined in RFC 4492
- TLS 1.2 as defined in RFC 5246
- TLS 1.1 as defined in RFC 4346

Server and client (TOE) establish a shared TLS premaster secret with the TLS key exchange. All key exchange methods use the same algorithm then to convert the premaster secret into the master secret.

Together with client random (in ClientHello message) and server random (in ServerHello message), client and server generate session encryption and MAC keys from the master secret with the TLS PRF.

If a TLS server requests client authentication from the TOE with the ClientCertificateRequest message, the TOE answers with ClientCertificate and CertificateVerify messages. The TOE is configured by administrators with an X.509v3 certificate which it presents to a TLS server requesting authentication.

The TOE supports TLS v1.2 with the ciphersuites listed above when acting as a TLS Server to administrative users and during SIP client enrollment. The SIP client enrollment and administrator Web UI interfaces do not require mutual authentication using TLS. Authentication on these interfaces uses other mechanisms. The TOE rejects all connection attempts using SSL and older TLS versions (1.0 and 1.1). The key agreement parameters of the server key exchange message are specified in the RFC 5246 (section 7.4.3) for TLSv1.2.

The TOE uses TLS (OpenSSL) for communication with SIP clients and registered SCA devices[3] when it is acting as a TLS server. The TOE TLS server requests client authentication by sending the ClientCertificateRequest message, and the client is expected to answer with ClientCertificate and CertificateVerify messages. This TLS authentication is certificate-based with the TOE presenting its X.509v3 certificate and expecting the client to present a valid X.509v3 certificate. The TOE will match the presented identifiers from certificates received during TLS negotiation against configured reference identifiers. Reference identifiers within certificate are either a Distinguished Name (DN) or Subject Alternate Name (SAN). Only the Common Name field within the DN is used; and the CN must be an exact match for the entire CN string. The SAN may be matched with a reference identifier that is either an IPv4 address or DNS name. The TOE performs all X.509v3 certificate validation checks as described by FIA_X509_EXT.1 and FIA_X509_EXT.2. If client certificate is determined to not be valid or not from an expected client identity, the TOE does not establish the connection.

The TOE TLS Server performs key agreement using NIST curves secp256r1 and secp384r1. By default, the TOE acting as a TLS client presents the supported elliptic-curve extension in the Client Hello with NIST curves secp256r1 and secp384r1.

The Cryptographic support function satisfies the following security functional requirements:

- NDcPP21:FCS_CKM.1: The TOE OpenSSL module performs key generation in accordance with the RSA (2048-bit) and ECDSA (P-256, P-384) schemes. The ECDSA is used in support of TLS and SSH. The TOE acts as both sender and receiver (i.e. depending on the channel) schemes in TLS. The RSA scheme is used to support CSR and device authentication.

[2] Trunking is communication with another ESC or PABX system.
[3] A registered SCA device may obtain services from the TOE's SCA1 interface by presenting the same certificate presented to the TOE SIP interface (e.g., a contacts list).

- NDcPP21:FCS_CKM.2: The TOE OpenSSL module performs key establishment in accordance with ECC / NIST SP 800-56A in support of TLS and SSH. In accordance with NIST SP 800-56B, the TOE does not reveal specific error details but raises generic errors during TLS handshake.
- NDcPP21:FCS_CKM.4: None of the TOE's symmetric keys, pre-shared keys, or private keys are stored in plaintext form.
- NDcPP21:FCS_COP.1/DataEncryption: The TOE OpenSSL module performs AES encryption and decryption in CBC and GCM mode with key sizes of 256-bits.
- NDcPP21:FCS_COP.1/Hash: The TOE OpenSSL module performs SHA-1, SHA-256, SHA-384 and SHA-512 cryptographic hashing in support of the functions listed above. The TOE OpenSSL module performs SHA-1, SHA-256, SHA-384 and SHA-512 cryptographic hashing with message digest sizes of 160-bit, 256-bit, 384-bit or 512-bit, for the corresponding hashing algorithm.
- NDcPP21:FCS_COP.1/KeyedHash: The TOE OpenSSL module performs HMAC-SHA-1, HMAC-SHA-256 and HMAC-SHA-384 keyed-hash message authentication with key sizes of 160-bit, 256-bit and 384-bit, with message digest sizes of 160-bit, 256-bit and 384-bit.
- NDcPP21:FCS_COP.1/SigGen: The TOE OpenSSL module performs RSA and ECDSA cryptographic signature services (generation and verification).
- NDcPP21:FCS_HTTPS_EXT.1: The TOE provides a Web User Interface (Web UI) for remote administration and fully supports RFC 2818. The TOE acts as an HTTPS server and waits for client connections on TCP port 443. The TOE's HTTPS server supports TLS version 1.2 only and will deny connection requests from TLS clients with lower versions. The TOE ignores any certificate provided during the TLS negotiations and does not authenticate the client using an X.509v3 certificate. The TOE sends all HTTP data as encrypted TLS "application data".
- NDcPP21:FCS_NTP_EXT.1: The TOE supports NTPv4 as defined by RFC 5905. The communication channel between the TOE and NTP time source is authenticated using a SHA-256 message digest. The TOE can be configured to obtain time from at multiple sources (up to 4) and does not accept time updates from broadcast or multicast addresses.
- NDcPP21:FCS_RBG_EXT.1: The TOE leverages Intel's RdRand TRNG to seed the OpenSSL CTR_DRBG (AES).
- NDcPP21:FCS_SSHS_EXT.1: The TOE supports SSHv2 as described above.
- NDcPP21:FCS_TLSC_EXT.2/NDcPP21:FCS_TLSC_EXT.2/ExternalCA: Depending upon service, the TOE supports TLS as shown in **Table 5-1** above. When acting as a TLS client the TOE can authenticate itself using an X.509v3 certificate to a TLS server requesting authentication. The TOE does not support certificate pinning. The TOE does support validation of certificates containing IP addresses and wildcards in DNS names. If the certificate presented by a TLS server cannot be validated, the TOE does not establish the connection.
- NDcPP21:FCS_TLSS_EXT.1: The TOE supports TLS v1.2 with the ciphersuites listed above when acting as a TLS Server to administrative users and during SIP client enrollment (SCA0). The SIP client enrollment and administrator WebUI interfaces do not require mutual authentication using TLS. Authentication on these interfaces uses other mechanisms. The TOE rejects all connection attempts using SSL and older TLS versions (1.0 and 1.1). The key agreement parameters of the server key exchange message sent by the TOE are specified in the RFC 5246 (section 7.4.3) for TLSv1.2.
- NDcPP21:FCS_TLSS_EXT.2: The TOE supports TLS v1.2 with the ciphersuites listed above when acting as a TLS Server with SIP clients (i.e., SCA1 and SIP Calling) and when acting as a TLS Server with trunking peers. The TOE rejects all connection attempts using SSL and older TLS versions (1.0 and 1.1). The key agreement parameters of the server key exchange message are specified in the RFC 5246 (section 7.4.3) for TLSv1.2.

6.3 Identification and authentication

The TOE supports certificate status checking using a Certificate Revocation List (CRL) as specified in RFC 5759 for X509v3 certificate validation. Revocation checking of a certificate is performed the same for all certificate validation operations. Revocation status checking is performed on leaf and intermediate CA certificates received by the TOE for authentication purposes. If the revocation status of a certificate cannot be verified because a current CRL is unavailable, the certificate will be accepted.

The following login methods are supported for administrative users:

- HTTPS Web UI. Administrators may login with a correct username and password combination.
- Local console. Administrators may login locally with a correct username and password combination.
- SSH. Administrators may login via SSH with either:
    - Correct username and password combination, or
    - Recognized ECDSA public keys

The Identification and authentication function satisfies the following security functional requirements:

- NDcPP21:FIA_AFL.1: The TOE implements a locking feature which prevents an account from being able to successfully authenticate after a configured number of failed login attempts. Failed login attempts at the Web UI or via SSH accumulate until the TOE locks the account, and it cannot be used to login. The TOE maintains the locked status of the account for the lockout period configured by the administrator (a value between 1 and 10080 minutes, with 10 minutes being the default). The administrator can configure the lockout threshold between 3 and 7 failed attempts. The TOE also offers commands via its CLI interface which can be used to unlock an account before the configured lockout period.
- NDcPP21:FIA_PMG_EXT.1: The TOE supports a defined character set for password-based authentication. Minimum password length is configurable by the TOE administrator, however, this minimum value must be between 8 and 32 characters in length (default is 15).
- NDcPP21:FIA_UAU.7: The TOE obscures feedback during password-based authentication.
- NDcPP21:FIA_UAU_EXT.2: The TOE implements password-based authentication for administrators accessing the TOE via the HTTPS Web UI or SSHv2 CLI. The TOE also supports public-key authentication through the SSHv2 interface.
- NDcPP21:FIA_UIA_EXT.1: The TOE requires entities to perform identification and authentication before performing any actions other than displaying a warning banner.
- NDcPP21:FIA_X509_EXT.1/Rev: Certificates are validated as part of the authentication process when they are presented to the TOE and when they are loaded into the TOE. The following fields are verified as appropriate: SAN checks, CN checks, key usages, basic constraints, chain validation, and expiration status. The common name must contain a FQDN, the SAN, if present, can include IP addresses or DNS names. Wildcards in DNS names are allowed in certificates. The TOE performs validation of a certificate including the following checks as appropriate:
    - Verify that the client rejects an otherwise valid server certificate that lacks the Server Authentication purpose in the extendedKeyUsage field.
    - Verify a server certificate that contains a CN that does not match the reference identifier and does not contain the SAN extension is rejected.
    - Verify a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier is rejected.
    - Verify a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches is accepted.
    - Verify the certificate chain is valid.
    - Verify the expiration status of the certificate.
    - Verify the revocation status of the certificate.

- NDcPP21:FIA_X509_EXT.2: The administrators configures a certificate for each service and those certificates are used for all further processing. Certificates are checked and if found not valid are not accepted. If a certificate contains a CRL Distribution Point, and a current version of the identified CRL cannot be obtained, then the TOE will accept the certificate as being not revoked. All other certificate validity checks must pass in order for the certificate to be treated as valid.
- NDcPP21:FIA_X509_EXT.3: The TOE OpenSSL module is able to generate certificate signing requests and validating the CA response. The TOE generates certificate requests with the ability to include values for Common Name, Organization, Organizational Unit, and Country. The TOE validates certificates imported into the TOE configuration during import.

6.4 Security management

The TOE provides administrators with the ability to perform management operation using either a local console or remote administrative connection (i.e., SSH protected CLI and TLS protected Web UI). Using these interfaces the following management operations are available to an administrator to:

- configure the access banner;
- configure the session inactivity time before session termination or locking;
- update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- configure password policy;
- configure the authentication failure parameters for FIA_AFL.1;
- configure audit behaviour;
- manage the cryptographic keys;
- configure the cryptographic functionality;
- set the time which is used for time-stamps;
- configure NTP;
- configure the reference identifier for the peer;
- manage the TOE's trust store and designate X509.v3 certificates as trust anchors; and
- import X.509v3 certificates to the TOE's trust store.

The Security management function satisfies the following security functional requirements:

- NDcPP21:FMT_MOF.1/ManualUpdate: Only the authorized administrator can update the TOE.
- NDcPP21:FMT_MTD.1/CoreData: Security management is restricted to administrators using either the CLI or Web UI. The TOE requires system administrators to be logged in before they are allowed to set the time or configure NTP servers (which can modify time). The TOE restricts manipulation of the TOE certificates and trust store to administrators.
- NDcPP21:FMT_MTD.1/CryptoKeys: The TOE restricts the ability to manage cryptographic keys to the administrator.
- NDcPP21:FMT_SMF.1: The TOE provides administrative interfaces to perform the functions identified above.
- NDcPP21:FMT_SMR.2: The TOE maintains administrative user roles.

6.5 Protection of the TSF

The TOE implements the Advanced Intrusion Detection Environment (AIDE) file and directory integrity checker to confirm the integrity of critical files and directories on start-up. AIDE is configured to perform the following:

- construct a database of TSF critical files
- uses OpenSSL to create a SHA-256 hash of each protected file
- perform an integrity check of protected files at start-up (utilizing OpenSSL to generate hashes for comparison to database)

- if the integrity test fails the TOE will generate an audit event

The TOE incorporates the OpenSSL FIPS Object Module as specified in section 6.2 which runs start-up self-tests to confirm the correct operation of the cryptographic functions of the TOE. OpenSSL performs the following power on self-tests:

- Software integrity – HMAC-SHA1
- HMAC Known Answer Tests (KAT)
- AES KATs
- TDES KATs
- RSA KAT
- DSA KAT
- DRBG KATs
- ECDSA pairwise consistency test
- ECC CDH KAT

Together, these tests ensure that the TOE is operating correctly.

Software updates are made available via an update server hosted within SecuSmart cloud services. Each update is a tar file signed with a private SecuSmart key dedicated for software package signing (ECDSA P-256) – the update package includes this digital signature. The SIP server has the corresponding public key in its filesystem, and only root can access this key. Using this public key the software update function of the TOE verifies the signature of the new update using the OpenSSL module.

When the security administrator starts the software update process, the software update function:

- checks the SecuSmart signature
- unpacks the tar file
- starts the installation script included in the tar file

Installation of the update fails if the digital signature verification fails. In this case an error message is displayed to the administrator, a log event is generated, the update is aborted and the original software remains unchanged. If an update is successful, a message is displayed to the administrator and the new software begins running.

The Protection of the TSF function satisfies the following security functional requirements:

- NDcPP21:FPT_APW_EXT.1: The TOE ensures that plaintext user passwords will not be disclosed even to administrators. Admin passwords are stored locally in a hashed form using SHA-512 hash. No interfaces are offered to administrators to view passwords in cleartext form.
- NDcPP21:FPT_SKP_EXT.1: The TOE stores all private keys in a secure directory that is not readily accessible to administrators. This directory is the /etc/pki directory which is protected with root only access permissions. Security administrators cannot view contents of this directory using commands available through the CLI (and the Web UI does not present the file system abstraction to Web users).
- NDcPP21:FPT_STM_EXT.1: The TOE implements an internal clock provided by the OS to keep reliable time. Time can be set manually by the Administrator. The TOE also allows the administrator to configure the use NTPv4 to set the clock and maintain accurate time. The TOE allows system administrators to set the time used by the TOE. This time is used by the TOE for audit timestamps and certificate validation activities.
- NDcPP21:FPT_TST_EXT.1: The TOE provides self-test functions as described above.
- NDcPP21:FPT_TUD_EXT.1: The TOE provides functions on the Web UI that will display the current running version of the TOE. The TOE performs software updates upon verification of the update using a digital signature within the update and the know SecuSmart update public key stored within the TOE as described above.

6.6 TOE access

The TOE access function satisfies the following security functional requirements:

- NDcPP21:FTA_SSL.3: The TOE terminates SSHv2 sessions after an administrator configured period of inactivity.
- NDcPP21:FTA_SSL.4: TOE administrators are able to terminate an interactive session.
- NDcPP21:FTA_SSL_EXT.1: The TOE terminates console sessions after a configured period of inactivity.
- NDcPP21:FTA_TAB.1: The TOE displays a configurable advisory notice and consent warning message regarding use of the TOE when connecting via Web UI, local console or SSHv2.

6.7 Trusted path/channels

The TOE communicates with network peers using TLS protected communication channels for connections with a VVoIP endpoint, a certificate authority, an NTP server, an audit server, ESC devices for trunking, and a push notification server. The TOE can act as a client to an audit server, certificate authority, NTP server, push server, or ESC device (i.e., SIP server for trunking). The TOE acts as a server for remote administration and SCA registration, as well as when communicating with VVoIP endpoints or other ESC devices for accepting a request for trunking.

The Trusted path/channels function satisfies the following security functional requirements:

- NDcPP21:FTP_ITC.1: In the evaluated configuration, the TOE must be configured to use TLS to ensure that exported all communication channels with network peers are sent only to the configured peer so they are not subject to inappropriate disclosure or modification. The TOE validates the peer and against the TOE configuration using the certificates presented during TLS negotiation.
- NDcPP21:FTP_TRP.1/Admin: TOE administrators can use either the Web UI protected by TLS or a command line interface available through an SSHv2 connection for remote administration. Administrators logging in to the TOE Web UI must negotiate a TLS connection and then provide a valid username and password combination. Administrators do not need to present a certificate during the TLS exchange.