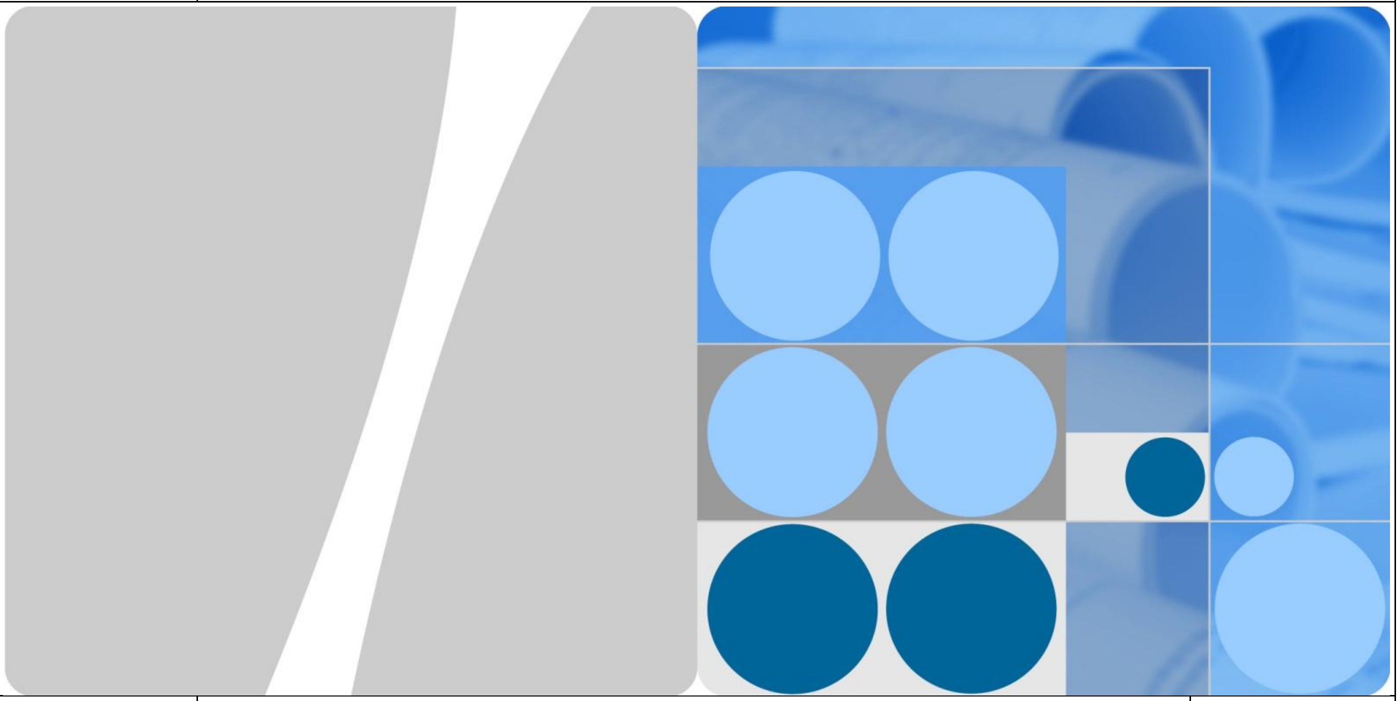


Huawei ATN Series Routers

Security Target

Issue	V2.5
Date	2025-07-14

HUAWEI TECHNOLOGIES CO., LTD.

Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Date	Version	Change Description	Author
2024-06-03	1.0	Initial Draft	hongdongmei
2024-08-05	1.1	update according to evaluator comments	hongdongmei
2024-08-15	1.2	update according to evaluator comments V2.0	hongdongmei
2024-09-20	1.3	update according to evaluator comments	hongdongmei
2025-5-22	1.4	update according to evaluator comments	hongdongmei
2025-6-12	2.0	update according to evaluator comments	hongdongmei
2025-6-18	2.1	update according to evaluator comments	hongdongmei
2025-07-07	2.2	update according to evaluator comments	hongdongmei
2025-07-10	2.3	update according to evaluator comments	hongdongmei
2025-07-11	2.4	update according to evaluator comments	hongdongmei
2025-07-14	2.5	update according to evaluator comments	hongdongmei

Contents

1 Introduction

1.1 ST reference and TOE Reference

Name	Description
ST Title	Security Target of Huawei ATN Series Routers
ST version	2.5
Vendor and ST author	Huawei Technologies Co., Ltd
TOE Name	Huawei ATN Series Routers
TOE Hardware Models	ATN 910C-M,ATN 910C-K and ATN 910D-B
TOE software version	V800R022C00SPC600

1.2 TOE overview

The Huawei ATN Series Routers TOE are used to satisfy the requirements for networks of various scales. They are deployed at the edge of MANs or at access sites that process heavy traffic to implement multi-service access. The TOE includes the hardware models as defined in Table 1-2 in section 1.3.

The TOE is comprised of several security features. as identified below:

(1) Security audit

(2) Cryptographic support

(3) Identification and authentication

(4) Secure Management

(5) Protection of the TSF

(6) TOE access through user authentication

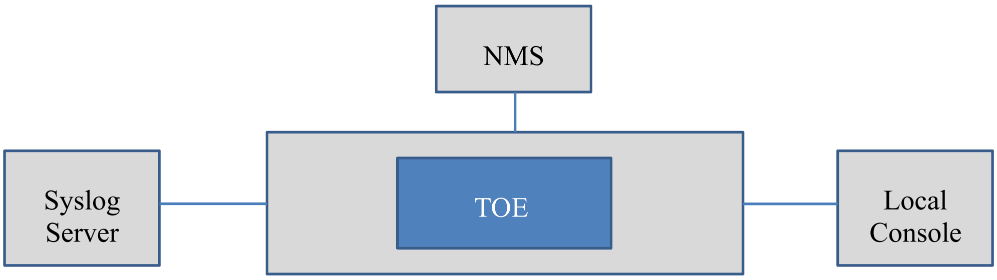
(7) Trusted path and channels for device authentication.

1.2.1 TOE usage

1. The TOE supports username/password, or public-key authentication mode and only users that are authenticated can access the TOE and its command line interface.
2. The TOE is accessed by CLI locally or a Network Management Server (NMS) remotely over SSH so that a secure channel is established to protect the data between TOE and NMS.
3. For secure transmission of audit information between the TOE and the Syslog server a secure TLS channel is used.
4. The TOE supports digital signature verification for software. Each of the software package or patch package released by Huawei includes a unique digital signature. When an NMS distributes the package to router, the TOE will verify the online digital signature before updating. The verification of the digital signature demonstrates the integrity and authenticity of the package. The package is only processed further after successful verification of the digital signature, otherwise the package will be discarded without processing.

The TOE provides security services onto a single and secure device. It supports (in some cases optionally) the following hardware, software, and firmware in its environment when the TOE is configured in Figure 1-1 (NMS: Network Management Server).

Figure 1-1 IT Entities which connect with TOE



1.2.2 TOE type

The TOE type is a network device that is connected to the network and has an infrastructure role within the network.

1.2.3 Non TOE Hardware and Software

The TOE supports the following hardware, software, and firmware components in its operational environment. All of the following environment components are supported by all TOE evaluated configurations.

Table 1-1 IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Network Management Server	YES	This includes any Management workstation with a SSH client installed that is used to establish a protected channel with the TOE.
Local Console	YES	This includes any Console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
Syslog Server	YES	This includes any syslog server to which the TOE would transmit syslog messages.
Open PGP	YES	The Open PGP is used to verify the integrity of software package that is necessary to perform the installation of the TOE.

1.3 TOE description

The TOE is ATN Series Routers comprised of both software and hardware. The TOE scope consists of the software running in the router device. The hardware is comprise of the following: ATN 910C-M,ATN 910C-K and ATN 910D-B.

TSF relevant functions depend on software implementation. Table 1-2 below describes the models that have been claimed within this evaluation.

Table 1-2 Hardware Scope

Hardware	Configuration	Processor	Interface
ATN 910C-K	ATN 910C-K Integrated Chassis,Fixed interfaces.	ARM	Based on TOE's I/O modules
ATN 910C-M	ATN 910C-M Integrated Chassis,Fixed interfaces.	ARM	Based on TOE's I/O modules
ATN 910D-B	ATN 910D-B Integrated Chassis,Fixed interfaces.	ARM	Based on TOE's I/O modules

1.4 Physical scope

This section will define the physical scope (table 1-3) of the Huawei ATN series routers to be evaluated.

Table 1-3 Physical scope

Type	Delivery Item	Version	Date
Hardware	ATN 910C-M,ATN 910C-K and ATN 910D-B The Hardware will be delivered by air, ship, train or automobile	NA	
Software	ATN Router V800R022C00SPC600 Format: ATN 910C-K:ATN910C-910D_V800R022C00SPC600.cc Info: Users can login the HUAWEI support website to download the software packet in accordance to the version of the TOE. Users can verify the software by digital signature(The digital signature is also published on HUAWEI support website) ATN 910C-M:ATN910C-910D_V800R022C00SPC600.cc Info: Users can login the HUAWEI support website to download the software packet in accordance to the version of the TOE. Users can verify the software by digital signature(The digital signature is also published on HUAWEI support website) ATN 910D-B: ATN910C-910D_V800R022C00SPC600.cc Info: Users can login the HUAWEI support website to download the software packet in accordance to the version of the TOE. Users can verify the software by digital signature(The digital signature is also published on HUAWEI support website)	V800R022 C00SPC600	2022-11-12
Product guidance	Huawei ATN Series Routers Operational user Guidance.pdf Info: The documentation is delivered by email.	1.2	2025-5-24
	Huawei ATN Series Routers Preparative Procedures.pdf Info: The documentation is delivered by email.	1.3	2025-07-10
	ATN 980C, 980B, 950D, 950C, 950B, 910C, 910D, 905 V800R022C00SPC600 Upgrade Guide.pdf Info: The documentation is delivered by email.	1.1	2022-10-31
	ATN 980C, 980B, 950D, 950C, 950B, 910D, 910C and 905 V800R022C00 Product Documentation Info: Users can obtain documents from Huawei engineers by email. The document format is *.hdx.	1.0	2022-11-23

1.5 Logical Scope of the TOE

The TOE is comprised of several security features. Each of the security features identified above consists of several security functionalities, as identified below.

(1) Security audit

The log module of the host software records operations on a device and events that occur to a device. The recorded operations and events are log messages. Log messages provide evidence for diagnosing and maintaining a system. Log messages reflect the operating status of a device and are used to analyze the conditions of a network and to find out the causes of network failure or faults.

Key elements of log messages include timestamp, host name, Huawei identity, version, module name, severity, brief description, etc.

IC component are the module processing, outputting log records. Information hierarchy is designed to help the user roughly differentiate between information about normal operation and information about faults. Since the information center needs to output information to the terminal, console, log buffer, and log file.

(2) Cryptographic support

The TOE provides cryptography in support of secure connections that includes remote administrative management.

The cryptographic services provided by the TOE are described in table below.

Table 1-4 Cryptography provided by TOE

Cryptography Function	Use in the TOE
DRBG	Used in session establishment of TLS and SSH
ECDSA	Used in the authentication of SSH
ECDH	Used in session establishment of SSH
RSA	Used in the authentication of TLS
DHE	Used in session establishment of TLS
SHA	Used to provide cryptographic hashing services
HMAC-SHA	Used to provide integrity and authentication verification
AES	Used to encrypt traffic transmitted through TLS and SSH

(3) Identification and authentication

The authentication functionality provides validation by user’s account name and password. Public key authentication is supported for SSH users. Detailed functionalities, for example max idle-timeout period, max log-in attempts, UI lock, user kick out, can be applied by administrator according to networking environment, customized security considerations, differential user role on TOE, and/or other operational concerns.

(4) Secure Management

The TOE restricts the ability to determine the behavior of and modify the behavior of the functions transmission of audit data to the security administrator. Only the security administrator can manage the cryptographic keys. Only the security administrator has the right of opening/closing the security services and creation/deletion/modification of the user accounts.

(5) Protection of the TSF

The TOE protects the pre-shared keys, symmetric keys, and private keys from reading them by an unauthorized entity. The TOE stores the users or administrator passwords in non-plaintext form preventing them from reading. The TOE verifies the packet before their installation and uses the digital signature.

(6) TOE access through user authentication

The TOE provides communication security by implementing SSH protocol.

To protect the TOE from eavesdrop and to ensure data transmission security and confidentiality, SSH implements:

- authentication by password or by public-key;
- AES encryption algorithms;
- secure cryptographic key exchange;
- Besides default TCP port 22, manually specifying a listening port is also implemented since it can effectively reduce attacks.

(7) Trusted path and channels for device authentication

The TOE supports the trusted connections using TLS for the communication with the audit server.

2 conformance claims

This Security Target and the TOE described are in accordance with the requirements of [CC].

This Security Target claims conformance with the following parts of Common Criteria:

- Conformant to [CC40R1P2] extended with the extended security functional components defined in chap. 5.
- Conformant to [CC40R1P3].

The methodology to be used for the evaluation is described in the **[CEM]** with an evaluation assurance level of EAL2 augmented with ALC_FLR.2.

The chosen assurance level for this Evaluation is EAL2 augmented with ALC_FLR.2, which the vulnerability analysis level is AVA_VAN.2 and conforms the assurance security requirements:ASE_CCL.1， ASE_ECD.1, ASE_INT.1, ASE_OBJ.2, ASE_REQ.2, ASE_SPD.1, ASE_TSS.1, ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1, ALC_CMC.2, ALC_FLR.2, ALC_CMS.2, ALC_DEL.1, ATE_COV.1, ATE_FUN.1, ATE_IND.2which are defined in [CC40R1P5] .

This Security Target does not claim conformance with any protection profile.

3 Security Problem Definition

3.1 Assets

The owner of the TOE presumably places value upon the following entities as long as they are in the scope of the TOE.

Table 3-1TOE Assets

Asset Name	Description
Audit data	The data which is provided during security audit logging. TOE Security characteristic: integrity.
Authentication data	The data which is used to identify and authenticate the external entities such as account, password, certificate, etc. TOE Security characteristic: confidentiality, integrity.
Cryptography data	The data which is used for digital signature and encryption/decryption such as key. TOE Security characteristic: confidentiality, integrity.
Management data	The data which is used for software updates, and software integrity checking. TOE Security characteristic: integrity.
Configuration data	TOE Security characteristic: integrity.
Software &firmware	device firmware; software; TOE Security characteristic: integrity.
Critical network traffic	Administration traffic; Authentication traffic containing Authentication data; Audit traffic; traffic containing cryptography data; traffic containing Management data TOE Security characteristic: confidentiality, integrity.
Security Functionality of the Device	The TOE Security Functions (TSF) (Remark: In the context of this ST the Security Functionality of the device refers to the security functions of the TOE). TOE Security characteristic: integrity.
Network on which the device resides	The network on which the device resides. TOE Security characteristic: integrity.
Network device	The network device itself. TOE Security characteristic: integrity.

Trust relations with other network devices	Trust relations of the TOE with other network devices. TOE Security characteristic: integrity, authenticity.

3.2 Threats

The threats for the Network Device are grouped according to functional areas of the device in the sections below.

3.2.1 T.UNAUTHORIZED_ADMINISTRATOR_ACCESS

Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.

SFR Rationale:

- The Administrator role is defined in FMT_SMR.2 and the relevant administration capabilities are defined in FMT_SMF.1 and FMT_MTD.1/CoreData, with optional additional capabilities in FMT_MOF.1/Services and FMT_MOF.1/Functions
- The actions allowed before authentication of an Administrator are constrained by FIA_UIA_EXT.1, and include the advisory notice and consent warning message displayed according to FTA_TAB.1
- The requirement for the Administrator authentication process is described in FIA_UAU_EXT.2
- Locking of Administrator sessions is ensured by FTA_SSL_EXT.1 (for local sessions), FTA_SSL.3 (for remote sessions), and FTA_SSL.4 (for all interactive sessions)
- The secure channel used for remote Administrator connections is specified in FTP_TRP.1/Admin
- (Malicious actions carried out from an Administrator session are separately addressed by T.UNDETECTED_ACTIVITY)
- (Protection of the Administrator credentials is separately addressed by T.PASSWORD_CRACKING).

3.2.2 T.WEAK_CRYPTOGRAPHY

Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.

SFR Rationale:

- Requirements for key generation and key distribution are set in FCS_CKM.1 and FCS_CKM.2 respectively
- Requirements for use of cryptographic schemes are set in FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash

- Requirements for random bit generation to support key generation and secure protocols (see SFRs resulting from T.UNTRUSTED_COMMUNICATION_CHANNELS) are set in FCS_RBG.1
- Management of cryptographic functions is specified in FMT_SMF.1

3.2.3 T.UNTRUSTED_COMMUNICATION_CHANNELS

Threat agents may attempt to target Network Devices that do not use standardized secure tunneling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.

SFR Rationale:

- The general use of secure protocols for identified communication channels is described at the top level in FTP_ITC.1 and FTP_TRP.1/Admin
- Requirements for the use of secure communication protocols are set for all the allowed protocols in FCS_SSHC_EXT.1, FCS_SSHS_EXT.1, FCS_TLSC_EXT.1
- Optional and selection-based requirements for use of public key certificates to support secure protocols are defined in FIA_X509_EXT.1/Rev, FIA_X509_EXT.2

3.2.4 T.WEAK_AUTHENTICATION_ENDPOINTS

Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.

SFR Rationale:

- The use of appropriate secure protocols to provide authentication of endpoints (as in the SFRs addressing T.UNTRUSTED_COMMUNICATION_CHANNELS) are ensured by the requirements in FTP_ITC.1 and FTP_TRP.1/Admin

3.2.5 T.UPDATE_COMPROMISE

Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.

SFR Rationale:

- Requirements for protection of updates are set in FPT_TUD_EXT.1
- Certificate-based protection of signatures is supported by the X.509 certificate processing requirements in FIA_X509_EXT.1/Rev, FIA_X509_EXT.2
- Requirements for management of updates are defined in FMT_SMF.1 and (for manual updates) in FMT_MOF.1/ManualUpdate

3.2.6 T.UNDETECTED_ACTIVITY

Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.

SFR Rationale:

- Requirements for basic auditing capabilities are specified in FAU_GEN.1 and FAU_GEN.2, with timestamps provided according to FPT_STM.1
- Requirements for protecting audit records stored on the TOE are specified in FAU_STG. 2
- Requirements for secure transmission of local audit records to an external IT entity via a secure channel are specified in FAU_STG.1 and FAU_STG.5
- Additional requirements for dealing with potential loss of locally stored audit records are specified in FAU_STG.4
- Configuration of the audit functionality is specified in FMT_SMF.1 and confining this functionality to Security Administrators is required by FMT_MOF.1/Functions.

3.2.7 T.CRYPTO_KEY_COMPROMISE

Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.

SFR Rationale:

- Protection of secret/private keys against compromise is specified in FPT_SKP_EXT.1
- Secure destruction of keys is specified in FCS_CKM.6
- Management of keys is specified in FMT_SMF.1 and confining this functionality to Security Administrators is required by FMT_MTD.1/CryptoKeys
- (Protection of passwords is separately covered under T.PASSWORD_CRACKING)

3.2.8 T.PASSWORD_CRACKING

Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices.

SFR Rationale:

- Requirements for password lengths and available characters are set in FIA_PMG_EXT.1
- Protection of password entry by providing only obscured feedback is specified in FIA_UAU.7
- Actions on reaching a threshold number of consecutive password failures are specified in FIA_AFL.1
- Requirements for secure storage of passwords are set in FPT_APW_EXT.1.

3.2.9 T.SECURITY_FUNCTIONALITY_FAILURE

An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

SFR Rationale:

- Requirements for running self-test(s) are defined in FPT_TST.1

3.3 Assumptions

This section describes the assumptions made in identification of the threats and security requirements for Network Devices. The Network Device is not expected to provide assurance in any of these areas, and as a result, requirements are not included to mitigate the threats associated.

3.3.1 A.PHYSICAL_PROTECTION

The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the ST does not include any requirements on physical tamper protection or other physical attack mitigations. The ST does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device.

Since the entropy source for the Network Device is in the processor for the device, the operating environment for the entropy source is the same as that expected for the Network Device. The Network Device operating temperature is between 0°C to +40°C.

[OE.PHYSICAL]

3.3.2 A.LIMITED_FUNCTIONALITY

The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality).

[OE.NO_GENERAL_PURPOSE]

3.3.3 A.NO_THRU_TRAFFIC_PROTECTION

A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by this ST.

[OE.NO_THRU_TRAFFIC_PROTECTION]

3.3.4 A.TRUSTED_ADMINISTRATOR

The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device.

[OE.TRUSTED_ADMIN]

3.3.5 A.REGULAR_UPDATES

The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

[OE.UPDATES]

3.3.6 A.ADMIN_CREDENTIALS_SECURE

The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.

[OE.ADMIN_CREDENTIALS_SECURE]

3.3.7 A.RESIDUAL_INFORMATION

The Administrator must ensure 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.RESIDUAL_INFORMATION]

3.4 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs.

3.4.1 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.

SFR Rationale:

- An advisory notice and consent warning message is required to be displayed by FTA_TAB.1

4 Security Objectives

The security objectives are high level declarations, concise and abstract of the solution to the problem exposed in the former section, which counteracts the threats and fulfills the security policies and the assumptions. These consist of:

- the security objectives for the operational environment.
- the security objectives for the TOE

4.1 Security Objectives for the TOE

4.1.1 O.ADMIN_AUTH

The TOE shall require identification and authentication of administrators before granting them access to the TOE management functions. The TOE management capabilities available to administrators shall be defined and limited, and actions available prior to authentication shall be limited and constrained. Administrators' authentication process shall consist in local authentication on the TOE using passwords through a secure communication channel, and authentication sessions will be closed by the TOE after a defined period of inactivity.

4.1.2 O.STRONG_CRYPTO

The TOE shall use robust cryptographic algorithms, compliant to industry approved standards, in order to provide robust cryptographic protection of network communications.

4.1.3 O.TRUSTED_COMM

The TOE shall implement secure channels that use standardized tunneling protocols to protect the critical network traffic, ensuring protection of the communications between the TOE and trusted external entities in confidentiality, integrity, and protection against communication replays.

4.1.4 O.STRONG_AUTHENTICATION_ENDPOINT

The TOE shall implement methods for robust and reliable authentication of trusted entities with the TOE, preventing attacks based on weak authentication methods (e.g. guessing or transported shared keys).

4.1.5 O.SECURE_UPDATES

The TOE shall provide to administrators the capability of installing software or firmware updates only after a successful verification of their authenticity using secure and strong cryptographic algorithms based on digital signatures.

4.1.6 O.ACTIVITY_AUDIT

The TOE shall generate audit records for relevant management actions carried by administrators. Audit records will be marked with precise timestamps, associated to user identity, and protected from unauthorized modification or deletion. The TOE will transmit the audit logs to SYSLOG server regularly.

4.1.7 O.CRYPTO_KEY_PROTECTION

The TOE shall protect stored cryptographic keys in a way that prevents unauthorized access. Management of cryptographic keys shall be restricted to Security Administrators and key destruction shall be performed in a secure way that prevents key recover from residual information.

4.1.8 O.PASSWORD_PROTECTION

The TOE shall protect the passwords user for local administrator authentication by enforcing complexity and quality rules. Also, the TOE shall limit failed authentication attempts and limit the feedback given to users on failed authentications, in order to prevent brute force or guessing attacks. Also, the TOE shall perform secure storage of passwords, refraining from storing them in plaintext.

4.1.9 O.SELF_TEST

The TOE shall perform self-tests of the TSF functionality in order to detect potential failures during start-up or operation and prevent further situations that could lead to malfunction.

4.1.10 O.BANNER

The TOE shall display an advisory notice and consent about use of the TOE to administrators before establishing an administrative user session

4.2 Security Objectives for the Operational Environment

The following subsections describe security objectives for the Operational Environment.

4.2.1 OE.PHYSICAL

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

4.2.2 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.

4.2.3 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.

4.2.4 OE.TRUSTED_ADMIN

Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE’s trust store in case such certificate can no longer be trusted.

4.2.5 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.

4.2.6 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.

4.2.7 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.

4.3 Security Objectives Rationale

The following table provides a mapping of security objectives tracing each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective, and each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective. This illustrates that the security objectives counter all threats, the security objectives enforce all OSPs and the security objectives for the operational environment uphold all assumptions.

Table 4-1 SFRs / TOE Security Objectives coverage

	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENTICATION_ENDPOINT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PROTECTION	O.PASSWORD_PROTECTION	O.SELF_TEST	O.BANNER	OE.PHYSICAL	OE.NO_GENERAL_PURPOSE	OE.NO_THRU_TRAFFIC_PROTECTION	OE.TRUSTED_ADMIN	OE.UPDATES	OE.ADMIN_CREDENTIALS_SECURE	OE.RESIDUAL_INFORMATION
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	X																
T.WEAK_CRYPTOGRAPHY		X															
T.UNTRUSTED_COMMUNICATION_CHANNELS			X														
T.WEAK_AUTHENTICATION_ENDPOINTS				X													
T.UPDATE_COMPROMISE					X												
T.UNDETECTED_ACTIVITY						X											
T.CRYPTO _KEY_COMPROMISE							X										
T. PASSWORD_CRACKING								X									
T.SECURITY_FUNCTIONALITY_FAILURE									X								
P.ACCESS_BANNER										X							
A.PHYSICAL_PROTECTION											X						
A.LIMITED_FUNCTIONALITY												X					
A.NO_THRU_TRAFFIC_PROTECTION													X				

	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENTICATION_ENDPOINT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PROTECTION	O.PASSWORD_PROTECTION	O.SELF_TEST	O.BANNER	OE.PHYSICAL	OE.NO_GENERAL_PURPOSE	OE.NO_THRU_TRAFFIC_PROTECTION	OE.TRUSTED_ADMIN	OE.UPDATES	OE.ADMIN_CREDENTIALS_SECURE	OE.RESIDUAL_INFORMATION
A.TRUSTED_ADMINISTRATOR														X			
A.REGULAR_UPDATES															X		
A.ADMIN_CREDENTIALS_SECURE																X	
A.RESIDUAL_INFORMATION																	X

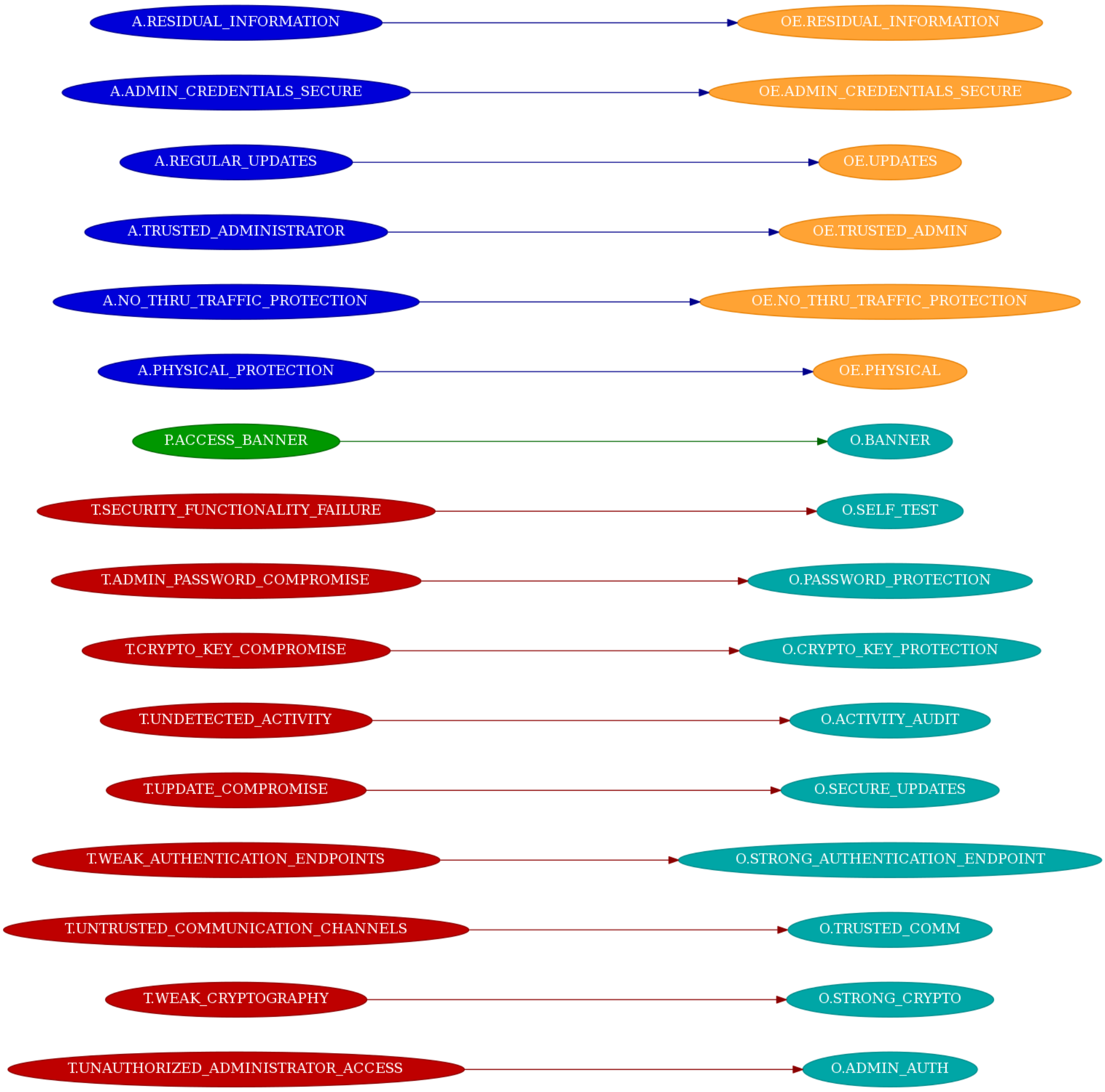


Figure 4-1 Mapping of Security Problem Definition to Security Objectives

4.3.1 Threats

T.UNAUTHORIZED_ADMINISTRATOR_ACCESS: This threat is countered by **O.ADMIN_AUTH** which requires identification and authentication of administrator before granting them access to the TOE and to management functions. It also enforces that the TOE requires identification and authentication of administrators before granting them access to the TOE management functions.

The TOE management capabilities available to administrators shall be defined and limited, and actions available prior to authentication shall be limited and constrained.

Administrators' authentication process shall consist in local authentication of the TOE using passwords through a secure communication channel, and authentication sessions will be closed by the TOE after a defined period of inactivity.

T.WEAK_CRYPTOGRAPHY: This threat is countered by **O.STRONG_CRYPTO** which requires usage of robust cryptographic algorithms, compliant to industry approved standards, in order to provide robust cryptographic protection of network communications.

T.UNTRUSTED_COMMUNICATION_CHANNELS: This threat is countered by **O.TRUSTED_COMM** which requires secure communication channels that use standardized tunneling protocols to protect the critical network traffic, ensuring protection of the communications between the TOE and trusted external entities in confidentiality, integrity, and protection against communication replays.

T.WEAK_AUTHENTICATION_ENDPOINTS: This threat is countered by **O.STRONG_AUTHENTICATION_ENDPOINT** which requires methods for strong authentication of trusted entities with the TOE, preventing attacks based on weak authentication methods.

T.UPDATE_COMPROMISE: This threat is countered by **O.SECURE_UPDATES** which requires verification of updates authenticity by administrators based on cryptographic digital signatures.

T.UNDETECTED_ACTIVITY: This threat is countered by **O.ACTIVITY_AUDIT** which requires the generation of audit records for relevant management actions carried by administrators, marked with precise timestamps, associated to user identity, and protected from unauthorized modification or deletion.

T.CRYPTO_KEY_COMPROMISE: This threat is countered by **O.CRYPTO_KEY_PROTECTION** which requires protection of stored cryptographic keys in order to prevent unauthorized access, restricting management of cryptographic keys to administrators, and enforcing secure key destruction methods.

T.PASSWORD_CRACKING: This threat is countered by **O.PASSWORD_PROTECTION** which requires the TOE to enforce password complexity and quality in passwords used by administrators for authentication, hence preventing successful attacks to weak passwords. The same objective also forbids plaintext storage of passwords in the TOE and prevents attacks based on massive authentication attempts or guessing passwords from feedback resulting from failed authentication attempts.

T.SECURITY_FUNCTIONALITY_FAILURE: This threat is countered by **O.SELF_TEST** which requires that The TOE carries outs TSF self-tests in order to detect potential failures during start-up or operation and prevent further situations that could lead to malfunction.

The following table maps the threats of the security problem established to the security objectives of the TOE and the security objectives of the operational environment.

Table 4-2 Threats vs Security Objectives

Threats	Security Objectives
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	O.ADMIN_AUTH
T.WEAK_CRYPTOGRAPHY	O.STRONG_CRYPTO
T.UNTRUSTED_COMMUNICATION_CHANNELS	O.TRUSTED_COMM
T.WEAK_AUTHENTICATION_ENDPOINTS	O.STRONG_AUTHENTICATION_ENDPOINT
T.UPDATE_COMPROMISE	O.SECURE_UPDATES
T.UNDETECTED_ACTIVITY	O.ACTIVITY_AUDIT
T.CRYPTO_KEY_COMPROMISE	O.CRYPTO_KEY_PROTECTION
T. PASSWORD_CRACKING	O.PASSWORD_PROTECTION
T.SECURITY_FUNCTIONALITY_FAILURE	O.SELF_TEST

4.3.2 Organizational Security Policies

P.ACCESS_BANNER: This policy is enforced by **O.BANNER** which requires that the TOE displays an advisory notice and consent about use of the TOE to administrators before establishing an administrative user session.

The following table maps the organizational security policies of the problem established to the security objectives of the TOE and the security objectives of the operational environment.

Table 4-3 OSPs vs Security Objectives

OSPs	Security Objectives
P.ACCESS_BANNER	O.BANNER

4.3.3 Assumptions

A.PHYSICAL_PROTECTION: This assumption is directly upheld by **OE.PHYSICAL**, which requires that physical protection to the TOE is provided by the operational environment.

A.LIMITED_FUNCTIONALITY: This assumption is directly upheld by **OE.NO_GENERAL_PURPOSE,** which requires that the TOE provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing.

A.NO_THRU_TRAFFIC_PROTECTION: This assumption is directly upheld by **OE.NO_THRU_TRAFFIC_PROTECTION**, which requires that the TOE does not provide any protection of traffic that traverses it, but such protection is covered by other security and assurance measures in the operational environment.

A.TRUSTED_ADMINISTRATOR: This assumption is directly upheld by **OE.TRUSTED_ADMIN**, which requires that TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

A.REGULAR_UPDATES: This assumption is directly upheld by **OE.UPDATES**, which requires that 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.

A.ADMIN_CREDENTIALS_SECURE: This assumption is directly upheld by **OE.ADMIN_CREDENTIALS_SECURE**, which requires that the administrator's credentials (private key) used to access the TOE are protected on any other platform on which they reside.

A.RESIDUAL_INFORMATION: This assumption is directly upheld by **OE.RESIDUAL_INFORMATION** which requires administrators to ensure that there is no unauthorized access possible for sensitive residual information on networking equipment when the equipment is discarded or removed from its operational environment.

The following table maps the assumptions of the problem established to the security objectives of the TOE and the security objectives of the operational environment.

Table 4-4 Assumptions vs Security Objectives for the Operational Environment

Assumptions	Security Objectives
A.PHYSICAL_PROTECTION	OE.PHYSICAL
A.LIMITED_FUNCTIONALITY	OE.NO_GENERAL_PURPOSE
A.NO_THRU_TRAFFIC_PROTECTION	OE.NO_THRU_TRAFFIC_PROTECTION

Assumptions	Security Objectives
A.TRUSTED_ADMINISTRATOR	OE.TRUSTED_ADMIN
A.REGULAR_UPDATES	OE.UPDATES
A.ADMIN_CREDENTIALS_SECURE	OE.ADMIN_CREDENTIALS_SECURE
A.RESIDUAL_INFORMATION	OE.RESIDUAL_INFORMATION

5 Extended Components Definition

5.1 Class FCS: Cryptographic support

The TSF may employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation. This class is used when the TOE implements cryptographic functions, the implementation of which could be in hardware, firmware and/or software.

The FCS class is composed of four families: FCS_CKM, FCS_COP, FCS_RBG and FCS_RNG. The FCS_CKM family addresses the management aspects of cryptographic keys, while the FCS_COP family is concerned with the operational use of those cryptographic keys. The FCS_RBG and FCS_RNG family define the random bit generation and random number generation.

FCS class is extended in order to add the families FCS_TLSC_EXT, FCS_SSHC_EXT and FCS_SSHS_EXT.FCS_TLSC_EXT cover cryptographic requirements associated to TLS communications. FCS_SSHC_EXT and FCS_SSHS_EXT includes cryptographic requirements related to SSH client and server implementation. None of the requirements of those classes already exist in the Common Criteria standard.

FCS class is extended in order to add the families FCS_TLSC_EXT, FCS_SSHC_EXT and FCS_SSHS_EXT. FCS_TLSC_EXT cover cryptographic requirements associated to TLS communications. , FCS_SSHC_EXT and FCS_SSHS_EXT includes cryptographic requirements related to SSH client and server implementation. None of the requirements of those classes already exist in the Common Criteria standard.

5.1.1 SSH Client (FCS_SSHC_EXT)

Family Behaviour

The component in this family addresses the ability for a client to use SSH to protect data between the client and a server using the SSH protocol.

Component levelling

FCS_SSHC_EXT.1 SSH Client requires that the client side of SSH be implemented as specified.

Management: FCS_SSHC_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FCS_SSHC_EXT.1

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

a) Failure of SSH session establishment
b) SSH session establishment
c) SSH session termination

FCS_SSHC_EXT.1 SSH Client Protocol

Hierarchical to: No other components
Dependencies:

FCS_CKM.1Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key destribution
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)

FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG.1 Random Bit Generation

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs *4251, 4252, 4253, 4254,* [selection: *4256, 4344, 5647, 5656, 6187, 6668, 8268, 8308 section 3.1, 8332*].
FCS_SSHC_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, [selection: *password-based, no other method*].
FCS_SSHC_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than *[assignment: number of bytes]* bytes in an SSH transport connection are dropped.
FCS_SSHC_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [assignment: *list of encryption algorithms*].
FCS_SSHC_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: *ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, x509v3-ssh-rsa, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, x509v3-rsa2048-sha256*] as its public key algorithm(s) and rejects all other public key algorithms
FCS_SSHC_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *[assignment: list of data integrity MAC algorithms]* as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).
FCS_SSHC_EXT.1.7 The TSF shall ensure that *[assignment: list of key exchange methods]* are the only allowed key exchange methods used for the SSH protocol.
FCS_SSHC_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 each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

FCS_SSHC_EXT.1.9 The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and [selection: *a list of trusted certification authorities, no other methods*] as described in RFC 4251 section 4.1.

5.1.2 SSH Server Protocol (FCS_SSHS_EXT)

Family behavior

The component in this family addresses the ability for a server to offer SSH to protect data between a client and the server using the SSH protocol.

Component levelling

SSH Server requires that the server side of SSH be implemented as specified.

Management: FCS_SSHS_EXT.1

The following actions could be considered for the management functions in FMT:
- There are no management activities foreseen.

Audit: FCS_SSHS_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:
- Failure of SSH session establishment
- SSH session establishment
- SSH session termination

FCS_SSHS_EXT.1: SSH Server Protocol

Hierarchical to:

No other components.

Dependencies:

FCS_CKM.1

FCS_CKM.2

FCS_COP.1/DataEncryption

FCS_COP.1/SigGen

FCS_COP.1/Hash

FCS_COP.1/KeyedHash

FCS_RBG.1

FCS_SSHS_EXT.1.1: ***The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, [selection: 4256, 4344, 5647, 5656, 6668, 8268, 8308 section 3.1, 8332]***

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, [selection: password-based, no other method]***.

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

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: [assignment: encryption algorithms]***.

FCS_SSHS_EXT.1.5: ***The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, x509v3-ssh-rsa, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, x509v3-rsa2048-sha256] as its public key algorithm(s) and rejects all other public key algorithms.***

FCS_SSHS_EXT.1.6: ***The TSF shall ensure that the SSH transport implementation uses [assignment: list of MAC algorithms] as its MAC algorithm(s) and rejects all other MAC algorithm(s).***

FCS_SSHS_EXT.1.7: ***The TSF shall ensure that [selection: diffie-hellman-group14-sha1, diffie-hellman-group15-sha512, ecdh-sha2-nistp256] and [selection: diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, diffie-hellman-group17-sha512, diffie-hellmangroup18-sha512, ecdh-sha2-nistp384, ecdh-sha2-nistp521, no other methods] are the only allowed key exchange methods used for the SSH protocol.***

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 each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

5.1.3 TLS Client Protocol (FCS_TLSC_EXT)

Family behavior

The component in this family addresses the ability for a client to use TLS to protect data between the client and a server using the TLS protocol.

Component levelling

TLS Client requires that the client side of TLS be implemented as specified.

Management: FCS_TLSC_EXT.1

The following actions could be considered for the management functions in FMT:
- There are no management activities foreseen.

Audit: FCS_TLSC_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:
- Failure of TLS session establishment
- TLS session establishment
- TLS session termination

FCS_TLSC_EXT.1: TLS Client Protocol

Hierarchical to:

No other components.

Dependencies:

FCS_CKM.1

FCS_CKM.2

FCS_COP.1/DataEncryption

FCS_COP.1/SigGen

FCS_COP.1/Hash

FCS_COP.1/KeyedHash

FCS_RBG.1

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.2 X.509 Certificate Authentication

FCS_TLSC_EXT.1.1: ***The TSF shall implement [selection: 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 [selection: TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268, TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268, TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268, TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492, TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246, TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246, TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246, TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246, TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288, 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 ciphersuites.

FCS_TLSC_EXT.1.2: ***The TSF shall verify that the presented identifier matches [selection:the reference identifier per RFC 6125 section 6, IPv4 address in CN or SAN, IPv6 address in the CN or SAN, IPv4 address in SAN, IPv6 address in the SAN, the identifier per RFC 5280 Appendix A using [selection: id-at-commonName, id-at-countryName, id-at-dnQualifier, idat-generationQualifier, id-at-givenName, id-at-initials, id-at-localityName, id-at-name, id-atorganizationalUnitName, id-at-organizationName, id-at-pseudonym, id-at-serialNumber, idat-stateOrProvinceName, id-at-surname, id-at-title] and no other attribute types].***

FCS_TLSC_EXT.1.3: ***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 [selection: Not implement any administrator override mechanism, require administrator authorization to establish the connection if the TSF fails to [selection: match the reference identifier, validate certificate path, validate expiration date, determine the revocation status] of the presented server certificate]***

FCS_TLSC_EXT.1.4: ***The TSF shall [selection: not present the Supported Elliptic Curves Extension, present the Supported Elliptic Curves Extension with the following NIST curves: [selection: secp256r1, secp384r1, secp521r1] and no other curves] in the Client Hello.***

5.2 Class FIA: Identification and authentication

Families in this class address the requirements for functions to establish and verify a claimed user identity.

Identification and Authentication is required to ensure that users are associated with the proper security attributes (e.g. identity, groups, roles, security or integrity levels).

The unambiguous identification of authorised users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies. The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorised user. Other classes of requirements (e.g. User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

FIA class is extended in order to add the families FIA_X509_EXT, FIA_UIA_EXT, FIA_PGM_EXT, and FIA_UAU_EXT.
Those families are defined to include new SFRs related to identification based on X509 cryptography (FIA_X509_EXT), common requirements for identification and authentication (FIA_UIA_EXT), password management requirements (FIA_PGM_EXT) and additional user authentication requirements (FIA_UAU_EXT). In the case of FIA_UAU_EXT, this new family is added due to the meaningful differences between the extended components defined and those already existing in FIA_UAU family.

5.2.1 Password Management (FIA_PMG_EXT)

Family behavior

The TOE defines the attributes of passwords used by administrative users to ensure that strong passwords and passphrases can be chosen and maintained.

Component levelling

FIA_PMG_EXT: Password Management
1

Password management requires the TSF to support passwords with varying composition requirements, minimum lengths, maximum lifetime, and similarity constraints.

Management: FIA_PMG_EXT.1

No management functions.

Audit: FIA_PMG_EXT.1

No specific audit requirements.

FIA_PMG_EXT.1: Password Management

Hierarchical to:

No other components.

Dependencies:

No dependencies.

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: [selection: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [assignment: other characters]]***

b) ***Minimum password length shall be configurable to between [assignment: minimum number of characters supported by the TOE] and [assignment: number of characters greater than or equal to 15] characters.***

5.2.2 User Identification and Authentication (FIA_UIA_EXT)

Family behavior

The TSF allows certain specified actions before the non-TOE entity goes through the identification and authentication process.

Component levelling

User Identification and Authentication requires Administrators (including remote Administrators) to be identified and authenticated by the TOE, providing assurance for that end of the communication path.   It also ensures that every user is identified and authenticated before the TOE performs any mediated functions

Management: FIA_UIA_EXT.1

The following actions could be considered for the management functions in FMT:
- Ability to configure the list of TOE services available before an entity is identified and authenticated

Audit: FIA_UIA_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- All use of the identification and authentication mechanism
- Provided user identity, origin of the attempt (e.g. IP address)

FIA_UIA_EXT.1: User Identification and Authentication

Hierarchical to:

No other components.

Dependencies:

FTA_TAB.1

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:***

a) ***Display the warning banner in accordance with FTA_TAB.1***

b) ***[selection: no other actions, automated generation of cryptographic keys, [assignment: list of services, actions performed by the TSF in response to non-TOE requests]]***

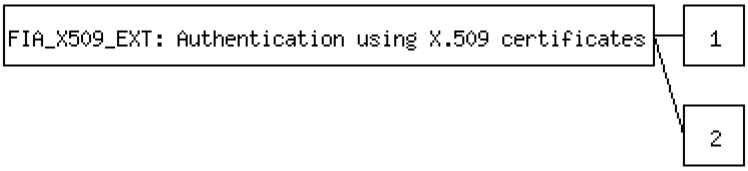
FIA_UIA_EXT.1.2: ***The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user.***

5.2.3 Authentication using X.509 certificates (FIA_X509_EXT)

Family behavior

This family defines the behaviour, management, and use of X.509 certificates for functions to be performed by the TSF. Components in this family require validation of certificates according to a specified set of rules, use of certificates for authentication for protocols and integrity verification, and the generation of certificate requests.

Component levelling



X509 Certificate Validation, requires the TSF to check and validate certificates in accordance with the RFCs and rules specified in the component.

X509 Certificate Authentication, requires the TSF to use certificates to authenticate peers in protocols that support certificates, as well as for integrity verification and potentially other functions that require certificates.

Management: FIA_X509_EXT.1

The following actions could be considered for the management functions in FMT:
- Remove imported X.509v3 certificates
- Approve import and removal of X.509v3 certificates
- Initiate certificate requests

Management: FIA_X509_EXT.2

The following actions could be considered for the management functions in FMT:
- Remove imported X.509v3 certificates
- Approve import and removal of X.509v3 certificates
- Initiate certificate requests

Audit: FIA_X509_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- Minimal: No specific audit requirements are specified.

Audit: FIA_X509_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- Minimal: No specific audit requirements are specified.

FIA_X509_EXT.1: X.509 Certificate Validation

Hierarchical to:

No other components.

Dependencies:

FIA_X509_EXT.2 X.509 Certificate Authentication

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

- ***RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.***
- ***The certification path must terminate with a trusted CA certificate designated as a trust anchor.***
- ***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 [selection: the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5]***
- ***The TSF shall validate the extendedKeyUsage field according to the following rules:***

[assignment: rules that govern contents of the extendedKeyUsage field that need to be verified].

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: X509 Certificate Authentication

Hierarchical to:

No other components.

Dependencies:

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.2.1: ***The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [selection: DTLS, HTTPS, IPsec, TLS, SSH, [assignment: other protocols], no protocols], and [selection: code signing for system software updates, code signing for integrity verification, [assignment: other uses], no additional uses]***

FIA_X509_EXT.2.2: ***When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [selection: allow the Administrator to choose whether to accept the certificate in these cases, accept the certificate, not accept the certificate]***

5.2.4 User authentication (FIA_UAU_EXT)

Family behavior

Provides for a locally based administrative user authentication mechanism.

Component levelling

The password-based authentication mechanism provides administrative users a locally based authentication mechanism.

Management: FIA_UAU_EXT.2

The following actions could be considered for the management functions in FMT:
- None

Audit: FIA_UAU_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- Minimal: All use of the authentication mechanism

FIA_UAU_EXT.2: Password-based Authentication Mechanism

Hierarchical to:

No other components.

Dependencies:

No dependencies.

FIA_UAU_EXT.2.1: ***The TSF shall provide a local [selection: password-based, SSH public key-based, certificate-based, [assignment: other authentication mechanism(s)]] authentication mechanism, to perform local administrative user authentication.***

5.3 Class FPT: Protection of the TSF

This class contains families of functional requirements that relate to the integrity and management of the mechanisms that constitute the TSF and to the integrity of TSF data. In some sense, families in this class may appear to duplicate components in the FDP class; they may even be implemented using the same mechanisms. However, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary to provide requirements that the SFPs in the TOE cannot be tampered with or bypassed.

From the point of view of this class, regarding to the TSF there are three significant elements:

- The TSF's implementation, which executes and implements the mechanisms that enforce the SFRs.
- The TSF's data, which are the administrative databases that guide the enforcement of the SFRs.

- The external entities that the TSF may interact with in order to enforce the SFRs.

FPT class is extended in order to add the families FPT_TUD_EXT, FPT_TST, FPT_SKP_EXT and FPT_APW_EXT. In the case of FPT_TST, a self-test requirement for the TOE is added that presents meaningful differences with those existing in FPT_TST. The rest of the mentioned families include new requirements related to TOE self-protection that are not covered in any of the existing FPT families of the Common Criteria standard.

5.3.1 Protection of TSF Data (FPT_SKP_EXT)

Family behavior

Components in this family address the requirements for managing and protecting TSF data, such as cryptographic keys. This is a new family modelled after the FPT_PTD Class.

Component levelling

Protection of TSF Data (for reading all symmetric keys), requires preventing symmetric keys from being read by any user or subject. It is the only component of this family.

Management: FPT_SKP_EXT.1

The following actions could be considered for the management functions in FMT:
- There are no management activities foreseen.

Audit: FPT_SKP_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- There are no auditable events foreseen.

FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)

Hierarchical to:

No other components.

Dependencies:

No dependencies.

FPT_SKP_EXT.1.1: ***The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.***

5.3.2 Protection of Administrator Passwords (FPT_APW_EXT)

Family behavior

Components in this family ensure that the TSF will protect plaintext credential data such as passwords from unauthorized disclosure.

Component levelling

Protection of Administrator passwords requires that the TSF prevent plaintext credential data from being read by any user or subject.

Management: FPT_APW_EXT.1

The following actions could be considered for the management functions in FMT:
- No management functions.

Audit: FPT_APW_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- No audit necessary.

FPT_APW_EXT.1: Protection of Administrator Passwords

Hierarchical to:

No other components.

Dependencies:

No dependencies.

FPT_APW_EXT.1.1: ***The TSF shall store passwords in non-plaintext form.***

FPT_APW_EXT.1.2: ***The TSF shall prevent the reading of plaintext passwords.***

5.3.3 Trusted Update (FPT_TUD_EXT)

Family behavior

Components in this family address the requirements for updating the TOE firmware and/or software.

Component levelling

Trusted Update requires management tools be provided to update the TOE firmware and software, including the ability to verify the updates prior to installation.

Management: FPT_TUD_EXT.1

The following actions could be considered for the management functions in FMT:
- Ability to update the TOE and to verify the updates
- Ability to update the TOE and to verify the updates using the digital signature capability (FCS_COP.1/SigGen)
- Ability to update the TOE and to verify the updates using the digital signature capability prior to installing those updates

Audit: FPT_TUD_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- Initiation of the update process.
- Any failure to verify the integrity of the update

FPT_TUD_EXT.1: Trusted Update

Hierarchical to:

No other components.

Dependencies:

FCS_COP.1/SigGen or FCS_COP.1/Hash

FPT_TUD_EXT.1.1: ***The TSF shall provide [assignment: Administrators] the ability to query the currently executing version of the TOE firmware/software and [selection: the most recently installed version of the TOE firmware/software, no other TOE firmware/software version]***

FPT_TUD_EXT.1.2: ***The TSF shall provide [assignment: Administrators] the ability to manually initiate updates to TOE firmware/software and [selection: support automatic checking for updates, support automatic updates, no other update mechanism]***

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

5.4 Class FTA: TOE access

This family specifies functional requirements for controlling the establishment of a user's session.

FTA class is extended in order to add the family FTA_SSL_EXT. This class includes requirements for authenticated sessions, considering different requirements for local and remote user sessions.

5.4.1 TSF-initiated Session Locking (FTA_SSL_EXT)

Family behavior

Components in this family address the requirements for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.
The extended FTA_SSL_EXT family is based on the FTA_SSL family.

Component levelling

FTA_SSL_EXT: TSF-initiated Session Locking
1

TSF-initiated session locking, requires system initiated locking of an interactive session after a specified period of inactivity. It is the only component of this family.

Management: FTA_SSL_EXT.1

The following actions could be considered for the management functions in FMT:
- Specification of the time of user inactivity after which lock-out occurs for an individual user.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:
- Any attempts at unlocking an interactive session.

FTA_SSL_EXT.1: TSF-initiated Session Locking

Hierarchical to:

No other components.

Dependencies:

FIA_UAU.1

FTA_SSL_EXT.1.1: ***The TSF shall, for local interactive sessions [selection: lock the session - disable any activity of the Administrator's data access/display devices other than unlocking the session, and requiring that the Administrator re-authenticate to the TSF prior to unlocking the session, terminate the session] after a Security Administrator-specified time period of inactivity.***

6 Security Functional Requirements

Conventions

The conventions used in descriptions of the SFRs are as follows:

- Refinement: the refinement text is indicated with **bold text**. Where part of the content of a SFR component has been removed, the removed text is shown in ~~strikethroughs~~;
- Selection: the selection values are indicated with underlined text;
- Assignment: the assignment text is indicated with *italicized and underlined text*;
- Iteration: It includes "/" and an "identifier" following requirement identifier that allows to distinguish the iterations of the requirement. Example: FCS_COP.1/XXX..
- Application Notes added by the ST author are called 'Additional Application Note' which are enumerated as 'a', 'b', ... and are formatted with underline such as "Additional Application Note a";
- References: Indicated with [square brackets].

6.1 Functional Security Requirements

6.1.1 Security Audit (FAU)

6.1.1.1 FAU_GEN.1 Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit data 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).*

- *Starting and stopping services.*

d) Specifically defined auditable events listed in 0.

Additional Application Note a: Audit functionality is enabled by default. The auditing functionality cannot be disabled.

Additional Application Note b: The TOE does not support using reset command to reset password directly, but it can modify password in the following way: re-create local-user or change local-user password.

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP, PP-Module, functional package or ST, *information specified in column three of 0.*

Table 6-1 Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
Mandatory Requirements		
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.
FAU_STG.2	None.	None.
FAU_STG.4	Low storage space for audit events.	None.
FAU_STG.5	None	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.6	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_RBG.1	None.	None.
FCS_RBG.3	None.	None.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g. IP address).
FIA_UAU.7	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MTD.1/CoreData	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST.1	None.	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
FPT_STM.1	None	None
FPT_STM.2	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged.)	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).
FPT_FLS.1	Failure of the event defined in FPT_TST.1	None
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None
FTA_TAB.1	None.	None.
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.
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	None
FCS_SSHC_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_TLSC_EXT.1	Failure to establish a TLS Session.	Reason for failure.
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store.
FIA_X509_EXT.2	None.	None.
FMT_MOF.1/Services	Starting and stopping of services.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FMT_MOF.1/Functions	Modification of the behaviour of the transmission of audit data to an external IT entity, the handling of audit data, the audit functionality when Local Audit Storage Space is full.	None.
FMT_MTD.1/CryptoKeys	Management of cryptographic keys.	None.

6.1.1.2 FAU_GEN.2 User identity association

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

6.1.1.3 FAU_STG.1 Audit data Storage location

FAU_STG.1.1 The TSF shall be able to store generated audit data on the TOE itself, transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.

6.1.1.4 FAU_STG.2 Protected audit data storage

FAU_STG.2.1 The TSF shall protect the stored audit data in the audit trail from unauthorized deletion.

FAU_STG.2.2 The TSF shall be able to prevent unauthorized modifications to the stored audit data in the audit trail.

6.1.1.5 FAU_STG.4 Action in case of possible audit data loss

FAU_STG.4.1 The TSF shall *generate a warning to inform the Administrator* if the audit data storage exceeds *the local audit trail storage capacity.*

6.1.1.6 FAU_STG.5 Prevention of audit data loss

FAU_STG.5.1 The TSF shall overwrite the oldest stored audit records, *no other actions* if the audit data storage is full.

6.1.2 Cryptographic Support (FCS)

6.1.2.1 FCS_CKM.1 Cryptographic Key Generation

FCS_CKM.1.1 The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm:

- *FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1*
- *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 [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].~~

6.1.2.2 FCS_CKM.2 Cryptographic Key distribution

FCS_CKM.2.1 The TSF shall ~~distribute cryptographic keys~~ **perform** cryptographic **key establishment** in accordance with a specified cryptographic key ~~distribution~~ **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";*
- *Finite field-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";*

~~that meets the following: [assignment: list of standards].~~

6.1.2.3 FCS_CKM.6 Timing and event of cryptographic key destruction

FCS_CKM.6.1 The TSF shall destroy *plaintext keys in volatile storage or in non-volatile storage* when no longer needed.

FCS_CKM.6.2 The TSF shall destroy cryptographic keys and keying material specified by FCS_CKM.6.1 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 *overwrite consisting of a new value of the key*

that meets the following: *No Standard*.

6.1.2.4 FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/ Decryption)

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in CTR, GCM mode* and cryptographic key sizes *128 bits, 256 bits* that meet the following*: AES as specified in ISO 18033-3, CTR as specified in ISO 10116, GCM as specified in ISO 19772.*

6.1.2.5 FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)

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*: 3072 bits and 4096 bits,*
- *Elliptic Curve Digital Signature Algorithm* and cryptographic key sizes*: 256 bits, 384 bits and 521 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 and P-521; ISO/IEC 14888-3, Section 6.4*

6.1.2.6 FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm: *SHA-256, SHA-384,* and ~~cryptographic key sizes [assignment: cryptographic key sizes]~~ **message digest sizes** *256, 384* **bits** that meet the following: *ISO/IEC 10118-3:2004.*

6.1.2.7 FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm: *HMAC-SHA-256* and cryptographic key sizes: *256 bits for HMAC-SHA-256* **and message digest sizes:** *256* **bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".*

6.1.2.8 FCS_RBG.1 Random Bit Generation（RBG）

FCS_RBG.1.1 The TSF shall perform deterministic random bit generation services using *Hash_DRBG (any)* in accordance with *ISO/IEC 18031:2011 Section C.2.2* after initialization.

FCS_RBG.1.2 The TSF shall use a TSF entropy source *non-physical true random number* for initialization and reseeding.

FCS_RBG.1.3 The TSF shall update the DRBG state by reseeding using a TSF entropy source *non-physical true random number* in the following situations: after *2^24-1 generate operations* in accordance with *ISO/IEC 18031:2011 Section C.2.2*.

6.1.2.9 FCS_RBG.3 Random bit generation (internal seeding – single source)

FCS_RBG.3.1 The TSF shall be able to seed the DRBG using a TSF software-based **entropy** source *non-physical true random number* with *256* bits of min-entropy.

6.1.2.10 FCS_SSHC_EXT.1 SSH Client Protocol

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668.

FCS_SSHC_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.

FCS_SSHC_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than *262144* bytes in an SSH transport connection are dropped.

FCS_SSHC_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: *aes128-ctr, aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com, aes256-gcm@openssh.com.*

FCS_SSHC_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521 as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHC_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *hmac-sha2-256, AEAD_AES_128_GCM, AEAD_AES_256_GCM, implicit* as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHC_EXT.1.7 The TSF shall ensure that *ecdh-sha2-nistp256* are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_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 each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

FCS_SSHC_EXT.1.9 The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and no other methods as described in RFC 4251 section 4.1.

6.1.2.11 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668.

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.

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than *262144* bytes in an SSH transport connection are dropped.

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-ctr, aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com, aes256-gcm@openssh.com.*

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521 as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *hmac-sha2-256, AEAD_AES_128_GCM, AEAD_AES_256_GCM, implicit* as its MAC algorithm(s) and rejects all other MAC algorithm(s).

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.

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 each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

6.1.2.12 FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication

FCS_TLSC_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_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

and no other ciphersuites

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches the reference identifier per RFC 6125 section 6.

FCS_TLSC_EXT.1.3 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.

FCS_TLSC_EXT.1.4 The TSF shall not present the Supported Elliptic Curves Extension in the Client Hello.

6.1.3 Identification and Authentication (FIA)

6.1.3.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within *3 to 5* unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password.*

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 a remote session using any authentication method that involves a password until action to unlock is taken by an Administrator; prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until an Administrator defined time period has elapsed.*

6.1.3.2 FIA_PMG_EXT.1 Password Management

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 *128* characters.

6.1.3.3 FIA_UIA_EXT.1 User Identification and Authentication

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.

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.

6.1.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism to perform local administrative user authentication.

6.1.3.5 FIA_UAU.7 Protected Authentication Feedback

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**.

6.1.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- 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 a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3.
- 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.*

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.

6.1.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

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.

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

6.1.4 Security Management (FMT)

6.1.4.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

FMT_MOF.1.1/ManualUpdate The TSF shall restrict the ability to enable the functions *to perform manual updates* to *Security Administrators*.

6.1.4.2 FMT_MOF.1/Functions Management of security functions behaviour

FMT_MOF.1.1/Functions The TSF shall restrict the ability to determine the behaviour of   modify the behaviour of the functions *transmission of audit data to an external IT entity* to *Security Administrators*.

6.1.4.3 FMT_MOF.1/Services Management of security functions behaviour

FMT_MOF.1.1/Services The TSF shall restrict the ability to [~~selection: determine the behaviour of, disable, enable, modify the behaviour of~~] **start and stop** ~~the functions~~ **services** to *Security Administrators*.

6.1.4.4 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1.1/CoreData The TSF shall restrict the ability to *manage* the *TSF data* to S*ecurity Administrators*.

Additional Application Note c: Where TSF data refers to audit data, TSFIs-related configuration data and user data. The word 'manage' includes but is not limited to create, initialize, view, change default, modify, delete, clear, and append. This SFR includes also the resetting of user passwords by the Security Administrator. The identifier 'CoreData' has been added here to separate this iteration of FMT_MTD.1 from FMT_MTD.1/CryptoKeys.

6.1.4.5 FMT_MTD.1/CryptoKeys Management of TSF data

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

6.1.4.6 FMT_SMF.1 Specification of Management Functions

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 start and stop services;*

- *Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);*
- *Ability to manage the cryptographic keys;*
- *Ability to configure thresholds for SSH rekeying;*
- *Ability to re-enable an Administrator account;*
- *Ability to set the time which is used for time-stamps;*
- *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;*

6.1.4.7 FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator*.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions:

- *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.

6.1.5 Protection of the TSF (FPT)

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

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

6.1.5.2 FPT_APW_EXT.1 Protection of Administrator Passwords

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

6.1.5.3 FPT_TST.1 TSF Testing

FPT_TST.1.1 The TSF shall run a suite of the following self-tests during initial start-up to demonstrate the correct operation of the TSF: *integrity of the firmware and software (software integrity check), the correct operation of cryptographic functions*.

FPT_TST.1.2 The TSF shall provide ~~authorized~~ **Administrator** users with the capability to verify the integrity of **software**~~[selection: [assignment: parts of TSF data], TSF data]~~.

FPT_TST.1.3 The TSF shall provide ~~authorized~~**Administrator** users with the capability to verify the integrity of TSF.

Additional Application Note d: The TSF integrity check is included in the software integrity check.

6.1.5.4 FPT_TUD_EXT.1 Trusted Update

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 the most recently installed version of the TOE firmware/software.

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.

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.

6.1.5.5 FPT_STM.1 Reliable Time Stamps

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

6.1.5.6 FPT_STM.2 Time Source

FPT_STM.2.1 The TSF shall allow the *Security Administrator* to *set the time*.

6.1.5.7 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: *when the event defined in FPT_TST.1 fails*.

6.1.6 TOE Access (FTA)

6.1.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions,

- terminate the session

after a Security Administrator-specified time period of inactivity.

6.1.6.2 FTA_SSL.3 TSF-initiated Termination

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

6.1.6.3 FTA_SSL.4 User-initiated Termination

FTA_SSL.4.1 The TSF shall allow ~~user-initiated~~ **Administrator-initiated** termination of the ~~user's~~ **Administrator's** own interactive session.

6.1.6.4 FTA_TAB.1 Default TOE Access Banners

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

6.1.7 Trusted path/channels (FTP)

6.1.7.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 The TSF shall **be capable of using TLS to** provide a communication channel between itself and ~~another trusted~~ **authorized IT** ~~product~~ **entities supporting the following capabilities: audit server** 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 ~~or disclosure~~ **of the channel data**.

FTP_ITC.1.2 The TSF shall permit the TSF to initiate communication via the trusted channel.

FTP_ ITC.1.3 The TSF shall initiate communication via the trusted channel for *Syslog server over TLS*.

6.1.7.2 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using SSH to** provide a communication path between itself and **authorized** remote ~~users~~ **Administrators** 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/Admin The TSF shall permit ~~[selection: *the TSF, local users, remote users*]~~ **remote Administrators** to initiate communication via the trusted path.

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

6.2 Assurance Security Requirements

The development and the evaluation of the TOE shall be done in accordance to the following security assurance requirements comply to EAL2 augmented with ALC_FLR.2:

Table 6-2 Security Assurance Requirements

Assurance Class	Assurance Components
Security Target (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives (ASE_OBJ.2)
	Derived security requirements (ASE_REQ.2)
	Security Problem Definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Security architecture description（ADV_ARC.1）
	Security-enforcing functional specification (ADV_FSP.2)

Assurance Class	Assurance Components
	Basic design（ADV_TDS.1）
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life cycle support (ALC)	Use of the CM system (ALC_CMC.2)
	Parts of the TOE CM coverage (ALC_CMS.2)
	Delivery procedures（ALC_DEL.1）
	Flaw reporting procedures（ALC_FLR.2）
Tests (ATE)	Evidence of coverage(ATE_COV.1)
	Functional testing(ATE_FUN.1)
	Independent testing – sample (ATE_IND.2)
Vulnerability assessment (AVA)	Vulnerability analysis (AVA_VAN.2)

6.3 Security Requirements Rationale

6.3.1 Necessity and sufficiency analysis

Table 6-3 SFRs / TOE Security Objectives coverage

SFR / TOE Security Objective	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PRO	O.PASSWORD_PROTE	O.SELF_TEST	O.BANNER
FAU_GEN.1						X				

SFR / TOE Security Objective	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PRO	O.PASSWORD_PROTE	O.SELF_TEST	O.BANNER
FAU_GEN.2						X				
FAU_STG.1						X				
FAU_STG.2						X				
FAU_STG.4						X				
FAU_STG.5						X				
FCS_CKM.1		X								
FCS_CKM.2		X								
FCS_CKM.6							X			
FCS_COP.1/DataEncryption		X								
FCS_COP.1/SigGen		X								
FCS_COP.1/Hash		X								
FCS_COP.1/KeyedHash		X								
FCS_SSHC_EXT.1			X							

SFR / TOE Security Objective	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PRO	O.PASSWORD_PROTE	O.SELF_TEST	O.BANNER
FCS_SSHS_EXT.1			X							
FCS_TLSC_EXT.1			X							
FIA_AFL.1								X		
FIA_PMG_EXT.1								X		
FIA_UIA_EXT.1	X									
FIA_UAU_EXT.2	X									
FIA_UAU.7								X		
FIA_X509_EXT.1/Rev			X		X					
FIA_X509_EXT.2			X		X					
FMT_MOF.1/ManualUpdate					X					
FMT_MOF.1/Functions	X					X				
FMT_MOF.1/Services	X									
FMT_MTD.1/CoreData	X									

SFR / TOE Security Objective	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PRO	O.PASSWORD_PROTE	O.SELF_TEST	O.BANNER
FMT_MTD.1/CryptoKeys							X			
FMT_SMF.1	X	X			X	X	X			
FMT_SMR.2	X									
FPT_SKP_EXT.1							X			
FPT_APW_EXT.1								X		
FPT_TUD_EXT.1					X					
FPT_STM.1						X				
FPT_STM.2						X				
FPT_FLS.1									X	
FTA_SSL_EXT.1	X									
FTA_SSL.3	X									
FTA_SSL.4	X									
FTA_TAB.1	X									X

SFR / TOE Security Objective	O.ADMIN_AUTH	O.STRONG_CRYPTO	O.TRUSTED_COMM	O.STRONG_AUTHENT	O.SECURE_UPDATES	O.ACTIVITY_AUDIT	O.CRYPTO_KEY_PRO	O.PASSWORD_PROTE	O.SELF_TEST	O.BANNER
FTP_ITC.1			X	X						
FTP_TRP.1	X		X	X						
FCS_RBG.1		X								
FCS_RBG.3		X								
FPT_TST.1									X	

6.3.2 Security Requirement Sufficiency

O.ADMIN_AUTH: The Administrator role is defined in **FMT_SMR.2** and along with **FMT_SMF.1**, they support this security objective by defining the TOE management capabilities available to administrators.

FMT_MTD.1/CoreData restricts the ability to manage the TSF to security administrators.

FMT_MOF.1/Functions restricts the ability to determine the feature of the transmission of audit data to an external IT entity. On the other hand, **FMT_MOF.1/Services** restricts the ability to disable or enable the services of the TOE to the security administrators.

FIA_UIA_EXT.1 supports this security objective by preventing any action before the identification and authentication process. **FTA_TAB.1** supports this security objective by showing an advisory notice and consent warning message is the only action allowed before the identification and authentication process occurs.

The administrators' authentication process is defined in **FIA_UAU_EXT.2** and consists on providing a local password-based authentication mechanism to perform local administrative user authentication.

This security objective is intended to perform the closure of authentication sessions after a defined period of inactivity. **FTA_SSL_EXT.1** ensures this feature for local sessions, **FTA_SSL.4** for all interactive sessions, and by **FTA_SSL.3** for remote sessions.

Another goal of this security objective is that the authentication of the TOE using passwords was through a secure communication channel. This task is covered by **FTP_TRP.1**.

O.STRONG_CRYPTO: This security objective is supported by the following security requirements, that provide use of robust cryptographic algorithms, compliant to industry approved standards:

- Requirements for key generation and key distribution are set in **FCS_CKM.1** and **FCS_CKM.2**, respectively.

- Requirements for use of cryptographic schemes are set in **FCS_COP.1/DataEncryption**, **FCS_COP.1/SigGen**, **FCS_COP.1/Hash**, and **FCS_COP.1/KeyedHash**.

- Requirements for random bit generation to support key generation and secure protocols are set in **FCS_RBG.1** and **FCS_RBG.3**.

- **FMT_SMF.1** is intended to establish the management of cryptographic functions by a security administrator.

O.TRUSTED_COMM: This security objective is intended to implement secure channels that use standardized tunneling protocols to protect the critical network traffic. The following security requirements support this security objective:

- **FTP_ITC.1** supports this security objective by providing a communication channel between itself and authorized IT entities by using the TLS protocol. **FTP_TRP.1** is able to provide this channel by using SSH.

- **FCS_SSHC_EXT.1**, **FCS_SSHS_EXT.1**, **FCS_TLSC_EXT.1** and support this security objective for the use of secure communication protocols by using SSH Client, SSH Server Protocol, and TLS Client Protocols.

- Requirements for the use of public-key certificates to support secure protocols are supported by **FIA_X509_EXT.1/Rev**, **FIA_X509_EXT.2**.

All the above-mentioned security requirements support this security objective by providing communication security in terms of confidentiality, integrity, and protection.

O.STRONG_AUTHENTICATION_ENDPOINT: **FTP_ITC.1** and **FTP_TRP.1** support this security objective by providing assured identification of its endpoints (using TLS and SSH respectively) and protection of the channel data from disclosure and detection of modification of the channel data.

O.SECURE_UPDATES: **FPT_TUD_EXT.1** supports this security objective by allowing administrators to query the current TOE version of the TOE firmware/software and manually initiate the installation of updates, that need to be authenticated using digital signature before installing.

FIA_X509_EXT.1/Rev and **FIA_X509_EXT.2** support this security objective by defining a set of rules to validate the authenticity of the certificate using secure and robust cryptographic algorithms.

FMT_SMF.1 supports this security objective by providing a set of management functions, which includes the ability to update the TOE and to verify the updates using digital signature capabilities. **FMT_MOF.1/ManualUpdate** likewise supports this security objective by restricting the ability of this feature to security administrators.

O.ACTIVITY_AUDIT: **FAU_GEN.1** supports this security objective by generating audit records of a set of auditable events. Each audit event will be marked with precise timestamps, associated subject identity, and the outcome (success or failure) of the event. **FAU_GEN.2** supports this security objective by providing a relationship between the audit event and the identity of the user that causes the event. **FPT_STM.1** and **FPT_STM.2** is intended to provide reliable time stamps for the own TOE's use.

FAU_STG.2 supports this security target by protecting the stored audit records from unauthorized deletion. This security requirement is also intended to prevent unauthorized modifications to the stored audit records in the audit trail.

FAU_STG.1 is intended to provide the ability to transmit the generated audit data to an external IT entity using a trusted channel by using secure communication protocols.

FAU_STG.5 provides a mechanism to overwrite the oldest log information always when the local storage space for audit data is full.

FAU_STG.4 provides a mechanism to generate a warning to inform the TOE's administrator if the audit trail exceeds the local audit trail storage capacity. This mechanism is intended to deal with the potential loss of locally stored audit records.

FMT_SMF.1 supports this security objective by providing a set of management functions, which includes the ability to configure audit behavior.

FMT_MOF.1/Functions likewise supports this security objective by restricting the ability to enable the transmission of audit data to an external IT entity to a security administrator.

O.CRYPTO_KEY_PROTECTION: **FPT_SKP_EXT.1** supports the protection of secret/private keys against compromise.

FCS_CKM.6 provides a mechanism to perform the key's destruction in a secure way that prevents key recovery from residual information.

FMT_SMF.1 supports this security objective by providing a set of management functions, which includes the ability to configure thresholds for SSH rekeying. **FMT_MTD.1/CryptoKeys** is intended to restrict the ability to manage cryptographic keys to security administrators.

O.PASSWORD_PROTECTION: **FIA_PMG_EXT.1** supports this security objective by providing the ability to establish password management capabilities based on password complexity (composed of any combination of upper and lower case letters numbers, and special characters) and minimum password length restriction.

Protection of password entry by providing obscured feedback is specified in **FIA_UAU.7**

FIA_AFL.1 supports this security objective by providing the ability to detect unsuccessful authentication attempts when administrators are attempting to authenticate remotely. This security requirement meets this security objective by locking the authentication of the user, which reaches the defined number of authentication attempts established.

FPT_APW_EXT.1 prevents the plaintext storage of passwords. Therefore, this security requirement also prevents the reading of plaintext passwords.

O.SELF_TEST: **FPT_TST.1** provides the necessary mechanisms to support this security objective by run a suite of tests during the initial start-up of the TOE to demonstrate its correct operation.

FPT_FLS.1 preserve a secure state when the event defined in FPT_TST.1 fails.

O.BANNER: This security objective is supported by **FTA_TAB.1** which requires that the TOE displays an advisory notice and consent about the use of the TOE to administrators before establishing an administrative user session.

6.3.3 SFR Dependency Rationale

The following table lists all SFRs contained in ST together with the classification whether they are mandatory, optional or selection-based, indicates which are included in this ST and provides a dependency rationale. Justifications for any unsupported dependencies will be given in the table as well.

Table 6-4 Dependency rationale for SFRs

Requirement	Dependencies	Satisfied by
Mandatory Requirements		
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1; FIA_UID.1	FAU_GEN.1; Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification timing
FAU_STG.1	FAU_GEN.1; FTP_ITC.1	FAU_GEN.1; FTP_ITC.1
FAU_STG.2	FAU_GEN.1	FAU_GEN.1
FAU_STG.4	FAU_STG.2	FAU_STG.2
FAU_STG.5	FAU_STG.2 FAU_GEN.1	FAU_STG.2 FAU_GEN.1

FCS_CKM.1	FCS_CKM.2 or FC_CKM.5 or FCS_COP.1;
	FCS_RBG.1 or FCS_RNG.1;
	FCS_CKM.6
		FCS_CKM.2;
		FCS_RBG.1
		FCS_CKM.6

FCS_CKM.2	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5;	FCS_CKM.1
FCS_CKM.6	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5	FCS_CKM.1
FCS_COP.1/DataEncryption	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5; FCS_CKM.6	FCS_CKM.1 FCS_CKM.6
FCS_COP.1/SigGen	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5; FCS_CKM.6	FCS_CKM.1 FCS_CKM.6
FCS_COP.1/Hash	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5; FCS_CKM.6	Unsupported Dependencies: This SFR specifies keyless hashing operations, so initialisation and access of keys are not relevant
FCS_COP.1/KeyedHash	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 or FCS_CKM.5; FCS_CKM.6	FCS_CKM.1 FCS_CKM.6
FCS_RBG.1	FCS_RBG.2 or FCS_RBG.3 FPT_FLS.1 FPT_TST.1	FCS_RBG.3 FPT_TST.1 FPT_FLS.1
FCS_RBG.3	FCS_RBG.1	FCS_RBG.1
FIA_AFL.1	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FIA_PMG_EXT.1	None	N/A
FIA_UIA_EXT.1	FTA_TAB.1	FTA_TAB.1
FIA_UAU_EXT.2	None	N/A
FIA_UAU.7	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator authentication
FMT_MOF.1/ManualUpdate	FMT_SMR.1; FMT_SMF.1	FMT_SMR.2; FMT_SMF.1
FMT_MTD.1/CoreData	FMT_SMR.1;	FMT_SMR.2;

Requirement	Dependencies	Satisfied by
	FMT_SMF.1	FMT_SMF.1
FMT_SMF.1	None	N/A
FMT_SMR.2	FIA_UID.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification
FPT_SKP_EXT.1	None	N/A
FPT_APW_EXT.1	None	N/A
FPT_TST.1	None	N/A
FPT_TUD_EXT.1	FCS_COP.1/SigGen or FCS_COP.1/Hash	FCS_COP.1/SigGen and FCS_COP.1/Hash
FPT_STM.1	None	N/A
FPT_STM.2	FPT_STM.1 FMT_SMR.1	FPT_STM.1 FMT_SMR.2
FPT_FLS.1	None	N/A
FTA_SSL_EXT.1	FIA_UAU.1	Satisfied by FIA_UIA_EXT.1, which specifies the relevant Administrator identification
FTA_SSL.3	FMT_SMR.1	FMT_SMR.2
FTA_SSL.4	None	N/A
FTA_TAB.1	None	N/A
FTP_ITC.1	None	N/A
FTP_TRP.1/Admin	None	N/A

FCS_SSHC_EXT.1	FCS_CKM.1;
	FCS_CKM.2;
	FCS_COP.1/DataEncryption;
	FCS_COP.1/SigGen;
	FCS_COP.1/Hash;
	FCS_COP.1/KeyedHash;
	FCS_RBG.1:
		FCS_CKM.1;
		FCS_CKM.2;
		FCS_COP.1/DataEncryption;
		FCS_COP.1/SigGen;
		FCS_COP.1/Hash;
		FCS_COP.1/KeyedHash;
		FCS_RBG.1:

FCS_SSHS_EXT.1	FCS_CKM.1;
	FCS_CKM.2;
	FCS_COP.1/DataEncryption;
	FCS_COP.1/SigGen;
	FCS_COP.1/Hash;
	FCS_COP.1/KeyedHash;
	FCS_RBG.1:
		FCS_CKM.1;
		FCS_CKM.2;
		FCS_COP.1/DataEncryption;
		FCS_COP.1/SigGen;
		FCS_COP.1/Hash;
		FCS_COP.1/KeyedHash;
		FCS_RBG.1:

FCS_TLSC_EXT.1	FCS_CKM.1;
	FCS_CKM.2;
	FCS_COP.1/DataEncryption;
	FCS_COP.1/SigGen;
	FCS_COP.1/Hash;
	FCS_COP.1/KeyedHash;
	FCS_RBG.1:
		FCS_CKM.1;
		FCS_CKM.2;
		FCS_COP.1/DataEncryption;
		FCS_COP.1/SigGen;
		FCS_COP.1/Hash;
		FCS_COP.1/KeyedHash;
		FCS_RBG.1:

FIA_X509_EXT.1/Rev	FIA_X509_EXT.2;	FIA_X509_EXT.2;
FIA_X509_EXT.2	FIA_X509_EXT.1;	FIA_X509_EXT.1/Rev;
FMT_MOF.1/Services	FMT_SMR.1; FMT_SMF.1	FMT_SMR.2; FMT_SMF.1
FMT_MOF.1/Functions	FMT_SMR.1; FMT_SMF.1	FMT_SMR.2; FMT_SMF.1

Requirement	Dependencies	Satisfied by
FMT_MTD.1/CryptoKeys	FMT_SMR.1; FMT_SMF.1	FMT_SMR.2; FMT_SMF.1

6.3.4 SAR Rationale

The TOE claims compliance to EAL2 augmented with ALC_FLR.2 and conforms the assurance security requirements:ASE_CCL.1 ， ASE_ECD.1, ASE_INT.1, ASE_OBJ.2, ASE_REQ.2, ASE_SPD.1, ASE_TSS.1, ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1, ALC_CMC.2, ALC_CMS.2, ALC_DEL.1, ALC_FLR.2，ATE_COV.1, ATE_FUN.1, ATE_IND.2, AVA_VAN.2 which are defined in [CC40R1P5] . The assurance security requirements indicate that the product is methodically designed, tested, and reviewed.

6.3.5 SAR Dependency Rationale

6.3.5.1 Table of SAR dependencies

Table 6-5 SAR dependencies

SAR	Required	Fulfilled	Missing
ASE_CCL.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.2 (hierarchically above ASE_REQ.1)	None
ASE_ECD.1	None	None	None
ASE_INT.1	None	None	None
ASE_OBJ.2	ASE_SPD.1	ASE_SPD.1	None
ASE_REQ.2	ASE_OBJ.2, ASE_ECD.1	ASE_OBJ.2, ASE_ECD.1	None
ASE_TSS.1	ASE_INT.1, ASE_REQ.1, ADV_FSP.1	ASE_INT.1, ASE_REQ.2 (hierarchically above ASE_REQ.1), ADV_FSP.2 (hierarchically above ADV_FSP.1)	None
ALC_CMC.2	ALC_CMS.1	ALC_CMS.2 (hierarchically above ALC_CMS.1)	None
ALC_CMS.2	None	None	None
ALC_FLR.2	None	None	None
ADV_FSP.2	ADV_TDS.1	ADV_TDS.1	None

SAR	Required	Fulfilled	Missing
AGD_OPE.1	ADV_FSP.1	ADV_FSP.2 (hierarchically above ADV_FSP.1)	None
AGD_PRE.1	None	None	None
ATE_IND.2	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1	None
AVA_VAN.2	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1	ADV_ARC.1, ADV_FSP.2, ADV_TDS.1, AGD_OPE.1, AGD_PRE.1	None
ASE_SPD.1	None	None	None
ALC_DEL.1	None	None	None
ADV_ARC.1	ADV_FSP.1, ADV_TDS.1	ADV_FSP.2 (hierarchically above ADV_FSP.1), ADV_TDS.1	None
ADV_TDS.1	ADV_FSP.2	ADV_FSP.2	None
ATE_COV.1	ADV_FSP.2, ATE_FUN.1	ADV_FSP.2, ATE_FUN.1	None
ATE_FUN.1	ATE_COV.1	ATE_COV.1	None

7 TOE Summary Specification

7.1 Security Audit (FAU)

7.1.1 FAU_GEN.1 Audit data generation

The TOE generates an audit record whenever an audited event occurs. The types of events that cause audit records to be generated include identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, “0 Security Functional Requirements and Auditable Events”). Each of the events specified in the audit record is in enough detail to identify the user for which the event is associated (e.g. user identity, MAC address, IP address), when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred.

The audit trail consists of the individual audit records; one audit record for each event that occurred. The audit record contains a lot of information, such as the type of event that occurred, and two percent signs (%%), which follows the device name. As noted above, the information includes at least all of the required information. Additional information can be configured and included if desired.

Administrators have the ability to execute CLI commands to generate/import of/delete cryptographic keys, each command will generate a log and will be stored in log file. The log contains the user name and IP address. The log does not contain the generated key information.

7.1.2 FAU_GEN.2 User identity association

Each auditable event is associated with the user that triggered the event and as a result, they are traceable to a specific user. For example, a human user, user identity or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented.

The security log of user account management should include user name. Other types of security log have other rules about the information.

7.1.3 FAU_STG.1 Audit data Storage location

The TOE supports the export of syslog records to a specified, external syslog server. The TOE protects communications with an external syslog server via TLS. The TOE stores audit records on flash memory whenever it is connected with syslog server or not. The transmission of audit information to an external syslog server can be done in real-time.

The size of an information file is configurable by the administrator with value 4M/8M/16M/32M bytes. The default maximum size of each information file is 8 MB. When the size of an information file exceeds the configured maximum size, the information file is compressed into a smaller file in standard log_slot ID_time.log.zip format. The maximum quantity of compressed files is configurable by the administrator with a value ranging from 3 to 500. A maximum of 200 files can be stored on a device by default. The unauthorized users are disallowed to handle the audit records.

7.1.4 FAU_STG.2 Protected audit data storage

Only authorized administrators can monitor the logfile record, and operate the log files. The unauthorized users have no access rights to perform these actions. All actions of the authorized administrators will be logged. The default maximum size is 8 MB for a common log file, 4 MB for a security log file, and 4 MB for an operation log file.

7.1.5 FAU_STG.4 Action in case of possible audit data loss

If the log files have already occupied more than 80% of the total audit storage in CF card, or delete the old log files after saving them to the other storage device, an event will be generated and sent to management server to notice the clients of the warning information.

If the number of compressed log files generated in the system exceeded 80% of the maximum number of compressed files, an event will also be generated to notice NMS the warning information.

If the number of recorded compressed files reach the maximum number that the security administrator has configured, or the storage with audit events reach the configured storage size, another event will be generated to notice NMS.

7.1.6 FAU_STG.5 Prevention of audit data loss

The logs are saved to flash memory so records can’t be lost in case of failures or restarts. The log buffer is circular, so newer messages overwrite older messages after the buffer is full. Administrators are instructed to monitor the log buffer using the show logging privileged CLI command to view the audit records. The first message displayed is the oldest message in the buffer. There are other associated commands to clear the buffer, to reset log buffer, etc. The size of the log buffer can be configured by users with sufficient privileges.

When the local audit data stored in flash memory exceeds the maximum allowed size of log file storage, it will always overwrite the oldest log information.

7.2 Cryptographic Support (FCS)

7.2.1 FCS_CKM.1 Cryptographic Key Generation

The TOE supports

1) FFC schemes using cryptographic key sizes of 3072-bit or greater that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.1. The FFC schemes is used for TLS.
2) 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. The ECC schemes is used for SSH.

7.2.2 FCS_CKM.2 Cryptographic Key Distribution

The TOE supports Elliptic curve-based key distribution schemes that meet the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”. The key size supports at least 256 bits. The TOE establishes a SSH connection based on Elliptic curve-based key establishment schemes. The Elliptic curve-based key establishment schemes is used when the TOE establishes SSH connection.

The TOE supports Finite field-based key distribution schemes that meet the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”. The key size supports at least 3072 bits. The Finite field-based key establishment schemes is used when the TOE establishes TLS connection.

Scheme	SFR	Service
DHE	FCS_TLSC_EXT.1	Audit
ECDH	FCS_SSHS_EXT.1	Administration
ECDH	FCS_SSHC_EXT.1	Administration

7.2.3 FCS_CKM.6 Timing and event of Cryptographic Key Destruction

The private key stored in SDRAM is used to verify the integrity of the certificate. After the device is restarted, the private key is imported into the SDRAM from the CF card. The private key is encrypted by the AES key and stored on the CF card. Users and administrator cannot access the private key that is stored in the SDRAM and CF card.

Table 7-1 Key Destructions

Name	**Description of Key**	**Storage**	**Key destruction method**
SSH/TLS session key	The key is used for encrypting/decrypting the traffic in a secure connection.	SDRAM (plaintext)	Automatically after session terminated. Overwritten with: zeroes.
FFC key	The key is used for key establishment.	SDRAM (plaintext)	Automatically after completion of use of the key. Overwritten with: zeroes.
TLS private key	The key is used for signature and authentication.	CF card (AES256 cipher)	Overwritten by a command. Overwritten with: a new value of the key.
ECC key pair	The ECC key pair is used for digital signature. The ECC host key pair is imported into the SDRAM from the CF card, which is the ECC key pair.	SDRAM (plaintext)	Automatically after completion of use of the key. Overwritten with: zeroes.
ECC host key pair	Using command generate a ECC host key pair.	CF card (AES256 cipher)	Zeroized using “ecc local-key-pair destroy” command. Overwritten with: zeroes.
AES key	The AES key is generated by root key. AES key is used to encrypt ECC host key pair and TLS private key. Note: The root key is generated by root key material. The root key material is saved many places, for example: code.	SDRAM (plaintext)	The AES key is stored in the SDRAM temporarily and destroyed after used. Overwritten with: zeroes.

7.2.4 FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/ Decryption)

The TOE provides symmetric encryption and decryption capabilities using AES as specified in ISO 18033-3 supporting the following modes:

- CTR mode as specified in ISO 10116.
- GCM mode as specified in ISO 19772.

The TOE uses AES in the following protocols:

- SSH: CTR mode with key sizes of 128 bits and 256 bits. GCM mode with key sizes of 128 bits and 256 bits.
- TLS: GCM mode with key sizes of 128 bits and 256 bits.

7.2.5 FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)

The TOE provides cryptographic signature services using RSA with key sizes between 3072 and 4096 bits as specified in FIPS PUB 186-4 “Digital Signature Standard (DSS)”.

- The RSA with key size of 3072 to 4096 is used for signature generation and verification of TLS.

The TOE provides cryptographic signature services using ECDSA with key sizes between 256 bits, 384 bits and 521 bits as specified in FIPS PUB 186-4 “Digital Signature Standard (DSS)”.

- The ECDSA with key size 256 bits, 384 bits and 521 bits is used for signature generation and verification of SSH.

The TOE provides cryptographic signature services using RSA with key sizes 4096 bits as specified in FIPS PUB 186-4 “Digital Signature Standard (DSS)”.

- The RSA with key size of 4096 is used for signature verification of Secure Update.

7.2.6 FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

The TOE provides cryptographic hashing services using SHA-256, and SHA-384 as specified in FIPS Pub 180-3 “Secure Hash Standard.”, it also meets the ISO/IEC 10118-3:2004.

The association of the hash function with other TSF cryptographic functions is:

Table 7-2 Usage of Hash Algorithm

Cryptographic Functions	**Hash Function**
HMAC-SHA-256	SHA-256
TLS Digital signature verification	SHA-256 SHA-384
SSH Digital signature verification	SHA-256
Hash_DRBG	SHA-256

7.2.7 FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

The TOE provides cryptographic keyed hash services using HMAC-SHA-256 according to RFC2104: HMAC, it also complies with the ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.

Table 7-3 Specification of Keyed Hash Algorithm

HMAC function	**Key length (bits)**	**Hash function**	**Block size (bits)**	**Output MAC length (bits)**
HMAC-SHA-256	256	SHA-256	512	256

7.2.8 FCS_RBG.1 Random Bit Generation

The TOE implements a deterministic random bit generator (DRBG) which is conformant to [ISO18031] using the DRBG mechanism Hash_DRBG as specified in [ISO18031], chap. C.2.2.

The entropy source is non-physical TRNG and based on software(internal noise source). Random numbers from the internal noise source are only used for seeding the DRBG.

DRBG parameters are predefined for the TOE and cannot be modified. Prediction resistance is disabled for the DRBG in the TOE.

7.2.9 FCS_RBG.3 Random Bit Generation(internal seeding – single source)

The entropy source is non-physical TRNG and based on software(internal noise source).Before generating random bits as the cryptographic key, the TOE sets a new seed using at least 256 of entropy.

7.2.10 FCS_SSHC_EXT.1 SSH Client Protocol

7.2.10.1 FCS_SSHC_EXT.1.1

The TOE implements the SSH protocol that comply with RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668.

7.2.10.2 FCS_SSHC_EXT.1.2

Both public key and password authentication modes are supported by SSH client function. Users can use any or both of those modes to login external SSH server successfully.

The supported public key algorithms for authentication include ECC with cryptographic key size of 256-bit, 384-bit and 521-bit. These public key algorithm conforms to FCS_SSHC_EXT.1.5.

7.2.10.3 FCS_SSHC_EXT.1.3

After the SSH connection is established, the TOE will drop packets greater than 256KB. During the connection establishment process, the TOE will drop packets greater than 35,000 bytes.

7.2.10.4 FCS_SSHC_EXT.1.4

The SSH client supports the encryption algorithms of aes128-ctr and aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com and aes256-gcm@openssh.com.

When SSH Client establishes a connection, it will send a list of encryption algorithms to SSH server. SSH Server will check each algorithm in the list one by one. If it finds one algorithm in the list that is also supported by it, this algorithm will be chosen as the encryption algorithm between client and server. If no algorithm in the list is supported by SSH server, the connection will be terminated.

After the encryption algorithm is selected, Server and Client will create a random number and exchange. Client and Server will use their own random number to create an encryption key.

Then, the SSH Client will use its own encryption key to encrypt each packet, and use SSH Server's encryption key to decrypt each packet.

7.2.10.5 FCS_SSHC_EXT.1.5

SSH client function supports the public key algorithm of ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521.

Before SSHC and SSHS build a connection, they both need to configure a local key-pair that is used for authentication. In Huawei's device, this local key-pair is used for SSH server and SSH client.

The first step to be done/taken when a Client authenticates a Server, is to consult public key algorithms. Client will send a list of public key algorithms to SSH server. SSH Server will check each algorithm in the list one by one. If it finds one algorithm in the list that is also supported by it, this algorithm will be chosen as the public key algorithm between client and server. If no algorithm in the list is supported by SSH server, the connection will be terminated.

7.2.10.6 FCS_SSHC_EXT.1.6

SSH client supports the data integrity algorithms of hmac-sha2-256, AEAD_AES_128_GCM and AEAD_AES_256_GCM.

AEAD_AES_128_GCM and AEAD_AES_256_GCM will be selected as MAC algorithms when the same algorithm is being used as the encryption algorithm. When aes*-gcm@openssh.com is negotiated as the encryption algorithm, the negotiated MAC might be decoded as "implicit".

7.2.10.7 FCS_SSHC_EXT.1.7

SSH client supports the following key exchange algorithm of ecdh-sha2-nistp256.

7.2.10.8 FCS_SSHC_EXT.1.8

The SSH connection will be rekeyed after one hour of session time or one gigabyte of transmitted data using that key whichever comes first.

The SSH allows either side to force another run of the key-exchange phase, changing the encryption and integrity keys for the session. The idea is to do this periodically, after one hour of session time or one gigabyte of transmitted data using that key whichever comes first.

7.2.10.9 FCS_SSHC_EXT.1.9

The SSH client will authenticate the identity of the SSH server using a local database associating each host name with its corresponding public key.

7.2.11 FCS_SSHS_EXT.1 SSH Server Protocol

7.2.11.1 FCS_SSHS_EXT.1.1

The TOE implements the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668.

7.2.11.2 FCS_SSHS_EXT.1.2

Both public key and password authentication modes are supported by SSH server function. The TOE implements the public key algorithms of ecdsa-sha2-nistp256, ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521.

SSH users can be authenticated in eight modes: ECC, password, password-ECC, and All (any authentication mode of ECC or password is allowed with “ALL” mode). The SSH user that is created by administrators shall be configured to one of the modes. Then the external SSH client can login SSH server successfully via the configured SSH user and authentication mode.

7.2.11.3 FCS_SSHS_EXT.1.3

After the SSH connection is established, the TOE will drop packets greater than 256KB. During the connection establishment process, the TOE will drop packets greater than 35,000 bytes.

7.2.11.4 FCS_SSHS_EXT.1.4

SSH server function supports the encryption algorithms of aes128-ctr, aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com and aes256-gcm@openssh.com.

When SSH Client establishes a connection, it will send a list of encryption algorithms to SSH server. SSH Server will check each algorithm in the list one by one. If it finds one algorithm in the list that is also supported by it, this algorithm will be chosen as the encryption algorithm between client and server. If no algorithm in the list is supported by SSH server, the connection will be terminated.

After the encryption algorithm is selected, Server and Client will create a random number and exchange. Client and Server will use their own random number to create an encryption key.

Then, the SSH server will use its own encryption key to encrypt each packet, and use SSH client’s encryption key to decrypt each packet.

7.2.11.5 FCS_SSHS_EXT.1.5

SSH server function supports the public key algorithm of ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521.

Before SSHC and SSHS build a connection, they both need to configure a local key-pair that is used for authentication. In Huawei's device, this local key-pair is used for SSH server and SSH client.

The first step to be done/taken when a Client authenticates a Server, is to consult public key algorithms. Client will send a list of public key algorithms to SSH server. SSH Server will check each algorithm in the list one by one. If it finds one algorithm in the list that is also supported by it, this algorithm will be chosen as the public key algorithm between client and server. If no algorithm in the list is supported by SSH server, the connection will be terminated.

7.2.11.6 FCS_SSHS_EXT.1.6

SSH server function supports the data integrity algorithms of hmac-sha2-256 , AEAD_AES_128_GCM and AEAD_AES_256_GCM.

AEAD_AES_128_GCM and AEAD_AES_256_GCM will be selected as MAC algorithms when the same algorithm is being used as the encryption algorithm. When aes*-gcm@openssh.com is negotiated as the encryption algorithm, the negotiated MAC might be decoded as "implicit".

7.2.11.7 FCS_SSHS_EXT.1.7

SSH server supports the following key exchange algorithm: ecdh-sha2-nistp256 and no other methods.

7.2.11.8 FCS_SSHS_EXT.1.8

The SSH connection will be rekeyed after one hour of session time or one gigabyte of transmitted data using that key whichever comes first.

The SSH allows either side to force another run of the key-exchange phase, changing the encryption and integrity keys for the session. The idea is to do this periodically, after one hour of session time or one gigabyte of transmitted data using that key whichever comes first.

7.2.12 FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication

7.2.12.1 FCS_TLSC_EXT.1.1

The TLS client supports the following ciphersuites:

- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

7.2.12.2 FCS_TLSC_EXT.1.2

The reference identifier is established by the user and by an application (a parameter of an API). Based on a singular reference identifier's source domain and application service type (e.g. syslog), the client establishes all reference identifiers including DNS names(case-insensitive) for the Subject Alternative Name field. The client then compares this list of all acceptable reference identifiers to the presented identifiers in the TLS server's certificate.

The TOE doesn't support certificate pinning and use of wildcards in digital certificates. The TOE doesn't support to use IP addresses in digital certificates.

7.2.12.3 FCS_TLSC_EXT.1.3

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.

7.2.12.4 FCS_TLSC_EXT.1.4

TLS don't support EC Extension in the Client Hello

7.3 Identification and Authentication (FIA)

7.3.1 FIA_AFL.1 Authentication Failure Management

Administrators can configure TOE's session lock time and number of unsuccessful authentication attempts from 3 to 5. When the defined number of unsuccessful authentication attempts has been met, the TOE will prevent the offending remote Administrator from successfully authenticating before the lock time or unlock is taken by a local Administrator or prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed.

To ensure account and password security of the administrators, the account locking function is enabled for administrators who fail remote authentication.

When an account logs in to the device within a specified period and the password is incorrect, the number of login failures of the account is recorded. When the number of login failures of the account reaches the upper limit (3 by default), the account is locked (the default locking duration is 5 minutes). After a certain period, the account is unlocked.

7.3.2 FIA_PMG_EXT.1 Password Management

The TOE supports the local definition of users with corresponding passwords which are used for security administrators' authentication of local or remote administration connections. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (not including spaces, question marks or single quotation mark). Minimum password length is settable by the Authorized Administrator, and passwords 8 characters or greater (maximum 128 characters) are supported. Password composition rules specifying the types and number of required characters that comprise the password are settable by the Authorized Administrator. Passwords have a maximum lifetime, configurable by the Authorized Administrator.

7.3.3 FIA_UIA_EXT.1 User Identification and Authentication

The TOE requires all users to be successfully identified and authenticated before allowing execution of any TSF mediated action except display of the banner and ICMP echo service.

Success-logon includes user-name, connect-type, IP-address, authentication-status, and so on.

The TOE supports user login over console or remote interface. Any login method needs authentication before a successful logon.

Local access is achieved by console port. Local authentication supports password-based authentication.

Remote access is achieved by SSH. It also supports associated identity authentication of password and public-key. Users can also login with any of the identity authentication modes of password, and ECC when their login mode is configured to be 'ALL'.

7.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

The TOE can be configured to require local authentication or remote authentication as defined in the authentication policy for interactive (human) users.

The policy for interactive (human) users (Administrators) can be authenticated to the local user database, or have redirection to a remote authentication server. Interfaces can be configured to try one or more remote authentication servers, and then fail back to the local user database if the remote authentication servers are inaccessible.

If the interactive (human) users (Administrators) password is expired, the user is required to create a new password after correctly entering the expired password.

7.3.5 FIA_UAU.7 Protected Authentication Feedback

When a user inputs their password at the local console, the console will not display the input so that the user password is obscured. For remote session authentication, the TOE does not echo any characters as they are entered. The TOE does not provide any additional information to the user that would give any indication about the authentication data.

7.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

The TOE supports the verification of the certificate and the certificate path by the rules specified in RFC 5280, using algorithm RSA.

The TOE supports the verification of the revocation status by CRLs as specified in RFC 5280.When the client receives TLS Handshake's Server Certificate message, the client will check validation of the certificates and certificate revocation list. When an administrator imports a certificate, the TOE will check certificate integrity and validation of the certificates.

The TOE validates 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 validates 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.

The TOE does not implements OCSP, so the id-kp-9 is not supported by the TOE. The TOE only acts as a client which only receives Server certificates, so the id-kp-2 is not supported by the TOE. The TOE does not use X509 certificates for the TOE updating, so the id-kp-3 is not supported by the TOE.

7.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

The certificate used by TLS authentication is sent by the TLS server. The CRL should be loaded for certificate validation.

The TOE will send a security log when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. TLS only supports RSA certificate.

The check of validity of the certificates takes place at the authentication of the TLS connection. When the certificate is valid, we can trust the peer identity and use the certificate to verify the integrity of the message.

TOE chooses a certificate which was configured by CLI for services (such as Syslog).

When the TSF cannot establish a connection to determine the validity of a certificate; the TSF shall not accept the certificate when all other checks pass in FIA_X509_EXT.1.

7.4 Security management (FMT)

7.4.1 FMT_MOF.1/ManualUpdate

The TSF shall restrict the ability to enable the functions to perform manual updates to Security Administrators.

Only administrators have the right to create or delete local users. When changing the local user privilege level, the configured new level of the local user cannot be higher than that of the login-in user. This way, no user except administrators can change another user to be at the privilege level of administrator. And only administrators have the ability to perform manual update. So the manual update is restricted to administrators. The TOE uses groups to organize users. Different kinds of users are in different groups and every group has a specific level that identifies its roles and scope of rights.

7.4.2 FMT_MOF.1/Functions Management of security functions behavior

Only administrators have the privilege to choose a trusted channel for transmitting the audit data to an external IT entity. A user with no administrator privileges cannot configure and enable the info-center function.

7.4.3 FMT_MOF.1/Services Management of security functions behaviour

Only administrators have ability to enable and disable the functions and services, the other users are disallowed to do it. The functions and services include “info-center loghost”, “undo info-center loghost”, “info-center loghost source”, “undo info-center loghost source” , “sftp server enable”, “undo sftp server enable”, for more please refer to the AGD.

7.4.4 FMT_MTD.1/CoreData Management of TSF Data

Only administrators have privilege to manage the TSF data, the other users are disallowed to do it.

The TOE provides the ability for authorized administrators to access TOE data, such as audit data and configuration data. Each of the predefined and administratively configured users have different rights to access the TOE data.

The access control mechanisms of the TOE are based on hierarchical access levels where a user level is associated with every user and terminal on the one hand, and a command level is associated with every command. Only if the user level is equal or higher to a specific command, authorizes the user to execute this command. Management of security function is realized through commands. So for every management function sufficient user level is required for the user to be able to execute the corresponding command.

7.4.5 FMT_MTD.1/CryptoKeys Management of TSF data

Only administrators have the right to delete and/or import the cryptographic keys (such as ecc key, rsa key), the other users are disallowed to do this.

7.4.6 FMT_SMF.1 Specification of Management Functions

The TOE provides all the capabilities necessary to securely manage the TOE. The administrative user can connect to the TOE using the CLI to perform these functions via SSH encrypted session.

The management functionality provided by the TOE includes the following administrative functions:

- Ability to manage the TOE locally as well as 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 start and stop services;
- Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);
- Ability to manage the cryptographic keys;
- Ability to configure thresholds for SSH rekeying;
- Ability to re-enable an Administrator account;
- Ability to set the time which is used for time-stamps;
- 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;

7.4.7 FMT_SMR.2 Restrictions on security roles

A Security Administrator is able to administer the TOE through the local console or through a remote mechanism.

An administrator can create, delete and modify the other users and endow them with a proper level of rights according to the users' roles. The TOE uses groups to organize users. Different kinds of users are in different groups and every group has a specific level that identifies its roles and scope of rights. Every user in one group has the same scope of rights that the group owns. The TOE has 4 default user groups: manage-ug, system-ug, monitor-ug, and visitor-ug.

7.5 Protection of the TSF (FPT)

7.5.1 FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric keys)

The TOE stores all symmetric keys, and private keys in SDRAM that can't be read, copied or extracted by administrators; hence no interface access.

7.5.2 FPT_APW_EXT.1 Protection of Administrator Passwords

The administrator passwords are stored to a configuration file in cryptographic form hashed with salt by SHA-256, including username passwords, authentication passwords, console and virtual terminal line access passwords.

In this manner, the TOE ensures that plaintext user passwords will not be disclosed to anyone through normal interfaces including administrators.

7.5.3 FPT_TST.1 TSF testing

The TSF runs a suite of self-tests during initial start-up to demonstrate the correct operation of the TSF, including software integration verification by integrity check and the correct operation of cryptographic functions. During initial start-up, software integrity is checked at first. If integrity check is failed, the start-up procedure will stop. Then the correct operation of cryptographic functions is tested. If this testing fails, the start-up procedure will also stop.

Self-tests include cryptographic algorithm known answer test and software integrity test:

- AES Known Answer Test
- HMAC Known Answer Test
- DRBG Known Answer Test
- SHA256/384 Known Answer Test
- RSA Signature Known Answer Test
- DHE Known Answer Test
- ECDH Known Answer Test
- ECDSA Known Answer Test
- Software Integrity Test: The hash value of software is stored in file header, the Integrity Test performs a hash function of the software and compares the result stored in file header.

7.5.4 FPT_TUD_EXT.1 Trusted Update

Only authenticated administrators have the ability to manually initiate an update to TOE firmware/software. During the updating procedure, digital signature as defined at FCS_COP.1/SigGen will be verified by the TOE at first.

The administrators can query the currently executing version of the TOE firmware/software as well as the most recently installed version by a command. The currently executing patches and most recently installed patches can also be checked out.

The validation of the firmware/software integrity is always performed before the process of replacing a non-volatile, system resident software component with another is initiated. All discrete software components (e.g. applications, drivers, kernel, and firmware) of the TSF are archived together into a whole package and the single package is digitally signed. RSA as specified in FCS COP.1/SigGen can be used for the firmware/software digital signature mechanism to authenticate it prior to installation and that installation fails if the verification fails.

When the digital signature is successfully verified, the new software will be installed successfully and become active when the TOE reboots.

When the digital signature verification fails, the new software will not be installed.

7.5.5 FPT_STM.1 Reliable Time Stamps

The router provides reliable timestamps. Time accuracy is guaranteed by the administrator for the first time and by the CPU in the long run.

The security functions that make use of time include:

1) The calculation of the real time for all audit data.

2) The calculation of the validation period of the certificate.

7.5.6 FPT_STM.2 Time Source

Only administrators have the ability to modify the time of TOE, and all modifications affecting time will be recorded.

7.5.7 FPT_FLS.1 Failure with preservation of secure state

The TSF shall preserve a secure state when the TSF self-testing fails.

7.6 TOE Access (FTA)

7.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

An administrator can configure maximum inactivity times for both local and remote administrative sessions. When a session is inactive (i.e., no session input) for the configured period of time, the TOE will terminate the session, flush the screen, and no further activity will be allowed, requiring the administrator to log in (be successfully identified and authenticated) again to establish a new session.

The allowable range is from 0 minutes 0 seconds to 35791 minutes 59 seconds.

7.6.2 FTA_SSL.3 TSF-initiated Termination

When the remote session is inactive (i.e., no session input) for the configured period of time, the TOE will terminate the session. By default, the timeout duration is 10 minutes.

7.6.3 FTA_SSL.4 User-initiated Termination

When the initiated administrator or local session is inactive (i.e., no session input) for the configured period of time, the TOE will terminate the session.

The administrator can terminate the interactive session by closing the login window of the device.

7.6.4 FTA_TAB.1 Default TOE Access Banners

To provide some prompts or alarms to users, an Administrator can use the header command to configure a title on the router. If a user logs in to the router, the title is displayed. An Administrator can specify the title, or specify the title information by using the contents of a file. The same title is displayed same for both local and remote users.

When a terminal (remote or local) connection is activated and someone attempts to log in, the terminal displays the contents of the title that is set by using the header login command. After the successful login, the terminal displays the contents of the title that is configured by using the header shell command.

The local Console port and the remote Secure Telnet interface are used by an administrator to communicate with the router.

7.7 Trusted path/channels (FTP)

7.7.1 FTP_ITC.1 Inter-TSF Trusted Channel

The TOE works as the client to protect the communications between the TOE and the Syslog server by using a TLS protocol. When the TOE acts as a client to establish a TLS connection with the Syslog server, the TOE uses the X.509 certificate defined by 6.1.3.7 to identify the audit server.

TLS protects the data from disclosure by encryption defined at 6.1.2.4 and ensures that the data has not been modified by MAC defined by 6.1.2.6.

7.7.2 FTP_TRP.1/Admin Trusted Path

All remote administrative communications take place over a secure encrypted SSH session. The remote users are able to initiate SSH communications with the TOE.

The TOE protects communications between the TOE and authorized remote administrators with SSH.

8 Crypto Disclaimer

The following cryptographic algorithms are used by ATN Series Routers software to enforce its security policy:

#	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Standard of Application	Comments
1	Key Generation	FFC schemes	-	3072-bit or greater	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.1	FCS_CKM.1
		ECC schemes	-	256 bits or greater	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4	
2	Key Establishment	Finite field-based key establishment schemes	Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography	3072 bit or greater	NIST Special Publication 800-56A Revision 2	FCS_CKM.2
		Elliptic curve-based key establishment schemes	Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography	256 bits	NIST Special Publication 800-56A Revision 2	FCS_CKM.2
3	Confidentiality	AES in GCM mode	-	128 bits or 256 bits	AES as specified in ISO 18033-3, GCM as specified in ISO 19772	FCS_COP.1/ DataEncryption
		AES in CTR mode	-	128 bits or 256 bits	AES as specified in ISO 18033-3, CTR as specified in ISO 10116	
4	Authentication	RSA signature	RSA: PKCS#1_V2.1, RSASSA-PKCS2v1_5	3072 bits to 4096 bits	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5	FCS_COP.1/ SigGen

			Digital signature scheme 2 or Digital Signature scheme 3		ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	
		ECDSA signature	“NIST curves” ISO/IEC 14888-3, Section 6.4	256 bits or greater	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D	
5	Secure Update	RSA signature	RSA: PKCS#1_V2.1, RSASSA-PKCS2v1_5	4096 bits	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5	FCS_COP.1/ SigGen
6	Integrity	SHA-256 and SHA-384	-	256 bits,384 bits	ISO/IEC 10118-3:2004	FCS_COP.1/Hash
7	Cryptographic Primitive	HMAC-SHA-256	-	256 bits	ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2	FCS_COP.1/ KeyedHash
8	Random Bit Generation	Hash_DRBG (any); DRG.2 acc. to SP800-90A	-	256 bits	SP800-90A ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”	FCS_RBG_EXT.1
9	Trusted Channel	SSH V2.0	RFC 4251 RFC 4252 RFC 4253 RFC 4254 RFC 4256 RFC 4344 RFC 5647 RFC 5656 RFC 6668	-	-	FTP_TRP.1/ Admin

		TLS1.2	RFC 5288 RFC 5246 RFC 6125	-	-	FTP_ITC.1
10	Cryptographic Primitive	Generation of prime numbers for RSA	None	-	-	Miller-Rabin-Test is used as primality test.

Referenced Documents

[FIPS 186-4] National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication FIPS PUB 186-4, July 2013

[PKCS#1] RSA Cryptography Specifications Version 2.1(RFC3447)

[PKCS#3] A cryptographic protocol that allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel.

[FIPS 198-1]The Keyed-Hash Message Authentication Code (HMAC)--2008 July

[RFC 4251]The Secure Shell (SSH) Protocol Architecture, January 2006

[RFC 4252]The Secure Shell (SSH) Authentication Protocol, January 2006

[RFC 4253]The Secure Shell (SSH) Transport Layer Protocol, January 2006

[RFC 4254]The Secure Shell (SSH) Connection Protocol, January 2006

[RFC 6668]SHA-2 Data Integrity Verification for the Secure Shell (SSH) Transport Layer Protocol

[RFC 3268]Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)

[RFC 4346]The Transport Layer Security (TLS) Protocol Version 1.1

[RFC 5246]The Transport Layer Security (TLS) Protocol Version 1.2

[RFC 8446]The Transport Layer Security (TLS) Protocol Version 1.3

[RFC 6125]Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)

[NIST SP 800-56A]National Institute of Standards and Technology, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, May 2013

[NIST SP 800-56B]National Institute of Standards and Technology, Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography August 2009

[ISO/IEC 18031:2011] Information technology -- Security techniques -- Random bit generation

[ISO 18033-3] Information technology — Security techniques — Encryption algorithms

[ISO/IEC 9796-2]Information technology -- Security techniques -- Digital signature schemes giving message recovery

[ISO/IEC 9797-2]Information technology -- Security techniques -- Message Authentication Codes (MACs)

[ISO/IEC 10118-3]Information technology -- Security techniques -- Hash-functions

[ISO/IEC 14888-3] Information technology -- Security techniques -- Digital signatures with appendix

9 Abbreviations Terminology and References

9.1 Abbreviations

Name	Explanation
AAA	Authentication Authorization Accounting
CA	Certificate Authority
CC	Common Criteria
CEM	Common Evaluation Methodology for Information Technology Security
CLI	Command Line Interface
EXEC	Execute Command
GUI	Graphical User Interface
IC	Information Center
IP	Internet Protocol
LMT	Local Maintenance Terminal
MAN	Metropolitan Area Network
NMS	Network Management Server
PP	Protection Profile
RMT	Remote Maintenance Terminal
SFR	Security Functional Requirement
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target

Name	Explanation
STP	Spanning-Tree Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions

9.2 Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Terminology	Explanation
Administrator:	An administrator is a user of the TOE who may have been assigned specific administrative privileges within the TOE. This ST may use the term administrator occasionally in an informal context, and not in order to refer to a specific role definition – from the TOE's point of view, an administrator is simply a user who is authorized to perform certain administrative actions on the TOE and the objects managed by the TOE. Since all user levels are assigned to commands and users and users can only execute a command if their associated level is equal or higher compared to the level assigned to a command, a user might have certain administrative privileges but lacking some other administrative privileges. So the decision whether a user is also an administrator or not might change with the context (e.g. might be able to change audit settings but cannot perform user management).
Operator:	See User.
User:	A user is a human or a product/application using the TOE which is able to authenticate successfully to the TOE. A user is therefore different to a subject which is just sending traffic through the device without any authentication.

9.3 References

Name	Description
[CC]	Common Criteria for Information Technology Security Evaluation. Part 1-5 November 2022 Version 4.0 Revision 1
CC40R5P1	Common Criteria (CC) Part 1: Introduction and general model November 2022 Version 4.0 Revision 1
[CC40R1P2]	Common Criteria (CC) Part 2: Security functional components November 2022 Version 4.0 Revision 1
[CC40R1P3]	Common Criteria (CC) Part 3: Security assurance components November 2022 Version 4.0 Revision 1
[CC40R1P5]	Common Criteria (CC) Part 5: Pre-defined packages of security requirements November 2022 Version 4.0 Revision 1
[CEM]	Common Methodology for Information Technology Security Evaluation Evaluation methodology November 2022 Version 4.0 Revision 1
[ISO18031]	Information technology — Security techniques — Random bit generation Second edition 2011-11-15

Name	Description
[RFC 3526]	This document defines new Modular Exponential (MODP) Groups for the Internet Key Exchange (IKE) protocol. It documents the well known and used 1536 bit group 5, and also defines new 2048, 3072, 4096, 6144, and 8192 bit Diffie-Hellman groups numbered starting at 14. Please refer to the following link: http://www.rfc-editor.org/info/rfc3526
[RFC 4251]	This document describes the architecture of the SSH protocol, as well as the notation and terminology used in SSH protocol documents. It also discusses the SSH algorithm naming system that allows local extensions. Please refer to the following link: http://www.rfc-editor.org/info/rfc4251
[RFC 5280]	This memo profiles the X.509 v3 certificate and X.509 v2 certificate revocation list (CRL) for use in the Internet. Please refer to the following link: http://www.rfc-editor.org/info/rfc5280
[RFC 5759]	This document specifies a base profile for X.509 v3 Certificates and X.509 v2 Certificate Revocation Lists (CRLs) for use with the United States National Security Agency's Suite B Cryptography. Please refer to the following link: http://www.rfc-editor.org/info/rfc5759
[SP800-56A]	Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography Revision 2 May 2013
[SP800-56B]	Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography Revision 1 September 2014
[SP800-90A]	Recommendation for Random Number Generation Using Deterministic Random Bit Generators Revision 1 June 2015