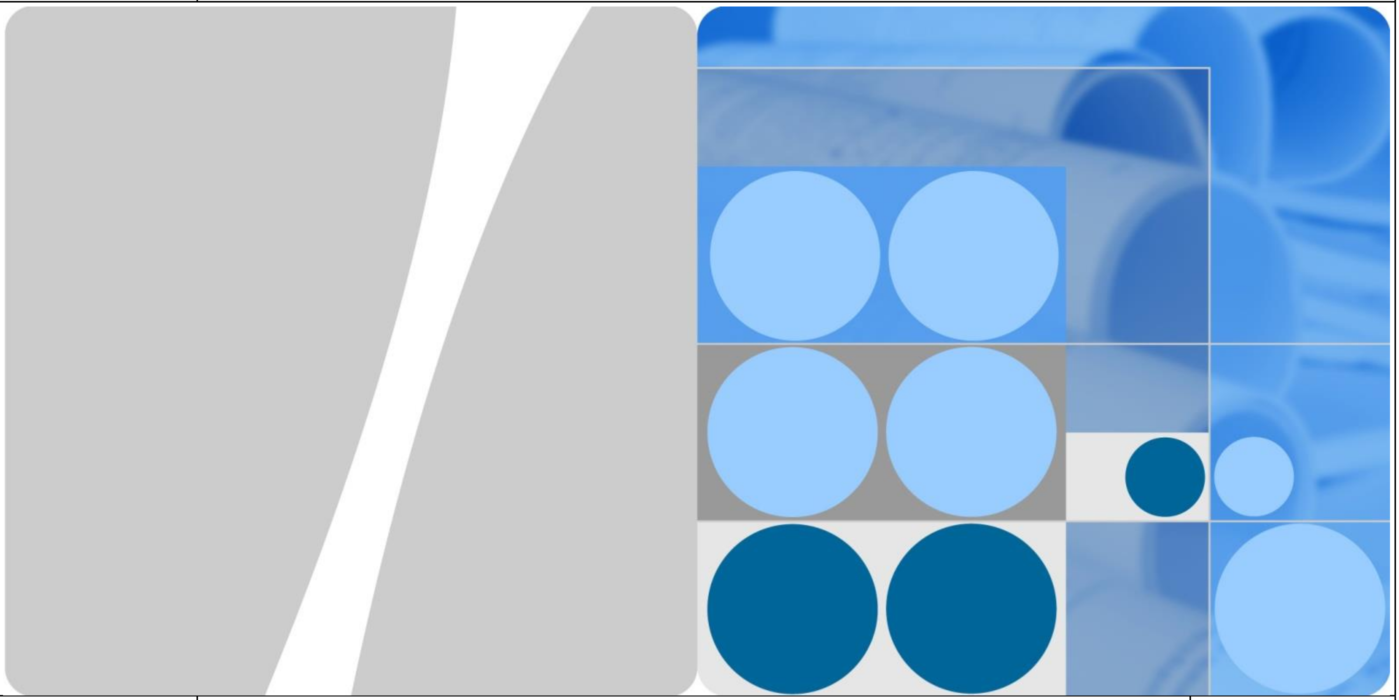


Huawei CX600&PTN 6900 Series Routers running VRP software

Security Target

Issue	**1.15**
Date	**2023-11-17**

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
2020-07-02	1.0	Initial Draft	hujunli
2020-09-09	1.1	Internal review completed.	hujunli
2021-01-26	1.2	Change TOE version	xiaxin
2021-03-05	1.3	Internal review completed.	xiaxin
2021-04-19	1.4	Internal review completed.	xiaxin
2021-08-23	1.5	External review completed.	xiaxin
2022-03-10	1.6	External review completed.	xiaxin
2022-03-23	1.7	External review completed.	xiaxin
2022-04-15	1.8	External review completed.	xiaxin
2022-05-05	1.9	External review completed.	xiaxin
2022-06-06	1.10	External review completed.	xiaxin
2022-06-11	1.11	External review completed.	xiaxin
2022-07-21	1.12	External review completed.	xiaxin
2023-04-24	1.13	Add Hash Value of the software	Huang Qiang
2023-07-04	1.14	Modified some descriptions about hash algorithm.	Wang Peiran
2023-11-17	1.15	Removed description about unused hash algorithm and minimum key size.	Wang Peiran

Contents

1 Introduction

1.1 ST reference and TOE Reference

Name	Description
ST Title	Huawei CX600&PTN 6900 Series Routers running VRP software Security Target
ST version	1.15
Vendor and ST author	Huawei Technologies Co., Ltd
TOE Name	Huawei CX600&PTN 6900 Series Routers running VRP software
TOE Hardware Models	CX600-M2K, CX600-M2K-B, CX600-X8A, CX600-X16A, PTN 6900-M2K, PTN 6900-M2K-B, PTN 6900-2-M8C, PTN 6900-2-M14
TOE software version	V800R021C00SPC100

1.2 TOE overview

The Huawei CX600&PTN 6900 series routers running VRP software TOE are used to satisfy the requirements for networks of various scales. They are deployed at the edge of IP backbone networks, IP metropolitan area networks (MANs), and other large-scale IP networks， also can be used to access, aggregate, and transmit carrier-class Ethernet services on Fixed-Mobile Convergence (FMC) Metropolitan Area Networks (MANs). 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

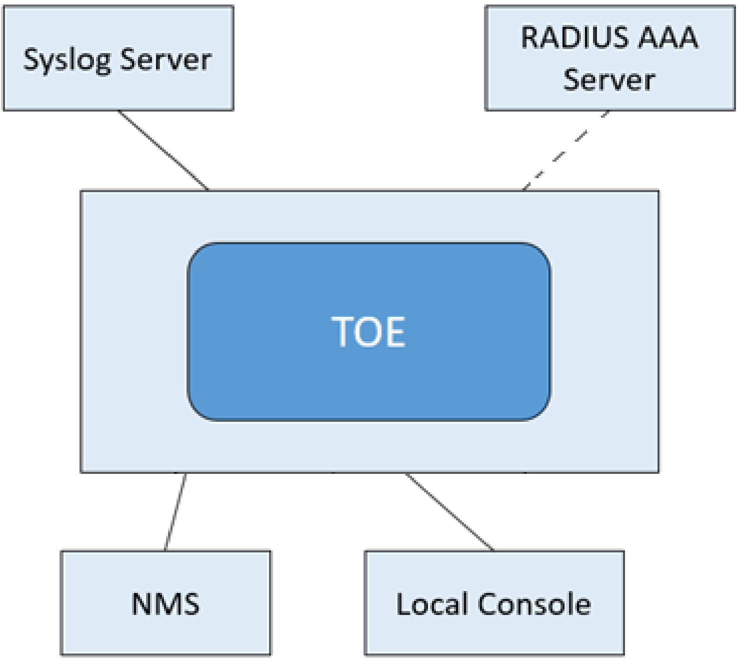
(7) Trusted path and channels for device authentication.

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



These IT entities should be physical protected in order to ensure that no one can attack them or stole information.

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**
RADIUS AAA Server	NO	This RADIUS AAA server provides user authentication. The TOE correctly leverages the services provided by this RADIUS AAA server to provide authentication to administrators.
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	YES	This includes any Console that is directly connected to the TOE via the

Component	Required	Usage/Purpose Description for TOE performance
Console		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 CX600&PTN 6900 Series Routers running VRP software comprised of both software and hardware. The software is comprised of Versatile Routing Platform (VRP) software, VRP is a network OS incorporating Huawei's proprietary intellectual properties and capable of supporting various network systems of Huawei. The hardware is comprised of the following: CX600-M2K, CX600-M2K-B, CX600-X8A, CX600-X16A, PTN 6900-M2K, PTN 6900-M2K-B, PTN 6900-2-M8C, PTN 6900-2-M14.

The Huawei CX600&PTN 6900 Series Routers running VRP software use the same VRP version. 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 Models

Hardware	Configuration	Processor	Interface
CX600-M2K	CX600-M2K Integrated Chassis,1 slot for fixed interfaces,2 slots for PIC(Physical Interface Card).	ARM	Based on TOE’s I/O modules
CX600-M2K-B	CX600-M2K-B Integrated Chassis,1 slots for fixed interfaces,2 slots for PIC.	ARM	Based on TOE’s I/O modules
CX600-X8A	CX600-X8A Integrated Chassis,2 slots for SRU(Switch and Route Processing Unit),2 slots for SFU (Switch Fabric Unit),8 slots for LPU(Line Processing Unit)	ARM	Based on TOE’s I/O modules
CX600-X16A	CX600-X16A Integrated Chassis,2 slots for MPU (Main Processing Unit）,4 slots for SFU,16 slots for LPU.	ARM	Based on TOE’s I/O modules

PTN 6900-M2K	PTN 6900-M2K Integrated Chassis, 1 slot for fixed interfaces,2 slots for PIC.	ARM	Based on TOE’s I/O modules
PTN 6900-M2K-B	PTN 6900-M2K-B Integrated Chassis, 1 slots for fixed interfaces,2 slots for PIC.	ARM	Based on TOE’s I/O modules
PTN 6900-2-M8C	PTN 6900-2-M8C Integrated Chassis, 2 slots for IPU(Integrated Network Processing Unit),8 slots(DC) or 6 slots(AC) for PIC.	ARM	Based on TOE’s I/O modules
PTN 6900-2-M14	PTN 6900-2-M14 Integrated Chassis,2 slots for IPU, 14 slots(DC) or 10 slots(AC) for PIC.	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 CX600&PTN 6900 Series Routers running VRP Software to be evaluated.

Table 1-3 Physical scope

Type	Delivery Item	Version
Hardware	CX600-M2K, CX600-M2K-B, CX600-X8A, CX600-X16A, PTN 6900-M2K, PTN 6900-M2K-B, PTN 6900-2-M8C, PTN 6900-2-M14 The Hardware will be delivered by air, ship, train or automobile.	NA
Software	CX600&PTN 6900 Router V800R021C00SPC100 Format: CX600-M2K : CX600-M2K_V800R021C00SPC100.cc Digital signature: CX600-M2K_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: 07020586a60abbc30e42091b327ce67164610eb164737be9f94687c62507b632 CX600-M2K-B：CX600-M2K-B_V800R021C00SPC100.cc Digital signature: CX600-M2K-B_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: f5a2d9324a9b3fbfac0a94fc7d6dbd36c1258255cedd00016def48c213e2b608 CX600-X8A: CX600-X8A-X16A_V800R021C00SPC100.cc Digital signature: CX600-X8A-X16A_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: d169db6a2db1e34703004a60d61c45bd3a728c53b26f23e9eca4e5423125dd18 CX600-X16A: CX600-X8A-X16A_V800R021C00SPC100.cc	V800R021C00SPC100

Type	Delivery Item	Version
	Digital signature: CX600-X8A-X16A_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: d169db6a2db1e34703004a60d61c45bd3a728c53b26f23e9eca4e5423125dd18 PTN 6900-M2K : PTN6900-M2K_V800R021C00SPC100.cc Digital signature: PTN6900-M2K_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: 9be57cf3fce417510017b8e5840715db18b4161ff26c33d45e7a58d1d28f5265 PTN 6900-M2K-B: PTN6900-M2K-B_V800R021C00SPC100.cc Digital signature: PTN6900-M2K-B_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: 9a11f20017efb222e9313e9d284a88fa5d25deb6005beac9b1d8c4a6a36ecd62 PTN 6900-2-M8C：PTN6900-2-M8C-M14_V800R021C00SPC100.cc Digital signature: PTN6900-2-M8C-M14_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also published on HUAWEI support website) HASH SHA256: 132163028f96f6935bd4422646cf6566dd72dcba7d4a55096dc9ad91d8bcc40c PTN 6900-2-M14 : PTN6900-2-M8C-M14_V800R021C00SPC100.cc Digital signature: PTN6900-2-M8C-M14_V800R021C00SPC100.cc.asc Info: User can obtain the software package directly from the local support engineer. Users can verify the software by digital signature (The digital signature is also	

Type	Delivery Item	Version
	published on HUAWEI support website) HASH SHA256: 132163028f96f6935bd4422646cf6566dd72dcba7d4a55096dc9ad91d8bcc40c	
Product guidance	Huawei CX600&PTN 6900 Series Routers V800R021C00SPC100 Operational user Guidance Info: The documentation is delivered by email in PDF format. HASH SHA256: 97d7443518cfc30dc1014ed7c34cf7a7a1a005986115972820c7d0b214db867a	1.7
	Huawei CX600&PTN 6900 Series Routers V800R021C00SPC100 Preparative Procedures Info: The documentation is delivered by email in PDF format. HASH SHA256: b2d7dbe305221415644bd6f2f87d70ac61059ceb847c9cab0c0c4772679bf943	1.7
	Huawei CX600&PTN 6900 Series Routers V800R021C00SPC100 Configuration and Reference Info: The documentation is delivered by email in PDF format. HASH SHA256: 9fa895c1b4ef8865bd33cd071847916b707346225a863229557c7588d5cfd0b0	1.4
	HUAWEI CX600 Product Documentation Product Version: V800R021C00 Library Version: 03 Date: 2021-12-31 HASH SHA256: B210EC09B884AD2DE0EE3142E02F62A63D4E9DB14DFAF54ECF6C1AA03B290211 HUAWEI CX600-M2 Product Documentation Product Version: V800R021C00 Library Version: 03 Date: 2021-12-31 HASH SHA256: 0A08951F5BFFD3DB8EDC1778D32377924DEB02893210299348EE6E8236EF5BDC HUAWEI PTN 6900-M2 Product Documentation Product Version: V800R021C00 Library Version: 03 Date: 2021-12-31 HASH SHA256:	refers to the “Library Version” shown in the left column

Type	Delivery Item	Version
	FD35813409F4AB82BB2C5914F95CAA6E1F13E5CD81C5B35DDB3BCAB357E107AF HUAWEI PTN 6900-2-M8C, PTN 6900-2-M14 Product Documentation Product Version: V800R021C00 Library Version: 03 Date: 2021-12-31 HASH SHA256: FEDBA960DCA605577918A22EE46B3B04F4699029CF2877B00472942D08F4E2C3 Info: The product documentations are delivered by email. The file format is *.hdx, user can download the *.hdx reader from Huawei support website. These product documentations apply for specific products of child versions belonging to the master product version V800R021C00. The evaluated TOE version V800R021C00SPC100 is considered as one of the child versions belonging to this master product version.	

There are only hardware differences between different devices. All the routers share the same platform so the SFRs are the same. Network management server, local console and syslog server are supported by all TOE evaluated configurations. The TOE only has one configuration.

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 identifier, version number, module name, log level, description of log, information type, system component ID and information about details.

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
ECDH	Used in session establishment of SSH
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
RSA	Used in the authentication of TLS
ECDSA	Used in the authentication of 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, account lock, user kick out, can be applies 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 for remote administrative session 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 attack.

The TOE supports password-based authentication for local administrative session.

(7) Trusted path and channels for device authentication

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

The TOE protects communications between a TOE and authorized remote administrator with SSH.

1.6 Standalone TOE

[CPP_ND], chapter 3 introduces distributed TOEs, i.e. TOEs that consist of more than one component. This does not refer to different software components running on one hardware component but same version software components running on each hardware components.

This ST refers to a standalone TOE which is not a distributed TOE in the sense of [CPP_ND], chapter 3. All additional requirements that are defined for distributed TOEs within [CPP_ND] are therefore ignored in this ST. There are dedicated paragraphs in several Application Notes of [CPP_ND] which are only applicable to distributed TOEs. These dedicated paragraphs have not been integrated into the Application Notes in this ST since the TOE is not a distributed TOE.

2 PP conformance claims

2.1 CC Conformance Claim

As defined by the references [CC1], [CC2] and [CC3], this ST:

- conforms to the requirements of Common Criteria v3.1, Revision 5
- is Part 2 extended, Part 3 conformant
- does not claim conformance to any other PP than the one specified in chap 2.2
- does not claim conformance to any Evaluation Assurance Level as defined in [CC3], chap. 8.

2.2 Protection Profile Conformance

This security target claims “Exact Conformance” to [CPP_ND]. Note that "Exact Conformance" is defined in [CPP_ND], chap. 2.

The methodology applied for the [CPP_ND] evaluation is defined in [CEM]. In addition to [CEM], the evaluation activities for [CPP_ND] are completed in [SD_ND].

2.3 Conformance Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the [CPP_ND].

2.3.2 TOE Security Problem Definition Consistency

The Threats, Assumptions, and Organization Security Policies included in the Security Target represent the Threats, Assumptions, and Organization Security Policies specified in [CPP_ND] for which conformance is claimed verbatim. All concepts covered in the collaborative Protection Profile Security Problem Definition are included in the Security Target.

2.3.3 Statement of Security Objectives Consistency

The security objectives included in the security target represent the security objectives specified in [CPP_ND] for which conformance is claimed verbatim. All concepts covered in Protection Profile`s Statement of security objectives are included in the Security Target.

2.3.4 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the [CPP_ND] for which conformance is claimed verbatim. All concepts covered the Protection Profile’s Statement of Security Requirements are included in the Security Target. Additionally, the Security Assurance Requirements included in the Security Target are identical to the Security Assurance Requirements included in section 6 of the [CPP_ND].

3 Security Problem Definition

3.1 Threats

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

3.1.1 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 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).

3.1.2 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_EXT.1
- Management of cryptographic functions is specified in FMT_SMF.1

3.1.3 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
- Requirements for use of public key certificates to support secure protocols are defined in FIA_X509_EXT.1/Rev, FIA_X509_EXT.2

3.1.4 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

3.1.5 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 and FIA_X509_EXT.2
- Requirements for management of updates are defined in FMT_SMF.1 and (for manual updates) in FMT_MOF.1/ManualUpdate

3.1.6 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_EXT.1
- Requirements for protecting audit records stored on the TOE are specified in FAU_STG.1
- Requirements for secure transmission of local audit records to an external IT entity via a secure channel are specified in FAU_STG_EXT.1
- Additional requirements for dealing with potential loss of locally stored audit records are specified in FAU_STG.3/LocSpace
- 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.

3.1.7 T.SECURITY_FUNCTIONALITY_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.4
- 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),

3.1.8 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.

3.1.9 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_EXT.1

3.2 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.

3.2.1 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 and/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 will not include any requirements on physical tamper protection or other physical attack mitigations. The ST will 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.

[OE.PHYSICAL]

3.2.2 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]

3.2.3 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. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of network devices (e.g., firewall).

[OE.NO_THRU_TRAFFIC_PROTECTION]

3.2.4 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 being 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.

For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).

[OE.TRUSTED_ADMIN]

3.2.5 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]

3.2.6 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]

3.2.7 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]

3.3 Organizational Security Policies

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

3.3.1 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

4.1 Security Objectives for the Operational Environment

The following subsections describe objectives for the Operational Environment.

4.1.1 OE.PHYSICAL

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

4.1.2 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.

4.1.3 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.

4.1.4 OE.TRUSTED_ADMIN

The 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.

4.1.5 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.

4.1.6 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.

4.1.7 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.

5 Extended Components Definition

The extended components used in this ST are defined in [CPP_ND]. The following table provide a chapter specific reference in which chapter of [CPP_ND] each of the extended components is defined.

Table 5-1 Definition of Extended Components - references to [CPP_ND]

Extended Component	Defined in [CPP_ND] chap.
Mandatory Requirements (<M>)	
FAU_STG_EXT.1	C.1.2.1
FCS_RBG_EXT.1	C.2.1.1
FIA_PMG_EXT.1	C.3.1.1
FIA_UIA_EXT.1	C.3.2.1
FIA_UAU_EXT.2	C.3.3.1
FPT_SKP_EXT.1	C.4.1.1
FPT_APW_EXT.1	C.4.2.1
FPT_TST_EXT.1	C.4.3.1
FPT_TUD_EXT.1	C.4.4.1
FPT_STM_EXT.1	C.4.5.1
FTA_SSL_EXT.1	C.5.1.1
Optional Requirements (<O>)	
None	None.
Selection-Based Requirements (<S>)	
FCS_SSHC_EXT.1	C.2.2.6
FCS_SSHS_EXT.1	C.2.2.7
FCS_TLSC_EXT.1	C2.2.8
FIA_X509_EXT.1	C.3.4.1
FIA_X509_EXT.2	C.3.4.2

6 Security Functional Requirements

Conventions

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

- Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD);
- Refinement made in the [CPP_ND] and ST: the refinement text is indicated with **bold text** and ~~strikethroughs~~;
- Selection wholly or partially completed in the [CPP_ND] and ST: the selection values (i.e. the selection values adopted in the [CPP_ND] or the remaining selection values available for the ST) are indicated with underlined text

  e.g. "[selection: *disclosure, modification, loss of use*]" in [CC2] or an ECD might become "disclosure" (completion) or "[selection: disclosure, modification]" (partial completion) in the [CPP_ND] and ST;
- Assignment wholly or partially completed in the [CPP_ND] and ST: indicated with *italicized text*;
- Assignment completed within a selection in the [CPP_ND] and ST: the completed assignment text is indicated with *italicized and underlined text*

  e.g. "[selection: *change_default, query, modify, delete, [assignment: other operations]*]" in [CC2] or an ECD might become "change_default, *select_tag*" (completion of both selection and assignment) or "[selection: change_default, *select_tag*, *select_value*]" (partial completion of selection, and completion of assignment) in the [CPP_ND] and ST;
- Iteration: indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").
- 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].

[CPP_ND] distinguishes mandatory requirements from optional requirements and selection-based requirements. This ST will mark mandatory requirements by <M>, optional requirements by <O> and selection-based requirements by <S>.

6.1 Functional Security Requirements

6.1.1 Security Audit (FAU)

6.1.1.1 FAU_GEN.1 Audit data generation<M>

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

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

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

c) *All administrative actions comprising:*

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- *Starting and stopping services.*

d) *Specifically defined auditable events listed in Table 6-1.*

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 by using the command: local-user change-password.

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *information specified in column three of Table 6-1.*

Table 6-1 Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
Mandatory Requirements (<M>)		
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FCS_COP.1/KeyedHash	None.	None.
FCS_RBG_EXT.1	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.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged.)	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).
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
Optional Requirements (<O>)		
FAU_STG.1	None.	None.
FAU_STG.3/LocSpace	Low storage space for audit events.	None.

Requirement	**Auditable Events**	**Additional Audit Record Contents**
Selection-Based Requirements (<S>)		
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	None.	None.
FMT_MOF.1/Functions	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.

6.1.1.2 FAU_GEN.2 User identity association<M>

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_EXT.1 Protected Audit Event Storage<M>

FAU_STG_EXT.1.1 The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. TOE shall consist of a single standalone component that stores audit data locally.

FAU_STG_EXT.1.3 The TSF shall overwrite previous audit records according to the following rule*:* *overwrite the oldest log information always* when the local storage space for audit data is full.

6.1.1.4 FAU_STG.3/LocSpace Action in case of possible audit data loss < O >

FAU_STG.3.1/LocSpace The TSF shall *generate a warning to inform the Administrator* if the audit trail *exceeds the local audit trail storage capacity*.

Additional Application Note c: The local storage that store audit data is CF card.

6.1.1.5 FAU_STG.1 Protected audit trail storage <O>

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

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

6.1.2 Cryptographic Support (FCS)

6.1.2.1 FCS_CKM.1 Cryptographic Key Generation (Refinement) <M>

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

- ECC schemes using "NIST curves" P-256, P-384 and P-521 that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4**;**
- 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**;**

~~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 Establishment (Refinement)<M>

FCS_CKM.2.1 The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method:

- Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";
- 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.4 Cryptographic Key Destruction<M>

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method:

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

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)<M>

FCS_COP.1.1/SigGen The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm:

- RSA Digital Signature Algorithm and cryptographic key sizes (modulus): *3072 bits or greater, 3072 bits is default.*
- 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) <M>

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*~~] and **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) <M>

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_EXT.1 Random Bit Generation<M>

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using Hash_DRBG (any).

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

6.1.2.9 FCS_SSHC_EXT.1 SSH Client Protocol<S>

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) 4251, 4252, 4253, 4254, 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 and no other 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 no more than one gigabyte of transmitted data. After either 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 or no other methods as described in RFC 4251 section 4.1.

6.1.2.10 FCS_SSHS_EXT.1 SSH Server Protocol <S>

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) 4251, 4252, 4253, 4254, 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 no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

6.1.2.11 FCS_TLSC_EXT.1 TLS Client Protocol <S>

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

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.

Additional Application Note d: The selected ciphersuites are accepted as legacy mechanisms by the SOG-IS.

6.1.3 Identification and Authentication (FIA)

6.1.3.1 FIA_AFL.1 Authentication Failure Management (Refinement)<M>

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*.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall prevent the offending remote Administrator from successfully authenticating until *unlock* is taken by a local Administrator*;* prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed.

6.1.3.2 FIA_PMG_EXT.1 Password Management<M>

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.

Additional Application Note e: Only when quotation marks (") are used around the password as both the first and last character, spaces can be used in the password. The space should be inside two quotation marks and it must not have the third quotation mark in the password, such as the available password like *"Abc123 "*, *"Abc 123"*, *" Adc123"*. The password like *"Adc "123"* is not allowed.

6.1.3.3 FIA_UIA_EXT.1 User Identification and Authentication <M>

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;
- *ICMP echo*.

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 <M>

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, and no other authentication mechanism to perform local administrative user authentication.

6.1.3.5 FIA_UAU.7 Protected Authentication Feedback <M>

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 <S>

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 <S>

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 accept the certificate.

6.1.4 Security Management (FMT)

6.1.4.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour <M>

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<S>

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 <S>

FMT_MOF.1.1/Services The TSF shall restrict the ability to ~~enable and disable~~ **start and stop** ~~the functions~~ **services** *to Security Administrators*.

6.1.4.4 FMT_MTD.1/CoreData Management of TSF Data <M>

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

6.1.4.5 FMT_MTD.1/CryptoKeys Management of TSF data <S>

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 <M>

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 behavior;
- Ability to modify 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;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- 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;

6.1.4.7 FMT_SMR.2 Restrictions on security roles <M>

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) <M>

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<M>

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_EXT.1 TSF Testing (Extended) <M>

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

6.1.5.4 FPT_TUD_EXT.1 Trusted Update <M>

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_EXT.1 Reliable Time Stamps <M>

FPT_STM_EXT.1.1 The TSF shall be able to provide reliable time stamps for its own use.

FPT_STM_EXT.1.2 The TSF shall allow the Security Administrator to set the time.

6.1.6 TOE Access (FTA)

6.1.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking <M>

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 (Refinement) <M>

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

6.1.6.3 FTA_SSL.4 User-initiated Termination (Refinement)<M>

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

6.1.6.4 FTA_TAB.1 Default TOE Access Banners (Refinement) <M>

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

6.1.7 Trusted path/channels (FTP)

6.1.7.1 FTP_ITC.1 Inter-TSF trusted channel (Refinement)<M>

FTP_ITC.1.1 The TSF shall **be capable of using TLS to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, no other capabilities** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *audit service*.

6.1.7.2 FTP_TRP.1/Admin Trusted Path (Refinement) <M>

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

FTP_TRP.1.2/Admin The TSF shall permit 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 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:

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 for the operational environment (ASE_OBJ.1)
	Stated security requirements (ASE_REQ.1)
	Security Problem Definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life cycle support (ALC)	Labeling of the TOE (ALC_CMC.1)
	TOE CM coverage (ALC_CMS.1)

Assurance Class	Assurance Components
Tests (ATE)	Independent testing – sample (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

This security target claims conformance with [CPP_ND]. In addition to [CEM], the evaluation activities for [CPP_ND] are completed in [SD_ND].

6.3 SFR 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 6-3 Dependency rationale for SFRs

Requirement	Dependencies	Satisfied by
Mandatory Requirements (<M>)		
FAU_GEN.1	FPT_STM.1	FPT_STM_EXT.1 included (which is hierarchic to 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_EXT.1	FAU_GEN.1; FTP_ITC.1	FAU_GEN.1; FTP_ITC.1
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1; FCS_CKM.4	FCS_CKM.2; FCS_CKM.4
FCS_CKM.2	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1; FCS_CKM.4	FCS_CKM.1 (also FTP_ITC.1 as a secure channel that could be used for import); FCS_CKM.4
FCS_CKM.4	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1	FCS_CKM.1 (also FTP_ITC.1 as a secure channel that could be used for import)
FCS_COP.1/DataEncryption	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1; FCS_CKM.4	FCS_CKM.1 (also FTP_ITC.1 as a secure channel that could be used for import); FCS_CKM.4
FCS_COP.1/SigGen	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1;	FCS_CKM.1 (also FTP_ITC.1 as a secure

Requirement	Dependencies	Satisfied by
	FCS_CKM.4	channel that could be used for import); FCS_CKM.4
FCS_COP.1/Hash	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1; FCS_CKM.4	Unsupported Dependencies: This SFR specifies keyless hashing operations, so initialisation and destruction of keys are not relevant
FCS_COP.1/KeyedHash	FTP_ITC.1 or FTP_ITC.2 or FCS_CKM.1; FCS_CKM.4	FCS_CKM.1 (also FTP_ITC.1 as a secure channel that could be used for import); FCS_CKM.4
FCS_RBG_EXT.1	None	N/A
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_SMF.1	FMT_SMR.2; 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_EXT.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~~ Note: The TOE uses a digital signature mechanism to authenticate firmware/software updates. Therefore, the dependency is only satisfied by FCS_COP.1/SignGen.
FPT_STM_EXT.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	None	N/A
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
Optional Requirements (<O>)		
FAU_STG.1	FAU_STG.3	FAU_STG.3/LocSpace

Requirement	Dependencies	Satisfied by
FAU_STG.3/LocSpace	FAU_STG.1	FAU_STG.1
Selection-Based Requirements (<S>)		

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_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_EXT.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_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_EXT.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_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_EXT.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
FMT_MTD.1/CryptoKeys	FMT_SMR.1; FMT_SMF.1	FMT_SMR.2; FMT_SMF.1

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, “Table 6-1 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., 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: date and time when a log is output, hostname (the default name is HUAWEI), Huawei identifier (indicated as two precent sign %%) which follows the device name, the version of the log format, the name of the module by which logs are output, the severity of a log, the further description of a log, the type of the log information, the system component ID to which a log belongs and the detail information about the system component output. 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 command 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. The generation, import, and destruction of key pairs of different types are distinguished based on the value of “Command” in command operation logs. Only one type of key pair exists on the device. The new key pair overwrites the original one.

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 Protected audit trail storage

Only the authorized administrators can monitor the logfile record, and operate the log files. The unauthorized users have no access to do those actions. And the actions of the authorized administrators will be logged.

7.1.4 FAU_STG_EXT.1 Protected audit event storage

The TOE supports to export 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 CF card 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 varies with log file types. 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. When the size of an information file exceeds the configured maximum size, the information file is compressed into a smaller file in standard log_SlotID_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.

When the local audit data store in CF card exceeds the maximum allowed size of log file storage, it will overwrite the oldest log information always.

The logs are saved to flash memory (internal CF card) 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. The size of the log buffer can be configured by users with sufficient privileges.

7.1.5 FAU_STG.3/LocSpace Action in case of possible audit data loss

If the log files have already occupied more than 85% 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.

7.2 Cryptographic Support (FCS)

7.2.1 FCS_CKM.1 Cryptographic Key Generation

The TOE supports

1) ECC schemes using “NIST curves” [P-256, P-384 and P-521] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4.
2) 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

7.2.2 FCS_CKM.2 Cryptographic Key Establishment

a) The TOE supports 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”. 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 are used when the TOE establishes SSH connection.

b) The TOE supports 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”. The key size supports at least 3072 bits. The Finite field-based key establishment schemes are used when the TOE establishes TLS connection.

7.2.3 FCS_CKM.4 Cryptographic Key Destruction

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: zeros.

Name	Description of Key	Storage	Key destruction method
FFC key	The key is used for key establishment.	SDRAM (plaintext)	Automatically after completion of use of the key. Overwritten with: zeros.
TLS private key	The key is used for signature and authentication.	CF card (AES256 cipher)	After using "certificate load" command, the private key will be deleted from CF card automatically. Overwritten with: a new value of the key.
ECDH key	The key pair is used for key establishment. Random number is used as ECDH key.	SDRAM (plaintext)	Automatically after completion of use of the key. Overwritten with: zeros.
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: zeros.
ECC host key pair	Using command generates a ECC host key pair.	CF card (AES256 cipher)	Zeroized using "undo ecc key-pair label" command. Overwritten with: zeros.
Key encrypt key	Key encrypt key is used to generate AES key. Note: Key encrypt key is encrypted by root key. The root key is generated by root key material. The root key material is saved many places, for example: code and CF card.	CF card (AES256 cipher)	After using "clear master-key" command, the key encrypt key will be overwritten with a new value. Overwritten with: a new value of the key.
AES key	The AES key is generated by Key encrypt key. AES key is used to encrypt ECC host key pair and TLS private key.	SDRAM (plaintext)	The AES key is stored in the SDRAM temporarily and destroyed after used. Overwritten with: zeros.

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

The TOE provides symmetric encryption and decryption capabilities using AES algorithm with key size 128 bits, 256 bits in GCM mode as specified in ISO 19772 and CTR mode as specified in ISO 10116.

- AES128 GCM, AES256 GCM are supported by TLS.
- AES128 GCM, AES256 GCM are supported by SSH.
- AES128 CTR, AES256 CTR are supported by SSH.

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.

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 meet the ISO/IEC 10118-3:2004.

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

Table 7-2 Usage of Hash Algorithm

Cryptographic Functions	Hash Function
HMAC-SHA-256	SHA-256

Cryptographic Functions	Hash Function
TLS Digital signature verification	SHA-256 SHA-384
SSH Digital signature verification	SHA-256 SHA-384
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_EXT.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 [SP800-90A], chap. 10.1.1.

The entropy source is based on hardware (internal noise source). Random numbers from the internal noise source are only used for seeding the DRBG.

The TOE set new seed using at least 256 bits entropy before generating random bits as cryptographic key.

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_SSHC_EXT.1 SSH Client Protocol

7.2.9.1 FCS_SSHC_EXT.1.1

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

7.2.9.2 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 algorithms conform to FCS_SSHC_EXT.1.5.

7.2.9.3 FCS_SSHC_EXT.1.3

The TOE drops packets greater than 256 KB in an SSH transport connection. Packets of size greater than 35000 bytes and smaller than 256 KB are not dropped because of that the TOE may support uncompressed big certificates.

7.2.9.4 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 own random number to create an encryption key.

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

7.2.9.5 FCS_SSHC_EXT.1.5

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

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

When Client authenticates Server, first step 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.

7.2.9.6 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”.

7.2.9.7 FCS_SSHC_EXT.1.7

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

7.2.9.8 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 which ever goes first. The session duration and maximum packet data volume that trigger the key re-negotiation should be configured to one hour and one gigabyte through the associated command.

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 which ever goes first.

7.2.9.9 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.

7.2.10 FCS_SSHS_EXT.1 SSH Server Protocol

7.2.10.1 FCS_SSHS_EXT.1.1

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

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

7.2.10.3 FCS_SSHS_EXT.1.3

The TOE drops packets greater than 256 KB in an SSH transport connection. Packets of size greater than 35000 bytes and smaller than 256 KB are not dropped because of that the TOE may support uncompressed big certificates.

7.2.10.4 FCS_SSHS_EXT.1.4

SSH server function 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 own random number to create an encryption key.

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

7.2.10.5 FCS_SSHS_EXT.1.5

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

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

When Client authenticates Server, first step 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_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”.

7.2.10.7 FCS_SSHS_EXT.1.7

SSH server supports the following key exchange algorithm: ecdh-sha2-nistp256.

7.2.10.8 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 which ever goes first. The session duration and maximum packet data volume that trigger the key re-negotiation should be configured to one hour and one gigabyte through the associated command.

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 which ever goes first.

7.2.11 FCS_TLSC_EXT.1 Extended: TLS Client Protocol

7.2.11.1 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

7.2.11.2 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.

7.2.11.3 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.

7.2.11.4 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

The TOE can be configured within 3 to 5 unsuccessful authentication attempts for remote authentication by Administrators. When the defined number of unsuccessful authentication attempts has been met, the TOE will prevent the offending remote Administrator from successfully authenticating until 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 administrators, the account locking function should be enabled for remote administrators who fail 3 to 5 authentication attempts.

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 automatically. We also have the command to manually unlock the user account by local administrator.

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 question marks. Only when quotation marks (") are used around the password as both the first and last character, spaces can be used in the password. It must not have the third quotation mark in the password). Minimum password length is settable by the Authorized Administrator, and support passwords between 8 and 128 characters.

7.3.3 FIA_UIA_EXT.1 User Identification and Authentication

For local and remote TOE administration, 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 need authentication before successfully 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 are configured to be 'ALL'.

7.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

The TOE can be configured to require local authentication based on username and password.

Through the local console, users have to input own correct username and password to pass the authentication during the login process.

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. 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 to verify the certificate and the certificate path by the rules specified in RFC 5280, using algorithm RSA.

The TOE supports to verify 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. All the checkpoints take place when a TLS trusted channel is established between the TOE and the syslog server.

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 implement 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 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 authentication of 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 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 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 Management of security functions behaviour

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

The users are assigned to different user levels with different privileges. A user cannot execute the commands that the privilege levels are higher than the privilege level of the user. The commands used for manual update are of Level 3: management level so that only the Security Administrators can execute these commands. Other users with lower level (Level 0,1,2) have no privilege to execute these commands.

7.4.2 FMT_MOF.1/Functions Management of security functions behaviour

Only Security Administrators have the privilege to choose the trusted channel for external audit server and decide whether transmit the audit data to an external IT entity or not.

7.4.3 FMT_MOF.1/Services Management of security functions behaviour

Only Security Administrators have ability to start and stop the services, the other users are disallowed to do it.

7.4.4 FMT_MTD.1/CoreData Management of TSF Data

Only Security 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, configuration data. Each of the predefined and administratively configured user has different right 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, the user is authorized 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 Security Administrators have the right to delete, import the cryptographic keys, the other users are disallowed to do this.

The private key file of the certificate is encrypted and saved. User and administrator cannot access private key file.

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 through local console or SSH to perform these functions.

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 behavior;

- Ability to modify 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;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- 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;

7.4.7 FMT_SMR.2 Restrictions on security roles

All the users are administrators since the TOE is a network device that is not designed to be managed by unprivileged users. The TOE uses groups to organize users. Different kinds of users are in different group and every group has a specific level that identity 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 with the user level from 0 to 3: visit-ug, monitor-ug, system-ug and manage-ug, corresponding to the user roles Visit Administrator, Monitoring Administrator, Configuration Administrator and Security Administrator respectively.

User with the user level 3 is the Security Administrator. Security Administrator is able to administer the TOE through the local console or through a remote mechanism.

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, copy or extract by administrators; hence no interface access.

7.5.2 FPT_APW_EXT.1 Protection of Administrator Passwords

The administrator passwords are stored to 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_EXT.1 TSF testing

The TSF run 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 power on start-up, software integrity is checked at first. If integrity check is failed, the start-up procedure will stop. After VRP gain control, it tests the correct operation of cryptographic functions with known-answer test. If this testing fails, the start-up procedure will also stop.

Self-test includes 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
- ECDH Known Answer Test
- DHE Known Answer Test
- ECDSA Known Answer Test
- Software Integrity Test: The hash value of software is stored in file header, the Integrity Test perform a hash function of the software and compare the result stored in file header.

7.5.4 FPT_TUD_EXT.1 Trusted Update

Only Security 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 started. 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 firmware/software digital signature mechanism to authenticate it prior to installation and that installation fails if the verification fails.

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

When the digital signature verification fails, the new software will not be installed. During the startup, the hash of the root public key burnt in the EFUSE is used to verify the boot integrity, and the hardcoded public key in the boot is used to verify the integrity of the lower-level OS. After the verification is passed, the system software package is started. During the upgrade, the hardcoded public key of the current system software package is used to verify the validity of the digital signature of the next system software package.

7.5.5 FPT_STM_EXT.1 Reliable Time Stamps

Only Security Administrators have the ability to modify the time of TOE, and all modification about time will be recorded. The 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) With this information the real time for all audit data can be calculated.

2) The validation period of the certificate can be calculated.

7.6 TOE Access (FTA)

7.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

Security Administrators can configure maximum inactivity duration for local 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 is allowed. It requires the user to log in (be successfully identified and authenticated) again to establish a new session.

The allowable range of time period is from 0 minute 0 second to 35791 minutes 59 seconds.

7.6.2 FTA_SSL.3 TSF-initiated Termination

Security Administrators can configure maximum inactivity duration for remote administrative sessions. When the remote session is inactive (i.e., not session input) for the configured period of time the TOE will terminate the session. It requires the user to log in (be successfully identified and authenticated) again to establish a new session.

The allowable range of time interval is from 0 minute 0 second to 35791 minutes 59 seconds.

7.6.3 FTA_SSL.4 User-initiated Termination

The user can use the associated command to exit or log off own local or remote session.

7.6.4 FTA_TAB.1 Default TOE Access Banners

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

When a terminal (remote or local) connection is activated and attempt 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 (STelnet) interface are used for an administrator to communicate with the router. STelnet is a secure Telnet service based on SSH2.0. The client and server set up a secure connection through negotiation.

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 audit server by TLS protocol. When the TOE as a client to establish a TLS connection with the audit 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 using the encryption algorithm AES128 GCM, AES256 GCM and ensure that the data has not been modified by the Hash algorithm SHA256, SHA384.

The supported TLS cipher suites are defined in 6.1.2.11.

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 a TOE and authorized remote administrator with SSH.

The TOE can act as the client or the server in an SSH session. When the TOE acts as SSH client, it supports ECC public-key authentication to identify the SSH server.

SSH protects the data from disclosure by the encryption algorithm aes128-ctr, aes256-ctr, AEAD_AES_128_GCM, AEAD_AES_256_GCM, aes128-gcm@openssh.com, aes256-gcm@openssh.com and ensure that the data has not been modified by the MAC algorithm hmac-sha2-256.

The implementation of SSH protocol is defined in 6.1.2.9 and 6.1.2.10.

8 Crypto Disclaimer

The following cryptographic algorithms are used by Huawei CX600&PTN 6900 series routers running VRP 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	FCS_CKM.1

2	Key Establishment	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
		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
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	FCS_COP.1/ DataEncryption
4	Authentication	RSA signature	RSA: PKCS#1_V2.1, RSASSA-PKCS2v1_5	3072 bits or greater	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5	FCS_COP.1/ SigGen
			Digital signature scheme 2 or Digital Signature scheme 3	3072 bits or greater	ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	FCS_COP.1/ SigGen
		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	FCS_COP.1/ SigGen
5	Integrity	SHA-256, SHA-384	-	256 bits,384 bits	ISO/IEC 10118-3:2004	FCS_COP.1/Hash
6	Cryptographic Primitive	HMAC-SHA-256	-	256 bits	ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2	FCS_COP.1/ KeyedHash
7	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
8	Trusted Channel	SSH V2.0	RFC 4251 RFC 4252 RFC 4253 RFC 4254 RFC 5647	-	-	FTP_TRP.1/ Admin

			RFC 5656 RFC 6668			
		TLS1.2	RFC 5246 RFC 5288 RFC 6125	-	-	FTP_ITC.1
9	Cryptographic Primitive	Generation of prime numbers for RSA	None			Miller-Rabin-Test is used as primality test.

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
EAL	Evaluation Assurance Level
EXEC	Execute Command

Name	Explanation
GUI	Graphical User Interface
IC	Information Center
IP	Internet Protocol
NMS	Network Management Server
cPP	collaborative Protection Profile for Network Devices
PP	Protection Profile
RMT	Remote Maintenance Terminal
SFR	Security Functional Requirement
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
STP	Spanning-Tree Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
VRP	Versatile Routing Platform
AC	Alternating Current
DC	Direct Current

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 1-3 April 2017 Version 3.1 Revision 5
[CC1]	Common Criteria (CC) Part 1: Introduction and general model April 2017 Version 3.1 Revision 5
[CC2]	Part 2: Security functional components April 2017 Version 3.1 Revision 5

[CC3]	Part 3: Security assurance components April 2017 Version 3.1 Revision 5
[CEM]	Common Methodology for Information Technology Security Evaluation Evaluation methodology April 2017 Version 3.1 Revision 5
[CPP_ND]	collaborative Protection Profile for Network Devices, Version 2.1, 24-Sep-2018
[SD_ND]	Evaluation Activities for Network Device cPP September-2018 Version 2.1
[SOG-IS]	SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms, v.1.2, January 2020
[FIPS 186-4]	Digital Signature Standard (DSS), July 2013
[SP800-56A]	Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, Rev. 3, April 2018
[ISO 18033-3]	Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers Edition : 2 Publication date : 2010-12
[ISO 10116]	Information technology — Security techniques — Modes of operation for an n-bit block cipher Edition : 4 Publication date : 2017-07
[ISO 19772]	Information technology — Security techniques — Authenticated encryption Edition : 1 Publication date : 2009-02
[ISO/IEC 9796-2]	Information technology — Security techniques — Digital signature schemes giving message recovery — Part 2: Integer factorization based mechanisms Edition : 3 Publication date : 2010-12
[ISO/IEC 14888-3]	IT Security techniques — Digital signatures with appendix — Part 3: Discrete logarithm based mechanisms Edition : 4 Publication date : 2018-11

[ISO/IEC 10118-3]	Information technology — Security techniques — Hash-functions — Part 3: Dedicated hash-functions Edition : 3 Publication date : 2004-03
[ISO/IEC 9797-2]	Information technology — Security techniques — Message Authentication Codes (MACs) — Part 2: Mechanisms using a dedicated hash-function Edition : 2 Publication date : 2011-05
[ISO 18031]	Information technology — Security techniques — Random bit generation Edition : 2 Publication date : 2011-11
[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 5647]	AES Galois Counter Mode for the Secure Shell Transport Layer Protocol, August 2009
[RFC 5656]	Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer, December 2009
[RFC 6668]	SHA-2 Data Integrity Verification for the Secure Shell (SSH) Transport Layer Protocol, July 2012
[RFC 5246]	The Transport Layer Security (TLS) Protocol Version 1.2, August 2008
[RFC 5288]	AES Galois Counter Mode (GCM) Cipher Suites for TLS, August 2008
[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), March 2011
[RFC 5280]	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008
[FIPS 180-3]	Secure Hash Standard (SHS), October 2008
[RFC 2104]	HMAC: Keyed-Hashing for Message Authentication, February 1997
[SP800-90A]	Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Rev. 1, June 2015