MMA10G-EXE Series II Security Target

Document Version: 1.2

Issued by: Acumen Security, LLC

intertek
acumen
security

2400 Research Blvd
Suite 395
Rockville, MD 20850

Version	Date	Changes
Version 1.0	February 17, 2024	Initial draft
Version 1.1	July 17, 2024	Multiple updates including the TOE name, CAVP Certificates, and Admin Guide References.
Version 1.2	August 12, 2024	Minor changes to correct typographical errors. Updates to remove 3 models from table 2.

Contents

1 Introduction

The Security Target (ST) serves as the basis for the Common Criteria (CC) evaluation and identifies the Target of Evaluation (TOE), the scope of the evaluation, and the assumptions made throughout. This document will also describe the intended operational environment of the TOE, and the functional and assurance requirements that the TOE meets.

1.1 Security Target and TOE Reference

This section provides the information needed to identify and control the TOE and the ST.

Table 1 – TOE/ST Identification

Category	Identifier
ST Title	MMA10G-EXE Series II Security Target
ST Version	1.2
ST Date	August 12, 2024
ST Author	Acumen Security
TOE Identifier	MMA10G-EXE Series II
TOE Version	1.5
TOE Developer	Evertz Microsystems Ltd. 5292 John Lucas Drive Burlington, Ontario CANADA
Key Words	Network Device

1.2 TOE Overview

The MMA10G-EXE Series II switches are Internet Protocol (IP) switches optimized for video-over-IP traffic (compressed or uncompressed). The TOE is classified as a network device (a generic infrastructure device that can be connected to a network). Models of the EXE included in the evaluation provide identical functionality. The only differences between them are the supported speed, the physical size, and the number of physical interfaces supported, and the processor. These differences are detailed at the end of this section.

The EXE builds on the capabilities of the existing Evertz line of video routing switches. Video routers receive video signals in various formats, such as Serial Digital Interface (SDI), Serial Data Transport Interface (SDTI), or Asynchronous Serial Interface (ASI), and switch dedicated physical input ports to dedicated physical output ports based on external commands. The EXE provides the same capability within the context of packet-based networks using shared network infrastructure.

The TOE provides a packet-based switching fabric from a video perspective, rather than relying on traditional packet-based network architecture.

A typical EXE installation will also include a standard video routing switch software platform (such as Evertz Magnum) to route data between program streams in a manner sufficient to meet broadcast video standards for signal availability and integrity. Equipment to prepare video for IP transport, or to convert it into other video formats, and non-network-based video switching/processing, is outside the scope of this TOE. Such equipment includes, but is not limited to, cameras, KVMs, codecs, video servers and video displays. Equipment to perform functions such as embedding audio and/or other information within the video stream is also outside the scope of this TOE.

The TOE provides secure remote management using an HTTPS/TLS web interface. Administrators only may access EXE via a dedicated management workstation operating over an Out-of-Band Management (OOBM) network. Sites may close this OOBM network or may operate EXE within an existing OOBM as long as the topology is compliant with the security parameters listed below. Users and administrators may also access EXE software via direct connection using a terminal session.

The TOE generates audit logs and transmits the audit logs to a remote syslog server over an authenticated TLS channel. The TOE verifies the authenticity of software updates by verifying the digital signature prior to installing any update.

The summary of the evaluated functionality provided by the TOE includes the following,

- Secure connectivity with remote audit servers and secure retention of audit logs locally
- Identification and authentication of the administrator of the TOE
- Secure remote administration of the TOE via TLS and secure Local administration of the TOE
- Secure access to the management functionality of the TOE
- Secure software updates
- Secure communication with the non-TOE ‘video switch control systems’ via TLS.

The TOE hardware devices are the Evertz:

Table 2 – TOE hardware models

Model	AV/ Broadcast	Supported Ports	Form Factor	Chassis Supported	Frame Controller	Processor
NATX-8-100G-CC	broadcast	4 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
NATX-16-100G-CC	broadcast	8 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
NATX-32-100G-1-CC	broadcast	16 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
NATX-64-100G-2-CC	broadcast	32 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
MMA10G-NATX-8-CC	AV	4 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
MMA10G-NATX-16-CC	AV	8 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
MMA10G-NATX-32-CC	AV	16 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
MMA10G-NATX-64-CC	AV	32 x DD QSFP (QSFP200G)	1	DragonFire frame	N/A	Intel(R) Core (TM) i3-6102E
MMA10G-IPX128	AV	32 x QSFP+	3 or 6	EV Frame	ev3-FC or ev6-FC	Intel(R) Core (TM) i3-6102E

Model	AV/ Broadcast	Supported Ports	Form Factor	Chassis Supported	Frame Controller	Processor
3080IPX-48-25G-CC	AV/ broadcast	12 x QSFP+	3 or 6	EV Frame	ev3-FC or ev6-FC	Intel(R) Core (TM) i3-6102E

The EXE firmware version 1.5 will be referred to as EXE throughout this document.

The EXE appliances are Ethernet switches optimized for video content.

1.3 TOE Description

This section provides an overview of the TOE architecture, including physical boundaries, security functions, and relevant TOE documentation and references. The item outlined in red is considered the TOE boundary for testing purposes.

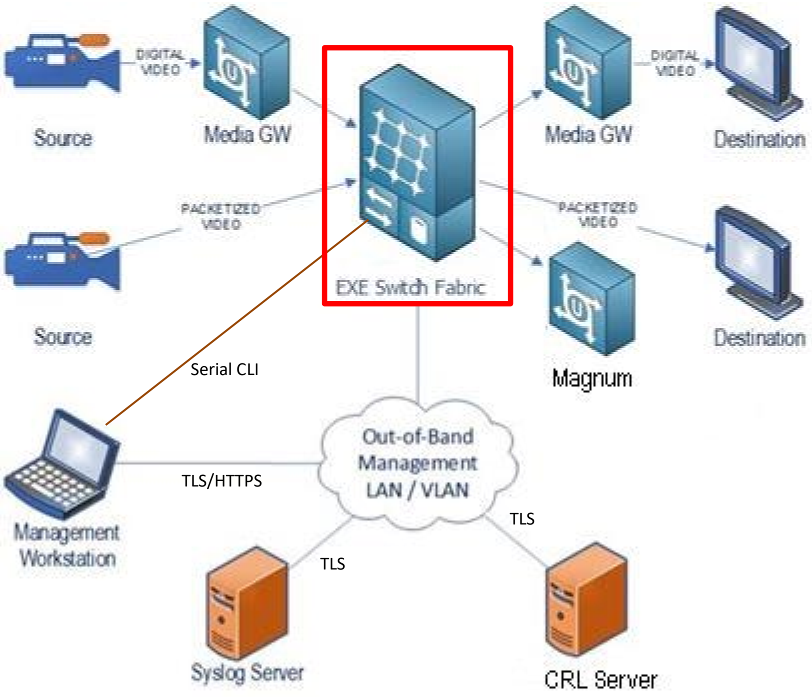

Figure 1 – Representative TOE Deployment

1.3.1 Physical Boundaries

The physical boundary of the TOE is the TOE hardware (as outlined in red in Figure 1 above). The TOE hardware is identified in section 1.2, table 2. The media and video components of the IT environment are NOT part of the TOE physical boundary. The TOE is shipped to the customer via commercial courier. The IT Testing Environment Components used to test the TOE are shown in Table 3 below:

Table 3 – TOE Physical Boundary Components

Component	Required	Purpose/Description
Syslog server	Yes	• Conformant with RFC 5424 (Syslog Protocol) • Supporting Syslog over TLS (RFC 5425)

Component	Required	Purpose/Description
		• Acting as a TLSv1.2 server • Supporting Client Certificate authentication • Supporting at least one of the following cipher suites: o TLS_RSA_WITH_AES_128_CBC_SHA o TLS_RSA_WITH_AES_256_CBC_SHA o TLS_RSA_WITH_AES_128_CBC_SHA256 o TLS_RSA_WITH_AES_256_CBC_SHA256 o TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 o TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
Local Management Workstation	Yes	• Computer with terminal emulation software to access the console interface (CLI)
Remote Management Workstation with web browser	Yes	• Internet Explorer, Google Chrome, or Firefox • Supporting TLSv1.2 • Supporting Client Certificate authentication • Supporting Server Certificate authentication • Supporting at least one of the following ciphersuites: o TLS_RSA_WITH_AES_128_CBC_SHA o TLS_RSA_WITH_AES_256_CBC_SHA o TLS_RSA_WITH_AES_128_CBC_SHA256 o TLS_RSA_WITH_AES_256_CBC_SHA256 o TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 o TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
CRL Server	Yes	• Conformant with RFC 5280
Magnum	Yes	• Provides remote management of the TOE's routing and switching of video signals. • Supporting TLSv1.2 with at least one of the following ciphersuites: o TLS_RSA_WITH_AES_128_CBC_SHA o TLS_RSA_WITH_AES_256_CBC_SHA o TLS_RSA_WITH_AES_128_CBC_SHA256 o TLS_RSA_WITH_AES_256_CBC_SHA256 o TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 o TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
Media Gateway	No	• Optional component for converting media streams. Not required for TOE operations
Video Source devices	No	• Optional component for creating video streams that are sent to the TOE. Not required for TOE operations. • Supporting packetized or digital video streams.
Video Destination devices	No	• Optional component for receiving video streams that are sent from the TOE. Not required for TOE operations. • Supporting packetized or digital video streams

1.3.2 Security Functions Provided by the TOE

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v2.2e or NDcPP.

1.3.2.1 Security Audit

The TOE's Audit security function supports audit record generation and review. The TOE provides date and time information that is used in audit timestamps. The Audit events generated by the TOE include:

- Establishment of a Trusted Path or Channel Session
- Failure to Establish a Trusted Path or Channel Session
- Termination of a Trusted Path or Channel Session
- Failure of Trusted Channel Functions
- Identification and Authentication
- Unsuccessful attempt to validate a certificate
- Changes to trust anchors in the TOE's trust store
- Any update attempts
- Result of the update attempt
- Management of TSF data
- Changes to Time
- Session termination for inactivity
- Power-on self tests verification
- Changes to audit server configuration
- Users locked out due to failed authentication attempts.

The TOE can store the generated audit data on itself, and it can be configured to send syslog events to a syslog server, using a TLS protected collection method. Logs are classified into various predefined categories. The logging categories help describe the content of the messages that they contain. Access to the logs is restricted to only Security Administrators, who are authorized to edit them, copy, or delete (clear) them. Audit records are protected from unauthorized modifications and deletions.

The TSF provides the capability to view audit data by using the Syslog tab in the local console. The log records the time, host name, facility, application, and "message" (the log details). The previous audit records are overwritten when the allocated space for these records reaches the threshold on a FIFO basis.

1.3.2.2 Cryptographic Support

The TOE includes an OpenSSL library (Version 1.1.1k with Fedora Patches) that implements CAVP validated cryptographic algorithms for random bit generation, encryption/decryption, authentication, and integrity protection/verification. These algorithms are used to provide security for the TLS/HTTPs connections for secure management and secure connections to a syslog and authentication servers. TLS and HTTPs are also used to verify firmware updates. The cryptographic services provided by the TOE are described below:

Table 4 – TOE Cryptographic Protocols

Cryptographic Protocol	Use within the TOE
HTTPS/TLS (client)	Secure connection to syslog FCS_HTTPS_EXT.1, FCS_TLSC_EXT.1
HTTPS/TLS (server)	Peer connections to MAGNUM and remote management FCS_HTTPS_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2
AES	Provides encryption/decryption in support of the TLS protocol. FCS_COP.1.1/DataEncryption, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2

DRBG	Deterministic random bit generation use to generate keys. FCS_TLSS_EXT.1, FCS_TLSS_EXT.2, FCS_RBG_EXT.1
Secure hash	Used as part of digital signatures and firmware integrity checks. FCS_COP.1/Hash, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2
HMAC	Provides keyed hashing services in support of TLS. FCS_COP.1/KeyedHash, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2
EC-DH	Provides key establishment for TLS. FCS_CKM.2, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2
ECDSA	Provides components for EC-DH key establishment. FCS_CKM.1, FCS_CKM.2, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2
RSA	Provide key establishment, key generation and signature generation and verification (PKCS1_V1.5) in support of TLS. FCS_CKM.1, FCS_CKM.2, FCS_COP.1/SigGen, FCS_COP.1/SigVer, FCS_TLSC_EXT.1, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2

Each of these cryptographic algorithms have been validated for conformance to the requirements specified in their respective standards, as identified below and are part of the EXE Cryptographic Module.

Table 5 – CAVP Algorithm Testing References

Algorithm	Standard	CAVP Certificate #	Processors
AES 128/256-bit CBC, GCM, CTR	ISO 10116 (CBC and CTR) IOS 19772 (GCM)	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
CTR DRBG using AES 256	ISO/IEC 18031:2011	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
EC-DH P-256, P-384, P-521	NIST SP 800-56A (key establishment)	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
ECDSA P-256, P-384, P-521	FIPS PUB 186-4 (key generation)	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
HMAC-SHA-1/256/384	ISO/IEC 9797-2:2011	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
SHA-1/256/384	ISO/IEC 10118-3:2004	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E
RSA 2048/3072/4096	FIPS PUB 186-4 (key generation and Digital Signature) ISO/IEC 9796-2 (digital signature)	A2573	Intel$^{(R)}$ Core $^{(TM)}$ i3-6102E

1.3.2.3 Identification and Authentication

All Administrators wanting to use TOE services are identified and authenticated prior to being allowed access to any of the services other than the display of the warning banner. ("Regular" EXE users do not access EXE directly; they control IP video switching through the EXE using a switch control system, such as Evertz's Magnum. The switching of those IP video transport streams is outside the scope of the TOE.)

Once an Administrator attempts to access the management functionality of the TOE, the TOE prompts the Administrator for a username and password for password-based authentication. The identification and authentication credentials are confirmed against a local user database. Only after the Administrator presents the correct identification and authentication credentials will access to the TOE functionality be granted. If the user fails to provide the correct authentication credentials, the user will be locked out after a configurable threshold until the user is manually unlocked by an Administrator.

The TOE provides the capability to set password minimum length rules. This is to ensure the use of strong passwords in attempts to protect against brute force attacks. The TOE also accepts passwords composed of a variety of characters to support complex password composition. During authentication, no indication is given of the characters composing the password.

Remote administrators are locked out after a configurable number of unsuccessful authentication attempts.

The EXE requires a password-protected serial connection to perform initial configuration of the system IP address(es). Once each address is established, administrators use IP connectivity for all further administrative actions, including configuration, operations, and monitoring.

The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS/HTTPS connections.

1.3.2.4 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. All TOE administration occurs either through a secure session or a local console connection. The TOE provides the ability to perform the following actions:

- Administer the TOE locally and remotely;
- Configure the access banner;
- Configure the session inactivity time before session termination or locking;
- Update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- Specify the time limits of session inactivity;
- Ability to modify the IP address and the port of the remote syslog server;
- Generate Certificate Signing Requests, import and manage x509 certificates, delete/replace x509 certificates;
- Re-enable an Administrator account;
- Set the time which is used for timestamps.

All these management functions are restricted to Security Administrators who are authorized to administer the TOE via a local CLI and a remote web interface. Administrators are individuals who manage specific types of administrative tasks. The EXE implements role-based access control of these management functions to users that have been identified, authenticated, and authorized with the Security Administrator role.

Primary management is done using the Webeasy web-based interface using HTTPS. This provides a network administration console from which one can manage various identity services. These services include authentication, authorization, and reporting. All these services can be managed from the interface, which uses a menu-driven navigation system.

There is also a very simple serial-based connection (DB9) that provides a simple menu interface. This is used to configure the IP interface (IP address, etc.). It is password-protected, and is typically only used once, for initial set-up.

1.3.2.5 Protection of the TSF

The TOE will terminate inactive sessions after an Administrator-configurable time period. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session. The TOE provides protection of TSF data (authentication data and cryptographic keys). In addition, the TOE internally maintains the date and time. This date and time is used as the time stamp that is applied to TOE generated audit records. The TOE also ensures firmware updates are from a reliable source. Finally, the TOE performs testing to verify correct operation.

An administrator initiates update processes from the web interface for all update installations. EXE automatically uses the RSA digital signature mechanism to confirm the integrity of the product before installing the update.

1.3.2.6 TOE Access

Aside from the automatic Administrators session termination due to inactivity described above, the TOE also allows Administrators to terminate their own interactive session. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session.

The TOE will display an Administrator-specified banner on the web browser management interface prior to allowing any administrative access to the TOE.

1.3.2.7 Trusted Path/Channels

The TOE allows the establishment of a trusted channel between a video control system (such as Evertz' Magnum) and the EXE. The TOE also establishes a secure connection for sending syslog data to a syslog server using TLS.

The TOE uses HTTPS/TLS to provide a trusted path between itself and remote administrative users. The TOE does not implement any additional methods of remote administration. The remote administrative users are responsible for initiating the trusted path when they wish to communicate with the TOE.

1.3.3 TOE Documentation

The following documents are essential to understanding and controlling the TOE in the evaluated configuration:

- MMA10G-EXE Series II Security Target v1.2, August 12, 2024
- MMA10G-EXE Series II Security Administrative Guide Addendum for Common Criteria, version 1.5, August 12, 2024

1.3.4 References

In addition to TOE documentation, the following reference may also be valuable when understanding and controlling the TOE:

- collaborative Protection Profile for Network Devices, Version 2.2e , 27 March 2020 [PP-ND]

1.4 TOE Environment

The following environmental components are required to operate the TOE in the evaluated configuration:

Table 6 – Required Environmental Components

Components	Description
Syslog server	• Conformant with RFC 5424 (Syslog Protocol) • Supporting Syslog over TLS (RFC 5425) • Acting as a TLSv1.2 server • Supporting Client Certificate authentication • Supporting at least one of the following cipher suites: ○ TLS_RSA_WITH_AES_128_CBC_SHA ○ TLS_RSA_WITH_AES_256_CBC_SHA ○ TLS_RSA_WITH_AES_128_CBC_SHA256 ○ TLS_RSA_WITH_AES_256_CBC_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
Local Management Workstation	• Computer with terminal emulation software to access the console interface (CLI)
Remote Management Workstation with web browser	• Internet Explorer 11, Google Chrome 50, or Firefox 38 • Supporting TLSv1.2 • Supporting Client Certificate authentication • Supporting at least one of the following ciphersuites: ○ TLS_RSA_WITH_AES_128_CBC_SHA ○ TLS_RSA_WITH_AES_256_CBC_SHA ○ TLS_RSA_WITH_AES_128_CBC_SHA256 ○ TLS_RSA_WITH_AES_256_CBC_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
CRL Server	• Conformant with RFC 5280
MAGNUM Server	• Provides remote management of the TOE's routing and switching of video signals • Supporting TLSv1.2 with at least one of the following ciphersuites: ○ TLS_RSA_WITH_AES_128_CBC_SHA ○ TLS_RSA_WITH_AES_256_CBC_SHA ○ TLS_RSA_WITH_AES_128_CBC_SHA256 ○ TLS_RSA_WITH_AES_256_CBC_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 ○ TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

1.5 Product Functionality not Included in the Scope of the Evaluation

The following product functionality is not included in the CC evaluation:

- SNMP Traps (Alarms)

- SNMP
- VistaLINK PRO module
- Network Time Protocol (NTP) Server
- External Authentication Servers for administrator authentication

These functions are outside the TOE. Alarm monitoring is the sending of SNMP traps to an alarm monitoring system (which is assigned by an Administrator).

In addition, EXE provides IP video stream switching. This IP video switching does not provide security functionality and was therefore not evaluated and is outside the scope of the TOE. The nature of video encryption and decryption is that a video stream is encrypted at the sending end and decrypted at the receiving end; since EXE is a midpoint device and therefore does not perform encryption or decryption functionality. This functionality, while present in the TOE, was not evaluated.

2 Conformance Claims

This section identifies the TOE conformance claims, conformance rationale, and relevant Technical Decisions (TDs).

2.1 CC Conformance Claims

The TOE is conformant to the following:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017 (Extended)
- Common Criteria for Information Technology Security Evaluations Part 3, Version 3.1, Revision 5, April 2017 (Conformant)

2.2 Protection Profile Conformance

This ST claims exact conformance to the following:

- collaborative Protection Profile for Network Devices, Version 2.2e, 27 March 2020 [PP-ND]

2.3 Conformance Rationale

This ST provides exact conformance to the items listed in the previous section. The security problem definition, security objectives, and security requirements in this ST are all taken from the Protection Profile (PP), performing only the operations defined there.

2.3.1 Technical Decisions

All NIAP TDs issued to date and applicable to NDcPP v2.2e have been considered. Table 7 identifies all applicable TDs.

Table 7 – Relevant Technical Decisions

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0527: Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	
TD0528: NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	No	Not claimed in ST
TD0536: NIT Technical Decision for Update Verification Inconsistency	Yes	
TD0537: NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	Yes	
TD0546: NIT Technical Decision for DTLS - clarification of Application Note 63	No	Not claimed in ST.
TD0547: NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
TD0555: NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
TD0556: NIT Technical Decision for RFC 5077 question	Yes	

TD0563: NiT Technical Decision for Clarification of audit date information	Yes	
TD0564: NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
TD0569: NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	Yes	
TD0570: NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
TD0571: NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
TD0572: NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
TD0580: IT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	
TD0581: NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
TD0591: NIT Technical Decision for Virtual TOEs and hypervisors	Yes	
TD0592: NIT Technical Decision for Local Storage of Audit Records	Yes	
TD0631: NIT Technical Decision for Clarification of public key authentication for SSH Server	No	TOE does not claim SSH
TD0632: NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
TD0635: NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
TD0636: NIT Technical Decision for Clarification of Public Key User Authentication for SSH	No	TOE does not claim SSH Client
TD0638: NIT Technical Decision for Key Pair Generation for Authentication	Yes	
TD0639: NIT Technical Decision for Clarification for NTP MAC Keys	No	TOE does not claim NTP
TD0670: NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	Yes	
TD0738: NIT Technical Decision for Link to Allowed-With List	Yes	
TD0790: NIT Technical Decision: Clarification Required for testing IPv6	Yes	

TD0792: NIT Technical Decision: FIA_PMG_EXT.1 - TSS EA not in line with SFR	Yes	
TD0800: Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	No	TOE does not claim IPSec as a secure channel

3 Security Problem Definition

The security problem definition has been taken directly from the claimed PP specified in Section 2.2 and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any Organizational Security Policies (OSPs) that the TOE is expected to enforce.

3.1 Threats

The threats included in Table 8 are drawn directly from the PP specified in Section 2.2.

Table 8 – Threats

ID	Threat
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.
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.
T.UNTRUSTED_COMMUNICATION_CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling 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.
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.

ID	Threat
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.
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.
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.
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.
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.

3.2 Assumptions

The assumptions included in Table 9 are drawn directly from NDcPP.

Table 9 – Assumptions

ID	Assumption
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
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). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
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 the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).

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

3.3 Organizational Security Policies

The OSPs included in Table 10 are drawn directly from the NDcPP.

Table 10 – OSPs

ID	OSP
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.

4 Security Objectives

The security objectives have been taken directly from the claimed PP and are reproduced here for the convenience of the reader.

4.1 Security Objectives for the Operational Environment

Security objectives for the operational environment assist the TOE in correctly providing its security functionality. These objectives, which are found in the table below, track with the assumptions about the TOE operational environment.

Table 11 – Security Objectives for the Operational Environment

ID	Objectives for the Operational Environment
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.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. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. 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.
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.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.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. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

5 Security Requirements

This section identifies the Security Functional Requirements (SFRs) for the TOE. The SFRs included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revisions 5, September 2017, and all international interpretations.

Table 12 – SFRs

Requirement	Description
FAU_GEN.1	Audit Data Generation
FAU_GEN.2	User Identity Association
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_CKM.1	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Establishment
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_TLSC_EXT.1	TLS Client Protocol without Mutual Authentication
FCS_TLSS_EXT.1	TLS Server Protocol without Mutual Authentication
FCS_TLSS_EXT.2	TLS Server Support for Mutual Authentication
FIA_AFL.1	Authentication Failure Management
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_UAU.7	Protected Authentication Feedback
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FMT_MOF.1/Functions	Management of Security Functions Behaviour
FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
FMT_MTD.1/CoreData	Management of TSF Data
FMT_MTD.1/CryptoKeys	Management of TSF Data
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on security roles
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_TST_EXT.1	TSF Testing

FPT_STM_EXT.1	Reliable Time Stamps
FPT_TUD_EXT.1	Trusted Update
FTA_SSL.3	TSF-initiated Termination
FTA_SSL.4	User-initiated Termination
FTA_SSL_EXT.1	TSF-initiated Session Locking
FTA_TAB.1	Default TOE Access Banner
FTP_ITC.1	Inter-TSF Trusted Channel
FTP_TRP.1/Admin	Trusted Path

5.1 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 PP: the refinement text is indicated with **bold text** and ~~strikethroughs~~;
- Selection wholly or partially completed in the PP: the selection values (i.e. the selection values adopted in the PP 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 PP;

- Assignment wholly or partially completed in the PP: indicated with *italicized text*;
- Assignment completed within a selection in the PP: 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 PP;
- Iteration: indicated by adding a string starting with '/' (e.g. 'FCS_COP.1/Hash').

Extended SFRs are identified by having a label 'EXT' at the end of the SFR name.

5.2 Security Functional Requirements

This section includes the security functional requirements for this ST.

5.2.1 Security Audit (FAU)

5.2.1.1 FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1

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

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

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

c) *All administrative actions comprising:*

  - *Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).*

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

d) *Specifically defined auditable events listed in Table 12*.

FAU_GEN.1.2

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of* Table 13.

Table 13 – Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
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
FCS_COP.1/KeyedHash	None	None
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session	Reason for failure
FCS_RBG_EXT.1	None	None
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.2	Failure to authenticate the client	Reason for failure
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 identification and authentication mechanism	Origin of the attempt (e.g., IP address)
FIA_UAU_EXT.2	All use of identification and authentication mechanism	Origin of the attempt (e.g., IP address)
FIA_UAU.7	None	None

Requirement	Auditable Events	Additional Audit Record Contents
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
FIA_X509_EXT.3	None	None
FMT_MOF.1/Functions	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
FMT_MTD.1/CoreData	None.	None
FMT_MTD.1/CryptoKeys	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_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	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_SSL_EXT.1 (if “terminate the session” is selected)	The termination of a local session by the session locking mechanism	None
FTA_TAB.1	None	None

Requirement	Auditable Events	Additional Audit Record Contents
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. • Failure of the trusted path functions.	None

5.2.1.2 FAU_GEN.2 User Identity Association

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

5.2.1.3 FAU_STG_EXT.1 Protected Audit Event Storage

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. In addition [*The 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: [on a circular (FIFO) basis]*] when the local storage space for audit data is full.

5.2.2 Cryptographic Support (FCS)

5.2.2.1 FCS_CKM.1 Cryptographic Key Generation

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

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;*

] ~~and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].~~

5.2.2.2 FCS_CKM.2 Cryptographic Key Establishment

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";*

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

Application Note: This SFR has been updated as per TD0580 and TD0581

5.2.2.3 FCS_CKM.4 Cryptographic Key Destruction

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

- *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 [zeroes]];*

that meets the following: *No Standard.*

5.2.2.4 FCS_COP.1/DataEncryption Cryptographic Operations (AES Data Encryption/Decryption)

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

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

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

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 3072 bits, 4096 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*

].

5.2.2.6 FCS_COP.1/Hash Cryptographic Operations (Hash Algorithm)

FCS_COP.1.1/Hash
The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [*SHA-1, SHA-256, SHA-384*] ~~and cryptographic key sizes [*assignment: cryptographic key sizes*~~] and **message digest sizes [*160, 256, 384*] bits** that meet the following: *ISO/IEC 10118-3:2004*.

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

FCS_COP.1.1/KeyedHash
The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384*] and cryptographic key sizes *[160, 256, 384]* **and message digest sizes [*160, 256, 384*] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

5.2.2.8 FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3
If a peer certificate is presented, the TSF shall [*not establish the connection*] if the peer certificate is deemed invalid.

5.2.2.9 FCS_RBG_EXT.1 Random Bit Generation

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

FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*[two] software-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 hashes that it will generate.

5.2.2.10 FCS_TLSC_EXT.1 TLS Client Protocol without Mutual Authentication

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

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC5246*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

] and no other ciphersuites.

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

FCS_TLSC_EXT.1.3
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [*Not implement any administrator override mechanism*].

FCS_TLSC_EXT.1.4
The TSF shall [*present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups:* [*secp256r1, secp384r1, secp521r1*] *and no other curves/groups*] in the Client Hello.

5.2.2.11 FCS_TLSS_EXT.1 TLS Sever Protocol Without Mutual Authentication

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

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC5246*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC5289*

] and no other ciphersuites.

FCS_TLSS_EXT.1.2
The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [*TLS 1.1*].

FCS_TLSS_EXT.1.3
The TSF shall perform key establishment for TLS using [*RSA with key size* [*2048 bits*]*, ECDHE curves* [*secp256r1, secp384r1, secp521r1*] *and no other curves*].

FCS_TLSS_EXT.1.4
The TSF shall support [*no session resumption or session tickets*].

5.2.2.12 FCS_TLSS_EXT.2 TLS Sever Support for Mutual Authentication

FCS_TLSS_EXT.2.1
The TSF shall support TLS communication with mutual authentication of TLS clients using X.509v3 certificates.

FCS_TLSS_EXT.2.2
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also [*Not implement any administrator override mechanism*].

FCS_TLSS_EXT.2.3
The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifiers according to RFC 6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.

5.2.3 Identification and Authentication (FIA)

5.2.3.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1
The TSF shall detect when an Administrator configurable positive integer within *[3 to 20]* unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

FIA_AFL.1.2
When the defined number of unsuccessful authentication attempts has been met, the TSF shall [*prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until [an Administrator unlocks the user] is taken by an Administrator*].

5.2.3.2 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1.1
The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“~”, ”`”, ”_”, ”-“, ”+”, “=”, “{“, “[“, “}”, “]”, “|”, “\”, “:”, “;”, (”), (’), “<”, “,”, “>”, “.”, “?”, “/”, (space)]];

b) Minimum password length shall be configurable to between [*15*] and [*20*] characters.

5.2.3.3 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1
The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

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

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

5.2.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

5.2.3.5 FIA_UAU.7.1 Protected Authentication Feedback

FIA_UAU.7.1
The TSF shall provide only *obscured feedback* to the administrative user while the authentication is in progress **at the local console**.

5.2.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certification path validation **supporting a minimum path length of three certificates** .
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
    - *Server certificates presented for TLS shall have the Server Authentication purpose(id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
    - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsagefield.*
    - *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.

5.2.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1
The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*HTTPS, 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*].

Application Note: This SFR has been updated as per TD0537.

5.2.3.8 FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.2.4 Security Management (FMT)

5.2.4.1 FMT_MOF.1/Functions Management of Security Functions Behaviour

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

5.2.4.2 FMT_MOF.1/ManualUpdate Management of Security Functions Behavior

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

5.2.4.3 FMT_MTD.1/CoreData Management of TSF Data

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

5.2.4.4 FMT_MTD.1/CryptoKeys Management of TSF Data

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

5.2.4.5 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
    - *Ability to modify the behaviour of the transmission of audit data to an external IT entity;*
    - *Ability to manage the cryptographic keys;*
    - *Ability to re-enable an Administrator account;*
    - *Ability to set the time which is used for time-stamps;*
    - *Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;*
    - *Ability to import X.509v3 certificates to the TOE's trust store*].

5.2.4.6 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1
The TSF shall maintain the roles [*Security Administrator*].

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

FMT_SMR.2.3
The TSF shall ensure that the conditions [

- *The Security Administrator role shall be able to administer the TOE locally;*
- *The Security Administrator role shall be able to administer the TOE remotely;*

] are satisfied.

5.2.5 Protection of the TSF (FPT)

5.2.5.1 FTP_APW_EXT.1 Protection of Administrator Passwords

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

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

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

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

5.2.5.3 FPT_STM_EXT.1 Reliable Time Stamps

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*].

5.2.5.4 FPT_TST_EXT.1 TSF Testing

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: [

- *firmware integrity check that compares the SHA256 checksum of the loaded firmware with a permanently stored hash value;*
- *Cryptographic library tests:*
  - *SHA-256 KAT*
  - *HMAC-SHA-256 KAT*
  - *AES 128 GCM Encrypt and Decrypt KAT*
  - *RSA 4096 SHA-256 Sign and Verify KAT*
  - *ECDSA Pairwise Consistency Test)*
  - *DRBG AES-CTR-256 KAT (invoking the instantiate, reseed, and generate functions)*

].

5.2.5.5 FPT_TUD_EXT.1 Trusted Update

FPT_TUD_EXT.1.1
The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [*the most recently installed version of the TOE firmware/software*].

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

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

5.2.6 TOE Access (FTA)

5.2.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

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

5.2.6.2 FTA_SSL.3 TSF-initiated Termination

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

5.2.6.3 FTA_SSL.4 User-initiated Termination

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

5.2.6.4 FTA_TAB.1 Default TOE Access Banners

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

5.2.7 Trusted Path/Channels (FTP)

5.2.7.1 FTP_ITC.1 Inter-TSF Trusted Channel

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, [*video switch control system (such as Evertz MAGNUM)*]** 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 [*auditing services and system logging, controlling video switches*].

5.2.7.2 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin
The TSF shall **be capable of using [*TLS, HTTPS*] to** provide a communication path between itself and **authorized** remote **Administrators** that is logically distinct from other communication paths and

provides assured identification of its end points and protection of the communicated data from **disclosure and provides detection of modification of the channel data**.

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*.

5.3 TOE SFR Dependencies Rationale for SFRs

The PP contain(s) all the requirements claimed in this ST. As such, the dependencies are not applicable since the PP has been approved.

5.4 Security Assurance Requirements

The TOE assurance requirements for this ST are taken directly from the PP, which is/are derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in Table 14.

Table 14 – Security Assurance Requirements

Assurance Class	Assurance Components	Component Description
Security Target	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE Summary Specification
Development	ADV_FSP.1	Basic functionality specification
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative Procedures
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_IND.1	Independent testing – conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability survey

5.5 Assurance Measures

The TOE satisfied the identified assurance requirements. This section identifies the Assurance Measures applied by Evertz Microsystems Ltd. to satisfy the assurance requirements. The following table lists the details.

Table 15 – TOE Security Assurance Measures

SAR Component	How the SAR will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional

SAR Component	How the SAR will be met
	requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes).
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and how potential changes are incorporated.
ALC_CMS.1	
ATE_IND.1	Vendor will provide the TOE for testing.
AVA_VAN.1	Vendor will provide the TOE for testing. Vendor will provide a document identifying the list of software and hardware components.

6 TOE Summary Specification

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Table 16 – TOE Summary Specification SFR Description

Requirement	TSS Description
FAU_GEN.1 FAU_GEN.2	EXE generates audit logs that consist of various auditable events or actions. This includes logins, use of trusted channel/path and cryptographic operations. Each audit event contains an associated date/time stamp, a label for the type of event, a user ID (if applicable), a description of the event and any additional information specified in column three of Table 12. Audit records are created when an auditable event that belongs to a set of predefined events has occurred. The set of auditable events can be sub-categorized into functional events and access events. The TSF generates audit records for the following events: • Startup and shutdown of the audit function • Administrative login and logout events • Changes to TSF data related to configuration changes • Generation of a CSR and associated keypair • Installation, removal or replacement of a certificate • Resetting passwords • Failure to establish a HTTPS/TLS session • Failure to establish a TLS session • All use of the identification and authentication mechanism (local and remote connections to the TSF) • Unsuccessful attempts to validate a certificate • Initiation of a software update • Result of a software update • Changes to the time • Modification of the behavior of the TSF • Failure of self-tests • Initiation and termination of the trusted channel • Initiation and termination of the trusted path • Attempts to unlock an interactive session • Termination of a session by the session locking mechanism Each audit record includes the date and time, type, subject identity (IP address, hostname, and/or username), the outcome (success or failure), and any additional information specified in column three of Table 12. The TOE only stores one certificate chain to support TLS. No other server certificates are stored. In the logs of Administrator actions which involves cryptographic keys (generating or deleting keys), the audit log will refer to the key as the “server private key”.
FAU_STG_EXT.1	The TOE is a standalone TOE. EXE stores audit logs internally. The internal logs are stored unencrypted, but they are only accessible (and then read-only) via the web browser, which can only be used by Administrators. Logs are initially written to messages file on /var/log/ directory and then moved to /nv/syslog/current when /var/log is full. The size limit for /var/log/ folder depends on the size of the memory used on each model. This folder can also contain files other than messages (syslog files), hence, the amount of audit logs that can be saved in the /var/log/ directory can vary. The current audit log is saved in the file name

Requirement	TSS Description
	‘messages’. Once the current messages file reaches 60MB, it will be saved as messages.0 and a new messages file will be generated to capture the new audit logs. The full messages files will be written to messages.0, messages.1, and up to messages.10. As each messages.X file is created, it is archived and sent to the /nv/syslog/current/ directory. The /nv/syslog/current/ is in the hard disk and has a size limit of 880MB. The TOE overwrites previous audit records on a circular (FIFO) basis when both the volatile /var/log and persistent /nv/syslog/current storage space for audit is full. Logs information is also sent to an external Syslog server via ‘Syslog over TLS using TLS v1.2’. Logs are sent to the Syslog servers in real-time. For this to happen, an external syslog server should be configured (IP address/TCP Port number). A trusted certificate chain that is used to sign syslog server’s certificate must also be uploaded to EXE.The [EXE CC Admin Guide] explains how to configure this connection. The trusted channel with the Syslog server is described in greater detail in the FCS_TLSC_EXT.1 description.
FCS_CKM.1	The TSF supports generation of 2048-bits, 3072-bits, and 4096-bits RSA keys for digital signatures in support of TLS sessions (FCS_TLSC_EXT.1 and FCS_TLSS_EXT.2) and the server certificate (FIA_X509_EXT.3). Generation of EC keys with NIST curves of P-256 or P-384 or P-521 are also used to generate EC DH components for key establishment in TLS sessions (FCS_TLSC_EXT.1 and FCS_TLSS_EXT.2).
FCS_CKM.2	The TOE acts as both sender and recipient for elliptic curve Diffie-Hellman key establishment schemes that meet the following: • NIST Special Publication (SP) 800-56A revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” – for FCS_TLSC_EXT.1 connections to the audit server and FCS_TLSS_EXT.2 connections to the MAGNUM server. or • RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, “Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specification Version 2.1”. The TOE uses RSA-based key establishment for backwards compatibility for FCS_TLSC_EXT.1 connections to audit server and FCS_TLSS_EXT.2 connections to the MAGNUM server. In the case of a decryption error, the TOE response is dependent on the stage of the connection process. If the connection has not been established, the TOE prevents a connection from occurring. If the connection has already been established, the TOE drops the packet(s) in question and logs the error internally.
FCS_CKM.4	Cryptographic keys are destroyed by first overwriting the key file content with zeros. A read-verification is then performed to ensure that the entire content has really been changed to zeros and not any other values. If these steps fail, then the file will be overwritten again with zeros until the read-verify step succeeds. A sudden, unexpected power could disrupt zeroization and cause keys to not be

Requirement	TSS Description
	zeroized. There are no other known circumstances where the TOE would not conform to these requirements. The keys/CSPs used by the TOE, their storage location and format, and their associated zeroization method are as below: • EC Diffie-Hellman Keys ○ Storage location and method: *Plaintext in RAM* ○ Usage: *Key agreement and key establishment* ○ Zeroization: *Overwritten with zeroes when no longer needed.* • Firmware Update Key ○ Storage location and method: *Public key is stored in plaintext in the Flash disk. Private key is not stored or used on the TOE.* ○ Usage: *Verification of firmware integrity when updating to new firmware versions using a SHA-256 hashed Public Key RSA signature.* ○ Zeroization: *Public key in non-volatile storage (RAM) is automatically replaced once new firmware is booted. Public key, which is part of non-volatile firmware image is replaced with new firmware image when the new image is installed on top of the existing image slot. zeroize does not act on this file.* • HTTPS/TLS Server/Host Key ○ Storage location and method: *Plaintext in RAM.* ○ Usage: *RSA and EC private key used in the HTTPS/TLS protocols* ○ Zeroization: *During boot they get erased. When the client closes TLS session, the keys get erased.* • HTTPS/TLS session authentication key ○ Storage location and method: *Plaintext in RAM.* ○ Usage: *HMAC SHA*-1, -256, or -384 key used for HTTPS/TLS session authentication. ○ Zeroization: *During boot they get erased. When the client closes TLS session, the keys get erased.* • HTTPS/TLS Session Encryption Key ○ Storage location and method: *Plaintext in RAM.* ○ Usage: *AES (128, 256) key used for HTTPS/TLS session encryption* ○ Zeroization: *During boot they get erased. When the client closes TLS session, the keys get erased.* • Locally Stored Passwords ○ Storage location and method: *SHA-256 Hashed in configuration file* ○ Usage: *User Authentication* ○ Zeroization: Temporary copy is created, modified, and replace the old file when no longer needed. • Configuration Encryption Key ○ Storage location and method: *Plaintext in the Flash Disk* ○ Usage: *Configuration Encryption* ○ Zeroization: Temporary copy is created, modified, and replace the old file when no longer needed. To delete the plain-text keys stored on the non-volatile NOR flash storage, direct interface/access is provided to view or modify the contents of these files. The CLI

Requirement	TSS Description
	provides Security Administrators with a menu item to destroy all CSPs, which would initiate key destruction. No direct interface/access is provided to view or modify the contents of the keys stored in the volatile memory. The TLS session keys stored on Flash are automatically destroyed when the TLS session ends. The DRBG state is zeroized using a single overwrite of zeros when the TSF is shutdown or restarted. The above destruction methods are followed in all configurations and circumstances.
FCS_COP.1/DataEncryption	The TOE provides AES encryption/decryption in CBC, CTR, or GCM mode with 128- and 256-bit keys.
FCS_COP.1/Hash	The TOE implements hashing in byte-oriented mode. The TOE provides cryptographic hashing services in support of TLS for SHA-1, SHA-256 and SHA-384. SHA-256 is used in firmware integrity checks during power-on-self-tests and upgrade where the hashed-value of the images is signed with Evertz's private key and the result file (signature) is included in the firmware package file. During upgrade, the signature file is first decrypted using the public key stored on EXE, then the hashed value is re-calculated from the uploaded image file and then compared with the decrypted hash value. These hashes must match for this validation to succeed. The locally stored passwords are salted using SHA-256. Key generation is performed using SHA-256 as specified in NIST SP 800-90 DRBG.
FCS_COP.1/KeyedHash	Keyed-hash message authentication is used as part of TLS protocol as part of the negotiated cipher suites between peers. It is also used for firmware image integrity check during the self-test of integrity of the ssl libraries such as libcrypto and libssl. During the initiation of the library, the existing HMAC-SHA256 of the library is compared with the HMAC-SHA256 value generated during run time. If there is a mismatch, the self-test fails and the TOE boot will fail. The following keyed-hash message authentication are used by EXE: • HMAC-SHA-1 with 160-bit keys, message digest size of 160 bits and 160-bits message block size, • HMAC-SHA-256 with 256-bit keys, message digest sizes of 256 bits, and block size of 512 bits, and • HMAC-SHA-384 with 384-bit keys, message digest sizes of 384 bits, and block size of 1024 bits.
FCS_COP.1/SigGen	The TOE supports signature generation and verification with RSA (2048-bits, 3072-bits, and 4096-bits) with SHA-1/256/384 in accordance with FIPS PUB 186-4. These signatures support TLS authentication and firmware verification. The TOE's server certificate is always 2048-bits.

Requirement	TSS Description
FCS_HTTPS_EXT.1 FCS_TLSC_EXT.1 FCS_TLSS_EXT.1 FCS_TLSS_EXT.2	The TOE acts as a TLS/HTTPS server to provide web access to administrators. The TOE's HTTPS functionality is in accordance with all should statements in RFC 2818. The TOE also acts as a TLS server when connecting to a video switch control system. For video switch control systems TLS trusted channels, the TOE requires TLS with mutual authentication. The TOE acts as a client when connecting to the syslog server and as a server when providing administrative access via TLS/HTTPS. The TSF only supports TLSv1.2 for HTTPS/TLS. Connection requests that include SSL 2.0, SSL 3.0, TLS 1.0 or TLS 1.1 are denied. If the TSF receives a ClientHello message that requests TLSv1.1 or earlier, the TSF sends a fatal handshake_failure message and terminates the connection. For all the TLS client and server connections, with the exception of 'revocation status verification failures', if the certificate verification fails for any other reason (including a failure to establish a connection), the connection attempt fails, and the trusted channel is not established. There are no fallback authentication functions for failed certificate authentication. TLS v1.2 is used to export audit records, communicate with a video switch control system, and support remote administrator connections. Server-side and client-side TLS work the same. EXE acts as the server in connections with remote administrators and with the MAGNUM video switch controller. EXE specifies only a restricted set of cipher suites that it supports during the negotiation phase with a client or a server. If no match of cipher suites can be found with peer, TLS session will not be started. These ciphersuites cannot be configured or changed by an Administrator. The following cipher suites are supported: • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_RSA_WITH_AES_256_CBC_SHA • TLS_RSA_WITH_AES_128_CBC_SHA256 • TLS_RSA_WITH_AES_256_CBC_SHA256 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 Protocols that do not conform to TLS v1.2 are explicitly excluded in EXE's cipher suites. EXE supports cipher suites that use ECDHE and RSA schemes for key exchange and RSA keys for authentication. These keys are generated with OpenSSL's RSA command line internally to the TSF. The elliptic curve Diffie Hellman and RSA are supported for key establishment in TLS for both client and server. The RSA key establishment uses keys with key-sizes 2048 bits, 3072 bits, and 4096 bits. EC-DH key establishment uses NIST curves, P-256, P-384, and P-521. By default, the TOE presents the supported Elliptic Curve Extensions, secp256r1, secp384r1, and secp521r1 in the Client Hello. The TOE conforms to RFC 5246, section 7.4.3 for key exchange. When validating a server's certificate or the client's certificate in mutual authentication, EXE uses CRL (certification revocation list) to check for invalid certificates. The TOE pulls the CRL file from the CRL-DP to use by EXE during

Requirement	TSS Description
	certificate validation process to check for revocation status of the peer certificates. EXE allows configuration of an RFC 6125 reference identifier from a peer it expects to connect with before connection is made. The reference identifier is matched to either the CN or the SAN in the certificate presented for authentication. The verification against peer certificate is implemented within OpenSSL using a bitwise comparison of the DN and SAN-DNS field. IP addresses are not supported as reference identifiers. EXE supports FQDN identifier types only. SRV-ID and URI-ID types are not supported. EXE does not support certificate pinning. EXE supports wildcard in certificates. The wildcard must be in the left-most label of the presented identifier and can only cover one level of subdomains. For the reference identifier without a left-most label as in the certificate, the connection will fail, i.e., awesome.com doesn't match *.awesome.com. EXE does not support session resumption based on session IDs or session tickets.
FCS_RBG_EXT.1	The TOE implements a DRBG in accordance with ISO/IEC 18031:2011 using a CTR DRBG with AES. The TSF seed the CTR_DRBG using 384-bits of data that contains at least 359 bits of entropy. The TSF gathers and pools entropy from two software-based noise sources: haveged and the Linux kernel provided entropy. The entropy sources are discussed in greater detail in the Entropy documentation.
FIA_AFL.1	An administrator can configure the number of unsuccessful attempts a remote administrator can make before a lock-out occurs. The attempts can range between 3 and 20. The default number of attempts is 10. If the user enters an incorrect password the configured number of times, the user is locked out and they cannot login through any remote interface on the TOE. The username will show the Lockout enabled on the Settings->Users page on the web interface. Users must have an administrator unlock their account before they can regain access. Administrators can also have a different administrator unlock their account. Lockouts are not enforced on the TOE's console interface. This ensures that authentication failures cannot lead to a situation where no administrator access is available.
FIA_PMG_EXT.1	EXE enforces that passwords must meet minimum requirements such as length, mix of number, lower/upper case letters, and the following special characters "!"; "@"; "#"; "$"; "%"; "^"; "&"; "*"; "("; ")";"~"; "`"; "_"; "-"; "+"; "="; "{"; "["; "}"; "]"; "|"; "\"; ":"; ";"; ["]; [']; "<"; ","; ">"; "."; "?"; "/"; [space]. At least two characters from each category are required (upper case letter, lower case letter, number special character). Passwords must be at least a minimum length settable by the administrator and support 15 to 20 characters.
FIA_UIA_EXT.1 FIA_UAU_EXT.2	The only accounts that the EXE will establish are Security Administrator accounts. Administrators are identified and authenticated via username and password prior to performing any operations other than acknowledging the warning banner. The EXE Administrators user accounts module maintains Security Administrator credentials. Security administrators can have different management roles assigned to create restricted users.

Requirement	TSS Description
	Administrators can log on via the web interface using HTTPS or locally on the serial port. A username and a password is required to authenticate the administrator for both methods. The Security Administrator is considered authenticated if the username and password match the stored credential values. For serial console, only the default ‘recovery’ user has access to the restricted shell. Prior to successful identification and authentication, the TSF displays the TOE access banner specified in FTA_TAB.1. Users must acknowledge the warning banner before they can login to the system.
FIA_UAU.7	When the user is entering their password over the local console, the TSF does not echo any characters back.
FIA_X509_EXT.1/Rev	EXE uses OpenSSL for X.509 certificate validation. The certificate path is validated by ensuring that all the CA certificates have the basicConstraints extension and the path must terminate with a trusted CA certificate. The extendedKeyUsage on each certificate is also checked to ensure there is no inappropriate usage. Server certificates must have the Server Authentication purpose, client’s certificates must have the Client Authentication purpose. Certificates for code signing and OCSP signing are not used or accepted by the TOE. Each certificate (other than the leaf certificate) in the certificate chain has the Subject Type=CA flag set. Certificates are not used for any purposes other than establishing TLS sessions. If certificates are uploaded to EXE for its own use those certificates are checked upon upload. When the TOE acts as a server for the Web GUI, it does not perform verification of its server certificate. When the TOE acts as a server for the Synergy (Magnum connection), it performs verification of every certificate in the chain, including its own certificate. The TOE’s client certificate is validated prior to use for authentication as well as upon upload. The certificate presented by remote TLS clients using mutual authentication is validated during the establishment of a TLS connection. The full certificate chain presented by TLS servers is validated during the establishment of a TLS connection. For an expired certificate, EXE will deny the connection. EXE also uses CRL to verify whether the leaf certificate or intermediate CA certificate have been revoked. During session establishment with EXE, any byte modification in the certificate will lead to the failure of connection. The TSF verifies the validity of a certificate when: • A TLS client establishes a TLS connection with mutual authentication. • A TLS server presents certificates to the TOE as a part of a TLS connection. If the Security Administrator loads a certificate with a Subject Type=CA, the TSF does not validate the certificate path.
FIA_X509_EXT.2	Instructions about generating/downloading CSR and loading certificate can be found on [EXE CC Admin Guide]. The Administrator can only upload one certificate chain to include a single CA certificate. The same certificate will be used by EXE for both web service and MAGNUM control. The same CA will be used for certificate verification. EXE enforces mutual authentication and therefore requires client certificates to establish a connection.

Requirement	TSS Description
	The CRLs are obtained from a CRL distribution point over HTTP and are refreshed according to the default CRL update-interval. If the TOE is unable to reach the CRL DP it will accept the certificate and the session associated with the certificate will be established, however, a log is generated indicating the reason for validation failure.
FIA_X509_EXT.3	The TSF allows Security Administrators to generate Certificate Signing Requests. The TSF generates a CSR with following values: • Common Name = <Management IP of the TOE> • Organization = Evertz Microsystems Ltd • Organization Unit = EXE • Locality = Burlington • State = Ontario • Country = CA • Key Length (2048) CSR can be generated only via the webGUI. When validating certificates, each certificate from the chain is sequentially validated, terminating at the root CA. If any invalid certificate is found in this process, the validation fails.
FMT_MOF.1/Functions FMT_MOF.1/ManualUpdate FMT_MTD.1/CoreData FMT_MTD.1/CryptoKeys FMT_SMF.1 FMT_SMR.2	EXE gives the Security Administrator the ability to manage the security functions: auditing operations, administrative user accounts, password and session policies, advisory banners, software updates, as well as cryptographic functions. EXE ensures that only secure values are accepted for security attributes. A Security Administrator can change passwords, and can add, edit and/or delete Security Administrator accounts. The (non-administrative) User has no direct access or control over EXE; a (non-administrative) User may only access an EXE card through MAGNUM. The (non-administrative) User can only view configurations. No administrative functionality is available prior to login. The TSF displays a warning banner prior to user authentication. Information on how a Security Administrator can manage Audit Operations is described in this table under FAU_STG_EXT.1 above. The TSF implements the Security Administrator role to authorized administrators of the TOE. The TSF allows the Security Administrators to administer the TSF via a local CLI and a remote web interface. The TSF implements role-based access control of these management functions to users that have been identified, authenticated, and authorized with the Security Administrator role. The web interface allows the Security Administrator to perform the following TSF management functions: • Edit login banner; • Create certificate signing request CSR, download a CSR; • Zeroize all Critical Security Parameters (CSP); • Import certificates; • Import Trusted CA certificate; • Delete (Replace) x509 certificates in the trust store; • Configure webGUI and console menu system timeout; • Verify/Install Firmware Updates; • View/Edit settings for sending audit data to the Syslog Server; • View/Edit authentication failure parameters; • Unlock a locked user after the login failure threshold is exceeded;

Requirement	TSS Description
	The following can only be performed from the console interfaces: • Configure EXE date and time; • Control port IP configuration; • Reset certificates. The TOE maintains a trust store where the TOE's certificate is stored. Only Security Administrators have access to the trust store. Security Administrators can upload a certificate chain. Uploading the certificate chain replaces the previously installed certificate chain. When a user account is created (by administrator), it must be assigned with a role that specifies the privileges the account will have. The administrator can choose to assign an existing role with pre-defined privileges or create a new role with customized privileges. Administrators can administer EXE locally through serial port connection. A console menu can be used to perform configurations tasks such as setting IP/system time/system reboot, etc. Administrators can administer EXE remotely through its web interface, which runs on HTTPS. The web interface supports a broader set of configuration settings that include configurations for certificate imports, syslog server, route mapping, etc. The CLI allow the Security Administrator to perform the following TSF management functions on cryptographic keys: • TLS Key Generation (TLS keys are automatically generated when creating a CSR) • TLS Key Reset/Replacement (when a CSR is generated, previous TLS key will be deleted and replaced by the new key. The TLS keys cannot be imported from outside the TOE. The administrators cannot delete TLS keys manually). The administrative interfaces provided by the TSF do not allow any of these functions to be accessed by unauthenticated or unauthorized users.
FPT_APW_EXT.1	The TSF does not store plaintext passwords. Passwords are hashed using SHA-256 and stored in a secure location which is not accessible to users. Secure (one-way) hash functions ensure that it's computationally impossible to recover a plaintext from its hashed value.
FPT_SKP_EXT.1	The TSF stores cryptographic keys in a directory. As there is no command line access, users cannot gain any direct access to these files. Information regarding the storage locations, usage, and method of storage of the cryptographic keys described in FCS_CKM.4 above.
FPT_STM_EXT.1	The TSF provides a reliable timestamp from the hardware clock on the TOE. Timestamps found in auditable log events use the system clock on EXE. In addition to the purpose of generating audit logs, this timestamp is used for the purposes of other time-sensitive operations on the TOE including cryptographic key regeneration intervals. Administrators can, as needed, set the system time clock through serial port console menu after each card reboot.

Requirement	TSS Description
	Other functions which make use of timestamps include verification of X.509 certificate validity periods. The new system time is also used to set the hardware clock, which is a clock that runs independently of any control program running in the CPU and even when EXE is powered off. During EXE system startup, system time is initialized to the time from the hardware clock.
FPT_TST_EXT.1	The TSF performs the following hardware self-tests at power-on: • firmware integrity check that compares the SHA256 checksum of the loaded firmware with a permanently stored hash value. The TSF enables FIPS mode on the OpenSSL library when Secure Mode is configured. Upon enabling FIPS mode the algorithm self-tests required by FIPS are performed. The OpenSSL library self-tests include: • Cryptographic library tests: ○ SHA-256 KAT ○ HMAC-SHA-256 KAT ○ AES 128 GCM Encrypt and Decrypt KAT ○ RSA 4096 SHA-256 Sign and Verify KAT ○ ECDSA Pairwise Consistency Test ○ DRBG AES-CTR-256 KAT (invoking the instantiate, reseed, and generate functions) After loading the image, a hash value is computed from the memory partition containing the image. This hash value is compared with a pre-stored hash value at another location on flash. The two hash values must match for the boot process to succeed. If any of the other checks fail, the TOE will fail to boot, and an error will be displayed. Administrators are instructed to contact Evertz service department for repair if the failure does not clear on reboot. These self-tests ensure the TOE software is the correct image and that cryptographic functions are performing appropriately. If failures are seen by the Administrator, they should be immediately corrected.
FPT_TUD_EXT.1	The site administrators do not have access to install any applications on the TOE. The EXE embedded system can only be updated with the valid firmware released by Evertz. The current firmware version is displayed on both webpage and in serial console menu. The TOE supports delayed activation of updates hence the inactive versions can be manually set to active by the Security Administrator on the web. Firmware updates are done from the EXE webGUI under “upgrade”. Digital delivery of new EXE firmware and updates may be provided via Microsoft OneDrive, which requires secure authentication and uses secure TLS (HTTPS). The first step of upgrading firmware image is to choose a desired image slot index. If any firmware image is pre-installed to the desired image slot, delete it. Continue using the image slot for firmware upload. After the firmware is uploaded, EXE will verify the firmware binary header with an Evertz-EXE-specific-file format header. If there is no mismatch, the new firmware code will be parsed for valid digital signatures. Verification of the firmware’s digital signatures is performed using the public key stored on EXE. If unsuccessful, the firmware update file is rejected, and an error is

Requirement	TSS Description
	displayed. The TSF does not provide an interface to change the local stored public key to administrators. If successful, firmware specific files are generated. Checksums of the firmware binary and firmware specific files are generated and stored under the image slot chosen when uploading the firmware binary. The generated checksum is used to verify the firmware binary copy to the image slot location without compromising the integrity of the files. Once the desired image slot upgrade with the firmware binary is completed successfully, the administrator must manually change the "Next Boot Image" value from the current boot image to the newly installed image slot. On setting the next image, the firmware binary is extracted to the location that will be used during boot. Once extraction is complete, boot specific files are created. Checksums for the extracted firmware files and boot specific files are created. The generated checksum is used to verify if the firmware files are extracted without compromising the integrity of the files. If the digital signature fails, the upgrade fails, and a log event is generated. If the digital signature succeeds, the upgrade proceeds and the updated firmware is installed onto the TOE. The administrator must manually reboot for the new update to take effect.
FTA_SSL_EXT.1 FTA_SSL.3 FTA_SSL.4	Security Administrators can configure a maximum allowable period of inactivity for a Security Administrator session on the web interface. The settings made on the web interface are applied to both local console and web interfaces. If there is no user interaction with the EXE for the specified amount of time, the session is terminated. The initial, default session timeout is 15 minutes. When the session is terminated, any unsaved changes will be discarded. Administrators may terminate their own sessions by clicking "Logout" at the upper right hand of the web interface or by exiting the top-level menu on the console.
FTA_TAB.1	The TSF presents the access banner prior to authentication when a user connects to the remote web interface or local console CLI described in the FIA_UIA_EXT.1, FIA_UAU_EXT.2 description. The TSF enables Security Administrators to alter the warning banner by navigating to the Perpetual User License Agreement tab on the web. From here the Security Administrator can modify the "Agree" text and/or the "Disagree" text. (The "Disagree" text shows up when a user "disagrees" with the Security Banner text. The banner can provide warnings against unauthorized access to the TOE as well as any other information that the Security Administrator wishes to communicate. Users who select "Disagree" are not permitted access to the TSF.
FTP_ITC.1	The TSF communicates with the external syslog server using Syslog over TLSv1.2 as described in the descriptions of FAU_STG_EXT.1 and FCS_TLS* above. The TSF initiates the trusted channel with the Syslog server. The TSF communicates with a MAGNUM server (Video Switch Server) through TLS as well as described in the FCS_TLS* above. The MAGNUM server initiates the trusted channel with the TOE and is a trusted IT entity.
FTP_TRP.1/Admin	The TSF provides a trusted path for remote administration using HTTPS/TLS as described in FCS_HTTPS_EXT.1 and FCS_TLSS_EXT.1 descriptions. EXE uses encryption and restricts the choices of ciphers, hashes, and key-exchange algorithms to those allowed by the NDcPP.

7 Acronym Table

Acronyms should be included as an Appendix in each document.

Table 17 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
CC	Common Criteria
CRL	Certificate Revocation List
DTLS	Datagram Transport Layer Security
EP	Extended Package
GUI	Graphical User Interface
IP	Internet Protocol
NDcPP	Network Device Collaborative Protection Profile
NIAP	National Information Assurance Partnership
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
PP	Protection Profile
RSA	Rivest, Shamir & Adleman
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TLS	Transport Layer Security
TSS	TOE Summary Specification