Ixia

Network Tool Optimizer 7303 and Vision ONE

v4.5.0.29

Security Target

Document Version: 1.2

Prepared for:

ixia

Ixia
26601 W. Agoura Road
Calabasas, CA 91302
United States of America

Phone: +1 818 871 1800
www.ixiacom.com

Prepared by:

Corsec

Corsec Security, Inc.
13921 Park Center Road
Suite 460
Herndon, VA 20171
United States of America

Phone: +1 703 267 6050
www.corsec.com

Table of Contents

List of Figures

List of Tables

1. Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), and the organization of the ST. The TOE is the Ixia NTO 7303 and Vision ONE appliances and will hereafter be referred to as the TOE. The NTO 7303 is also referred to as Vision 7303. The TOE is a network visibility tool (also known as a network packet broker).

1.1 Purpose

This ST is divided into ten sections, as follows:

- Introduction (Section 1) – Provides a brief summary of the ST contents and describes the organization of other sections within this document. It also provides an overview of the TOE security functionality and describes the physical and logical scope for the TOE as well as the ST and TOE references.
- Conformance Claims (Section 2) – Provides the identification of any Common Criteria (CC), Protection Profile (PP), and Evaluation Assurance Level (EAL) package claims. It also identifies whether the ST contains extended security requirements.
- Security Problem (Section 3) – Describes the threats, Organizational Security Policies (OSPs), and assumptions that pertain to the TOE and its environment.
- Security Objectives (Section 4) – Identifies the security objectives that are satisfied by the TOE and its environment.
- Extended Components (Section 5) – Identifies new components (extended Security Functional Requirements (SFRs) and extended Security Assurance Requirements (SARs)) that are not included in CC Part 2 or CC Part 3.
- Security Assurance Requirements (Section 6) – Presents the SARs met by the TOE.
- Security Functional Requirements (Section 7) – Presents the SFRs met by the TOE.
- TOE Summary Specification (Section 8) – Describes the security functions provided by the TOE that satisfy the security functional requirements and objectives.
- Rationale (Section 9) - Presents the rationale for the SFR dependencies as to their consistency, completeness, and suitability.
- Acronyms and Terms (Section 10) – Defines the acronyms and terminology used within this ST.

1.2 Security Target and TOE References

Table 1 below shows the ST and TOE references.

Table 1 – ST and TOE References

ST Title	Ixia NTO 7303 and Vision ONE v4.5.0.29 Security Target
ST Version	Version 1.2
ST Author	Corsec Security, Inc.
ST Publication Date	October 23, 2017
TOE Reference	Ixia NTO 7303 and Vision ONE v4.5.0.29

1.3 Product Overview

The Product Overview provides a high-level description of the product components (NTO 7303 appliance, Vision ONE appliance, and NTO firmware) that are the subject of the evaluation. Each of these components will be referred to as 7303, Vision ONE, and NTO firmware respectively throughout this document. The capabilities of the product components described in the Product Overview were not subject to evaluation during the ND[1] cPP[2] v1.0 evaluation of the TOE. The following section, TOE Overview, will provide the introduction to the parts of the overall product offering that are specifically being evaluated.

The 7303 and Vision ONE appliances are deployed between a customer's data center network and monitoring tools via SPAN[3] ports and TAP[4]s. The NTO firmware that runs on the appliances is responsible for aggregating, filtering, replicating, stripping, trimming, masking, timestamping, and de-duplicating network traffic before it reaches the monitoring tool farm. The NTO firmware also performs load balancing, spreading network traffic evenly across multiple monitoring tools while keeping sessions intact. All these features help to eliminate SPAN and TAP shortages, increase network visibility, and improve the overall performance of the monitoring tools.

1.3.1.1 NTO 7303 Appliance

The 7303 is a NEBS[5]-level 3 certified chassis-based high port density NTO platform. It supports up to 384 1G[6]/10G SFP+[7] or 96 40G QSFP+[8] network and tool ports and is composed of the following elements:

- Two supervisor modules that host the firmware that runs on the NTO 7303 chassis and controls the line cards. Supervisors also expose the management interface that administrators use to manage the TOE.
- Up to six of the following line cards, which provide the front panel ports for transceivers:
  - **48-port 1G/10G SFP/SFP+ line card**
  - **48-port 1G/10G SFP+ ATIP[9] line card** – provides Layer 4-7 filtering, application detection and discovery, geolocation identification, network inspection for security, NetFlow, and SSL decryption.
  - **48-port 1G/10G SFP+ PCM[10] line card** – PCM can be used to capture and display packets at dynamic filters.
  - **16-port 40G QSFP+ (or 64 1/10G SFP+) line card**
  - **4-port QSFP+ Carrier line card** – includes two cassette slots for advanced hardware features such as the following:
    - 4-port 40G (QSFP+) 16-Port 10G AFM[11] Cassette
    - 100G Port Interface Cassette with support for a single 100G CFP[12] port
    - 16-port 1G/10G AFM Cassette which can be licensed for 1/10G SFP+ or 1G/10G Advanced features

[1] ND – Network Device
[2] cPP – Collaborative Protection Profile
[3] SPAN – Switched Port Analyzer
[4] TAP – Test Access Point
[5] NEBS – Network Equipment-Building System
[6] G – Gigabit
[7] SFP+ – Small Form-Factor Pluggable Enhanced
[8] QSFP+ – Quad Small Form-Factor Pluggable Enhanced
[9] ATIP – Application and Threat Intelligence Processor
[10] PCM – Packet Capture Module
[11] AFM – Advanced Feature Module
[12] CFP – C Form-Factor Pluggable

- **Smart blank module line card** – used when a chassis slot is not in use.

With dual supervisor modules the 7303 has 3.84Tb[13] backplane capacity. Each line card is connected to a supervisor module via an 8x40G interface. The links from the line cards to the supervisor modules are configured into a logical 640G trunk interface on which load balancing is done.

The NTO 7303 appliances relies on two processors to run the NTO and ATIP firmware. The supervisor modules and ATIP line card include the Intel Core i7-3555LE processor, and the remaining line cards (except for the smart blank module line card) include the Intel Atom N2600 processor.

Figure 1 – 7300/7303 Appliance

The 7303 system provides an orthogonal switching architecture. Data packets flow from a network (ingress) line card to a tool (egress) line card via the supervisor modules. After entering a network port, the packets go through a filtering process. If the packets are destined to another destination line card, they are forwarded from the line card to one of the supervisor modules, based on a load balancing decision. After having been switched in the supervisor module, packets are forwarded to the tool ports where they are filtered again as they leave the system on the tool ports.

1.3.1.2 Vision ONE Appliance

The Vision ONE is a 1RU[14] non-interchangeable small form factor NTO platform that provides 48 1G/10G SFP+ and 4 QSFP+ data ports (or 16 1/10G SFP+ ports). The Vision ONE provides aggregation, filtering, replication, and load balancing features. The integrated ATIP can be attached to any dynamic filter in the system to provide Layer 4-7 filtering, application detection and discovery, geolocation identification, and network inspection. Vision ONE also

[13] Tb – Terabit
[14] RU – Rack Unit

includes a 150GB[15] Advanced Features Resource that can be configured on any port or dynamic filter to provide advanced feature capabilities including packet stripping, trimming, masking, timestamps, and de-duplication.

Figure 2 – Vision ONE Appliance

Network packets flow through the Vision ONE through network ports, then through dynamic filters, and finally out through tool ports.

The Vision ONE includes the Intel Core i7-3555LE processor (for control and the ATIP functionality).

1.3.1.3 NTO Firmware

The NTO firmware is preinstalled on the 7303 and Vision ONE appliances and provides configuration and network optimization features. These features include:

- Packet processing techniques including de-duplication and packet trimming
- Deep packet inspection to classify traffic in real time and direct it to the correct tool according to configured parameters
- Load balancing to distribute traffic across monitoring tools for network traffic optimization

The main firmware component is the Java and Linux-based Network Visibility Operating System (NVOS) v4.5, which provides filtering, load balancing, network traffic optimization, user management, and port management support. The ATIP firmware (v1.4.1) that runs on the ATIP module utilizes dynamic application discovery and static traffic pattern identification to provide application-level detection, filtering, and monitoring.

The supervisor on both the 7303 and Vision ONE appliances includes a Java based management stack that provides access to the NTO Web GUI[16] and NTO Java Console.

1.4 TOE Overview

The TOE Overview summarizes the usage and major security features of the TOE. The TOE Overview provides a context for the TOE evaluation by identifying the TOE type, describing the product, and defining the specific evaluated configuration.

The TOE is a firmware and hardware TOE consisting of the NTO 7303 appliance, Vision ONE appliance, and NTO firmware that runs on the appliances. The NTO 7303 appliance includes the following line cards in the evaluated configuration:

- 48-port 1G/10G SFP+ ATIP line card
- 48-port 1G/10G SFP+ PCM line card
- 16-port 40G QSFP+ line card

[15] GB – Gigabyte
[16] GUI – Graphical User Interface

- 4-port QSFP+ Carrier line card with two 16-port 10G AFM Cassettes
- 4-port QSFP+ Carrier line card with two 100G CFP Cassettes
- Smart blank module line card

The Vision One appliance is a fixed-configuration platform and does not include line cards. The TOE provides the NTO Web Console and NTO Java Console management interfaces. The NTO appliances are typically deployed adjacent to customer networks.

1.4.1 TOE Environment

The TOE relies on non-TOE hardware/software for its essential operation. Though this hardware/software is necessary for the TOE's operation, it is not part of the TOE. The following non-TOE hardware/software is required for essential operation of the TOE:

- NTP[17] server
- Syslog server
- DNS[18] server
- Local management workstation with Ethernet port
- Remote Windows 7 64-bit management workstation with Google Chrome (version 60) web browser and 64-bit JRE[19] 1.8
- 1 RJ45 Crossover Ethernet cables for direct connections to the management port from the local management workstation
- 1 RJ45 Straight Ethernet cables for connecting the internal network with the remote management workstation to the TOE (via the appliance's management ports)

1.5 TOE Description

This section primarily addresses the physical and logical components of the TOE that are included in the evaluation.

1.5.1 Physical Scope

Figure 3 illustrates the physical scope and the physical boundary of the overall solution and ties together all of the components of the TOE and the constituents of the TOE Environment. The TOE is a hardware and firmware TOE; its components are the same as the product components as specified in Section 1.3. Previously undefined acronyms that appear in Figure 3 are:

- HTTPS – Hypertext Transfer Protocol over Secure Sockets Layer (SSL)
- TLS – Transport Layer Security

[17] NTP – Network Time Protocol
[18] DNS – Domain Name System
[19] JRE – Java Runtime Environment

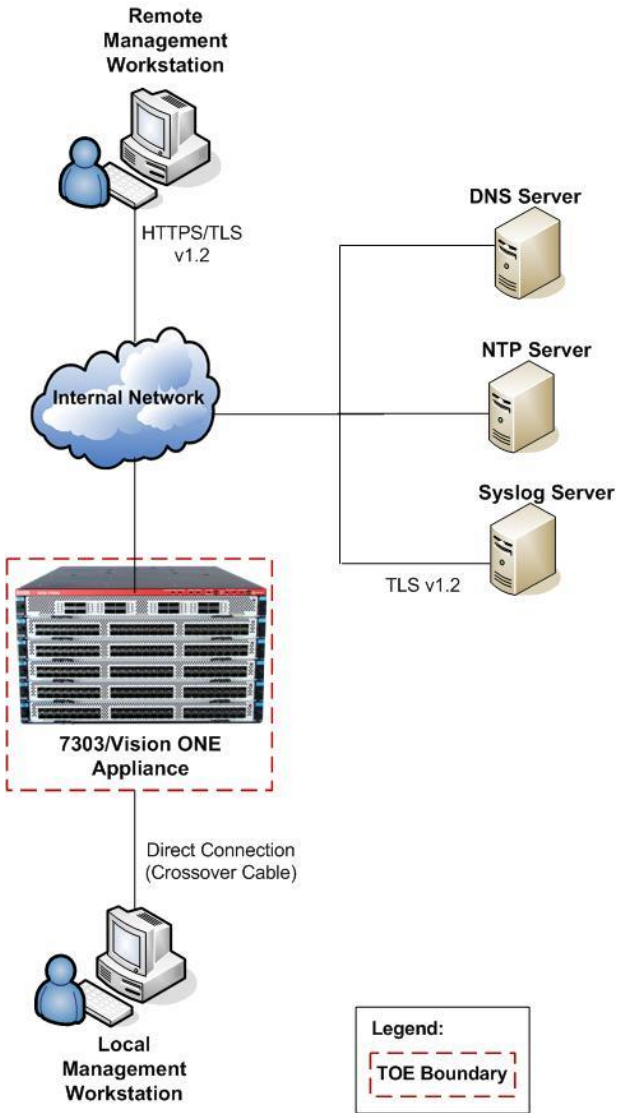

Figure 3 – Physical TOE Boundary

The TOE Boundary includes all the Ixia developed parts of the NTO 7303 and Vision ONE product. Any third-party source code or firmware on the NTO 7303 and Vision ONE that Ixia has modified is considered to be TOE firmware. The TOE Boundary specifically does not include any of the third-party software that the TOE relies upon as described in Section 1.4.1 of the ST.

1.5.1.1 TOE Hardware and Firmware

The TOE is a hardware and firmware TOE. For the evaluated configuration, the TOE firmware is pre-installed and runs on each of the following Ixia NTO TOE hardware models:

- Ixia Vision ONE Appliance
- Ixia 7303 Appliance

1.5.1.2 Guidance Documentation

Table 2 lists the TOE Guidance Documentation needed to install, configure, and maintain the TOE.

Table 2 – Guidance Documentation

Document Name	Description
Ixia Vision ONE Installation Guide, NVOS 4.5	Includes steps for the basic initialization and setup of the TOE components (including the 7303 line cards).
Ixia Vision 7300/7303 Installation Guide, NVOS 4.5	
Ixia Vision ONE User Guide, NVOS 4.5.0	Contains detailed steps for how to properly configure and maintain the TOE.
Ixia Net Tool Optimizer 7300/7303 User Guide, NVOS 4.5.0	
Ixia NTO 7303 and Vision ONE v4.5.0 Guidance Documentation Supplement	Contains information regarding the specific configuration for the TOE evaluated configuration.

1.5.2 Logical Scope

The logical boundary of the TOE will be broken down into the following security functions implemented by the TOE which are further described in sections 7 and 8 of this ST.

1.5.2.1 Security Audit

The TOE generates audit records for security relevant actions of the authorized administrators within the NTO Web Console and NTO Java Console. Audit records are sent simultaneously to the local circular log file buffer and the external syslog server. The audit records include the date and time of the events, type of events, subject identity, and outcome of the events. When the local log file buffer approaches its size limit, the earliest entries are overwritten. Local audit records can be viewed by authorized administrators and are protected from unauthorized modification or deletion.

1.5.2.2 Cryptographic Support

The Cryptographic Support of the TSF[20] function provides cryptographic functions to secure TLS v1.2 and HTTPS connections from external hosts connecting via the management interfaces (NTO Web Console and NTO Java Console) to the TOE. Cryptographic functions are also used to secure TLS v1.2 trusted channels between the TOE and the Syslog server.

The TOE supports AES[21] 128-bit and 256-bit CBC[22] mode for encryption and decryption. The TOE supports RSA 2048 for signature verification (with SHA[23]-256). The TOE supports SHA-1 for hashing local passwords and TLS, SHA-256 for upgrades and X.509 certificates, and HMAC[24] SHA-1 for TLS services.

[20] TSF – TOE Security Functionality
[21] AES – Advanced Encryption Standard
[22] CBC – Cipher Block Chaining mode
[23] SHA – Secure Hash Algorithm
[24] HMAC – Hashed Message Authentication Code

The TOE supports the generation of asymmetric cryptographic keys and utilizes RSA 2048 for key establishment. The TOE utilizes a SHA-256 Hash DRBG[25] and zeroizes cryptographic keys by overwriting with zeros.

1.5.2.3 Identification and Authentication

The TOE provides functionality that requires administrators and users to verify their claimed identity. The Identification and Authentication TSF ensures that only legitimate administrators can gain access to the configuration settings and management settings of the TOE. Besides loading the NTO Web Console and NTO Java Console login pages, the NTO Webstart launch page, and the login banner, administrators and users must log in with a valid user name and password before the TOE will permit access to TOE functionality.

The TOE utilizes X.509 certificates for TLS v1.2 communications. These certificates are validated by the TOE and used to support authentication for TLS v1.2. The TOE supports the generation of Certificate Request Messages.

The TOE implements requirements for password complexity and length, and the TOE provides obscured feedback to administrative users while authentication is in progress for the NTO Web Console and the NTO Java Console.

1.5.2.4 Security Management

The TOE provides a set of management interfaces for administrators to manage the security functions, configuration, and other features of the TOE components. The Security Management function specifies the administrator defined access for the management of the TOE components. The TOE management functions include:

- Administer the TOE locally and remotely
- Manually update the TOE and verify TOE updates
- Configure the cryptographic functionality
- Configure the login banner
- Configure session timeouts

1.5.2.5 Protection of the TSF

The TOE provides reliable timestamps for its own use by synchronizing with an NTP server. Digital signatures are used to verify all firmware updates that are applied to the TOE. The TOE runs a suite of self-tests at power-on and during normal operation to ensure the integrity of the TSF. The TOE hashes passwords and prevents access and reading of plaintext keys and passwords.

1.5.2.6 TOE Access

The TOE automatically logs out a user from the local and remote management interfaces after an administrator-specified amount of idle time. Users can also terminate their own session. The TOE displays an access banner prior to all administrative sessions.

1.5.2.7 Trusted Path/Channels

The TOE provides trusted paths and trusted channels using its cryptographic functions. The TOE secures administrative communications using TLS v1.2 over its management interfaces.

The TOE also provides trusted TLS v1.2 communications channels between the TOE and the Syslog server.

[25] DRBG – Deterministic Random Bit Generator

1.5.3 Product Physical/Logical Features and Functionality not included in the TOE

Features and Functionality that are not part of the evaluated configuration of the TOE are:

- Tcl[26] API[27]
- Web API access to the TOE
- Craft serial port (used for initial configuration only)
- SNMP[28] Services
- TACACS+[29] authentication
- RADIUS[30] authentication

1.5.4 Scope of Evaluation

The evaluation is limited in scope to the secure management features described in the *Collaborative Protection Profile for Network Devices v1.0, Feb 27, 2015* and detailed in Section 1.5.2.

[26] Tcl – Tool Command Language
[27] API – Application Programming Interface
[28] SNMP – Simple Network Management Protocol
[29] TACACS+ – Terminal Access Controller Access-Control System Plus
[30] RADIUS – Remote Authentication Dial-In User Service

2. Conformance Claims

This section provides the identification for any CC, PP, and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. Rationale for CC and PP conformance claims can be found in Section 9.1.

Table 3 – CC and PP Conformance

CC Identification and Conformance	Common Criteria for Information Technology Security Evaluation, Version 3.1, Release 4, September 2012; CC Part 2 extended; CC Part 3 conformant; PP claim to the collaborative Protection Profile for Network Devices v1.0, Feb 27, 2015 conformant.
PP Identification	Exact Conformance[31] to the collaborative Protection Profile for Network Devices, v1.0.

[31] Exact Conformance is a type of Strict Conformance such that the set of SFRs and the SPD/Objectives are exactly as presented within the accepted NDPP without changes.

3. Security Problem Definition

This section describes the security aspects of the environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It provides the statement of the TOE security environment, which identifies and explains all:

- Known and presumed threats countered by either the TOE or by the security environment
- Organizational security policies with which the TOE must comply
- Assumptions about the secure usage of the TOE, including physical, personnel, and connectivity aspects

A network device has a network infrastructure role it is designed to provide. In doing so, the network device communicates with other network devices and other network entities (an entity not defined as a network device) over the network. At the same time, it must provide a minimal set of common security functionality expected by all network devices. The security problem to be addressed by a compliant network device is defined as this set of common security functionality that addresses the threats that are common to network devices, as opposed to those that might be targeting the specific functionality of a specific type of network device. The set of common security functionality addresses communication with the network device, both authorized and unauthorized; the ability to perform valid or secure updates; the ability to audit device activity, the ability to securely store and utilize device and administrator credentials and data; and the ability to self-test critical device components for failures.

3.1 Threats to Security

This section identifies the threats to the IT[32] assets against which protection is required by the TOE or by the security environment. The threat agents are divided into two categories:

- Attackers who are not TOE users: These threat agents have public knowledge of how the TOE operates and are assumed to possess a low skill level, limited resources to alter TOE configuration settings or parameters, and no physical access to the TOE.
- TOE administrative users: These threat agents have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters, and physical access to the TOE. (TOE administrative users are, however, assumed to operate in a trusted manner.)

Both are assumed to have a low level of motivation. The IT assets requiring protection are the TSF and user data saved on the TOE. Removal, diminution, and mitigation of the threats are achieved through the objectives identified in Section 4. Section 3.1 below lists the applicable threats.

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

[32] IT – Information Technology

3.1.1 Communications with the Network Device

A network device communicates with other network devices and other network entities. The endpoints of this communication can be geographically and logically distant and may pass through a variety of other systems. The intermediate systems may be untrusted providing an opportunity for unauthorized communication with the network device or for authorized communication to be compromised. The security functionality of the network device must be able to protect any critical network traffic (administration traffic, authentication traffic, audit traffic, etc.). The communication with the network device falls into two categories: authorized communication and unauthorized communication.

Authorized communication includes network traffic allowable by policy destined to and originating from the network device as it was designed and intended. This includes critical network traffic, such as network device administration and communication with an authentication or audit logging server, which requires a secure channel to protect the communication. The security functionality of the network device includes the capability to ensure that only authorized communications are allowed and the capability to provide a secure channel for critical network traffic. Any other communication is considered unauthorized communication.

The primary threats to network device communications addressed in this cPP focus on an external, unauthorized entity attempting to access, modify, or otherwise disclose the critical network traffic. A poor choice of cryptographic algorithms or the use of non-standardized tunneling protocols along with weak administrator credentials, such as an easily guessable password or use of a default password, will allow a threat agent unauthorized access to the device. Weak or no cryptography provides little to no protection of the traffic allowing a threat agent to read, manipulate and/or control the critical data with little effort. Non-standardized tunneling protocols not only limit the interoperability of the device but lack the assurance and confidence standardization provides through peer review.

3.1.1.1 T.UNAUTHORIZED_ADMISTRATOR_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.

3.1.1.2 T.WEAK_CRYPTOGRAPHY

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

3.1.1.3 T.UNTRUSTED_COMMUNICATION_CHANNELS

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

3.1.1.4 T.WEAK_AUTHENTICATION_ENDPOINTS

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

3.1.2 Valid Updates

Updating network device software and firmware is necessary to ensure that the security functionality of the network device is maintained. The source and content of an update to be applied must be validated by cryptographic means; otherwise, an invalid source can write their own firmware or software updates that circumvent the security functionality of the network device. Methods of validating the source and content of a software or firmware update by cryptographic means typically involve cryptographic signature schemes where hashes of the updates are digitally signed.

Unpatched versions of software or firmware leave the network device susceptible to threat agents attempting to circumvent the security functionality using known vulnerabilities. Non-validated updates or updates validated using non-secure or weak cryptography leave the updated software or firmware vulnerable to threat agents attempting to modify the software or firmware to their advantage.

3.1.2.1 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.

3.1.3 Audited Activity

Auditing of network device activities is a valuable tool for administrators to monitor the status of the device. It provides the means for administrator accountability, security functionality activity reporting, reconstruction of events, and problem analysis. Processing performed in response to device activities may give indications of a failure or compromise of the security functionality. When indications of activity that impact the security functionality are not generated and monitored, it is possible for such activities to occur without administrator awareness. Further, if records are not generated and retained, reconstruction of the network and the ability to understand the extent of any compromise could be negatively affected. Additional concerns are the protection of the audit data that is recorded from alteration or unauthorized deletion. This could occur within the TOE or while the audit data is in transit to an external storage device.

Note this cPP requires that the network device generate the audit data and has the capability to send the audit data to a trusted network entity (e.g., a Syslog server).

3.1.3.1 T.UNDETECTED_ACTIVITY

Threat agents may attempt to access, change, 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.

3.1.4 Administrator and Device Credentials and Data

A network device contains data and credentials which must be securely stored and must appropriately restrict access to authorized entities. Examples include the device firmware, software, configuration authentication credentials for secure channels, and administrator credentials. Device and administrator keys, key material, and authentication credentials need to be protected from unauthorized disclosure and modification. Furthermore, the security functionality of the device needs to require default authentication credentials, such as administrator passwords, be changed.

Lack of secure storage and improper handling of credentials and data, such as unencrypted credentials inside configuration files or access to secure channel session keys, can allow an attacker to not only gain access to the network device, but also compromise the security of the network through seemingly authorized modifications to configuration or though man-in-the-middle attacks. These attacks allow an unauthorized entity to gain access and perform administrative functions using the Security Administrator's credentials and to intercept all traffic as an authorized endpoint. This results in difficulty in detection of security compromise and in reconstruction of the network, potentially allowing continued unauthorized access to administrator and device data.

3.1.4.1 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 include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.

3.1.4.2 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.

3.1.5 Device Failure

Security mechanisms of the network device generally build up from roots of trust to more complex sets of mechanisms. Failures could result in a compromise to the security functionality of the device. A network device self-testing its security critical components at both start-up and during run-time ensures the reliability of the device's security functionality.

3.1.5.1 T.SECURITY_FUNCTIONALITY_FAILURE

A component of the network device may fail during start-up or during operations causing a compromise or failure in the security functionality of the network device, leaving the device susceptible to attackers.

3.2 Assumptions

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

3.2.1 A.PHYSICAL_PROTECTION

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

[OE.PHYSICAL]

3.2.2 A.LIMITED_FUNCTIONALITY

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

[OE.NO_GENERAL_PURPOSE]

3.2.3 A.NO_THRU_TRAFFIC_PROTECTION

A standard or 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 and 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 for particular types of network devices (e.g., firewall).

[OE.NO_THRU_TRAFFIC_PROTECTION]

3.2.4 A.TRUSTED_ADMINISTRATOR

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

[OE.TRUSTED_ADMIN]

3.2.5 A.REGULAR_UPDATES

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

[OE.UPDATES]

3.2.6 A.ADMIN_CREDENTIALS_SECURE

The administrator's credentials (private key) used to access the network device are protected by the platform on which they reside.

[OE.ADMIN_CREDENTIALS_SECURE]

3.3 Organizational Security Policies

An OCP is a set of rules, practices, and procedures imposed by an organization to address its security needs. For the purposes of this cPP, a single policy is described in the section below.

3.3.1 P.ACCESS_BANNER

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

[FTA_TAB.1]

4. Security Objectives

4.1 Security Objectives for the Operational Environment

The following subsections describe objectives for the Operational Environment.

4.1.1 OE.PHYSICAL

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

4.1.2 OE.NO_GENERAL_PURPOSE

There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration, and support of the TOE.

4.1.3 OE.NO_THRU_TRAFFIC_PROTECTION

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

4.1.4 OE.TRUSTED_ADMIN

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

4.1.5 OE.UPDATES

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

4.1.6 OE.ADMIN_CREDENTIALS_SECURE

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

5. Extended Components

This section defines the extended SFRs and extended SARs met by the TOE. These requirements are presented following the conventions identified in Section 7. This section contains the definitions for the extended requirements that are used in the cPP, including those used in sections 6 and 7.

5.1 Extended TOE Security Functional Components

All of the extended requirements in this ST have been drawn from the ND cPP v1.0. The ND cPP v1.0 defines the following extended SFRs and since they are not redefined in this ST the ND cPP v1.0 should be consulted for more information in regard to those CC extensions. Table 4 identifies all extended SFRs implemented by the TOE.

Table 4 – Extended TOE Security Functional Requirements

Name	Description
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_TLSC_EXT.2	TLS Client Protocol with Authentication
FCS_TLSS_EXT.1	TLS Server Protocol
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_X509_EXT.1	Certificate Validation
FIA_X509_EXT.2	Certificate Authentication
FIA_X509_EXT.3	Certificate Requests
FPT_SKP_EXT.1	Protection of TSF data (for reading of all symmetric keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_TST_EXT.1	TSF Testing
FPT_TUD_EXT.1	Trusted Update
FTA_SSL_EXT.1	TSF-initiated session locking

5.2 Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6. Security Assurance Requirements

This cPP identifies the SARs to frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

This section lists the set of SARs from CC Part 3 that are required in evaluations against this cPP. Individual Evaluation Activities to be performed are specified in the *Supporting Document, Mandatory Technical Document, Evaluation Activities for Network Device cPP* [SD].

The general model for evaluation of TOEs against STs written to conform to this cPP is as follows. After the ST has been approved for evaluation, the ITSEF[33] will obtain the TOE, supporting environmental IT (if required), and the guidance documentation for the TOE. The ITSEF is expected to perform actions mandated by the Common Evaluation Methodology (CEM) for the ASE and ALC SARs. The ITSEF also performs the Evaluation Activities contained within the SD, which are intended to be an interpretation of the other CEM assurance requirements as they apply to the specific technology instantiated in the TOE. The Evaluation Activities that are captured in the SD also provide clarification as to what the developer needs to provide to demonstrate the TOE is compliant with the cPP.

The TOE security assurance requirements are identified in Table 5.

Table 5 – Security Assurance Requirements

Assurance Requirements	
Security Target (ASE)	Conformance claims (ASE_CCL.1)
	Extended components definition (ASE_ECD.1)
	ST introduction (ASE_INT.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Stated security requirements (ASE_REQ.1)
	Security Problem Definition (ASE_SPD.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labeling of the TOE (ALC_CMC.1)
	TOE CM[34] coverage (ALC_CMS.1)
Tests (ATE)	Independent testing – sample (ATE_IND.1)
Vulnerability assessment (AVA)	Vulnerability survey (AVA_VAN.1)

[33] ITSEF – Information Technology Security Entrepreneurs Forum
[34] CM – Configuration Management

7. Security Functional Requirements

The individual security functional requirements are specified in the sections below.

7.1 Conventions

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

- Assignment: Indicated with italicized text surrounded by brackets (e.g., [*assignment*]);
- Refinement: Indicated with bold text and strikethroughs (e.g., "**refinement**" or "~~refinement~~");
- Selection: Indicated with underlined text surrounded by brackets (e.g., [selection]);
- Assignment within a Selection: Indicated with italicized and underlined text surrounded by brackets (e.g., [*assignment within a selection*]);
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/".
- Extended SFRs are identified by having a label 'EXT' at the end of the SFR name.

Only the requirements and residual assignments and operations from the ND cPP v1.0 are completed in section 7. The ND cPP v1.0 includes a number of refinements and completed operations. Refer to the ND cPP v1.0 for more details.

7.2 Security Functional Requirements

This section specifies the SFRs for the TOE. This section organizes the SFRs by CC class. Table 6 identifies all SFRs implemented by the TOE and indicates the ST operations performed on each requirement.

Table 6 – TOE Security Functional Requirements

Name	Description	S	A	R	I
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(1)	Cryptographic Operation (AES Data Encryption/Decryption)	✓			
FCS_COP.1(2)	Cryptographic Operation (Signature Generation and Verification)	✓			
FCS_COP.1(3)	Cryptographic Operation (Hash Algorithm)	✓			
FCS_COP.1(4)	Cryptographic Operation (Keyed Hash Algorithm)	✓	✓		
FCS_HTTPS_EXT.1	HTTPS Protocol	✓			

Name	Description	S	A	R	I
FCS_RBG_EXT.1	Random Bit Generation	✓	✓		
FCS_TLSC_EXT.2	TLS Client Protocol with Authentication	✓			
FCS_TLSS_EXT.1	TLS Server Protocol	✓			
FIA_PMG_EXT.1	Password Management	✓	✓		
FIA_UAU.7	Protected Authentication Feedback				
FIA_UAU_EXT.2	Password-based Authentication Mechanism	✓			
FIA_UIA_EXT.1	User Identification and Authentication	✓	✓		
FIA_X509_EXT.1	X.509 Certificate Validation	✓			
FIA_X509_EXT.2	X.509 Certificate Authentication	✓			
FIA_X509_EXT.3	X.509 Certificate Requests	✓			
FMT_MOF.1(1)/ TrustedUpdate	Management of security functions behavior				
FMT_MTD.1	Management of TSF data				
FMT_MTD.1/AdminAct	Management of TSF data				
FMT_SMF.1	Specification of management functions	✓			
FMT_SMR.2	Restrictions on Security Roles				
FPT_APW_EXT.1	Protection of Administrator Passwords				
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)				
FPT_STM.1	Reliable Time Stamps				
FPT_TST_EXT.1	TSF testing	✓	✓		
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 banners				
FTP_ITC.1	Inter-TSF Trust Channel	✓	✓		
FTP_TRP.1	Trusted Path	✓			

Note: S=Selection; A=Assignment; R=Refinement; I=Iteration

7.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit Data Generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

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).
   - Security related configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
   - Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
   - Resetting passwords (name of related user account shall be logged).
   - ~~Starting and stopping services (if applicable)~~
   - [no other actions]
d. Specifically defined auditable events listed in **Table 7**.

Table 7 – 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(1)	None.	None.
FCS_COP.1(2)	None.	None.
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure
FCS_TLSC_EXT.2	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_RBG_EXT.1	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP[35] address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1	Unsuccessful attempt to validate a certificate	Reason for failure

[35] IP – Internet Protocol

Requirement	Auditable Events	Additional Audit Record Contents
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1(1)/TrustedUpdate	Any attempt to initiate a manual update	None.
FMT_MTD.1	All management activities of TSF data.	None.
FMT_MTD.1/AdminAct	Modification, deletion, generation/import of cryptographic keys	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_STM.1	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_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	No additional information.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempts.
FTP_TRP.1	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	Identification of the claimed user identity.

FAU_GEN.1.2

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

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

b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of **Table 7.**

FAU_GEN.2 User identity association

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

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.

FAU_STG_EXT.1 Protected Audit Event Storage

Hierarchical to: **No other components.**

Dependencies: **FAU_GEN.1 Audit data generation**
FTP_ITC.1 Inter-TSF trusted channel

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.

FAU_STG_EXT.1.3

The TSF shall [*delete the oldest 1 MB*[36] *of local audit data*] when the local storage space for audit data is full.

[36] MB – Megabyte

7.2.2 Class FCS: Cryptographic Support

FCS_CKM.1 Cryptographic Key Generation
Hierarchical to: No other components.
Dependencies: FCS_COP.1 Cryptographic operation
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1

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

[

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS[37] PUB[38] 186-4, "Digital Signature Standard (DSS)", Appendix B.3;

].

FCS_CKM.2 Cryptographic Key Establishment
Hierarchical to: No other components.
Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

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 meets the following: NIST[39] Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography";

].

FCS_CKM.4 Cryptographic key destruction
Hierarchical to: No other components.
Dependencies: FCS_CKM.1 Cryptographic key generation

FCS_CKM.4.1

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

[

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

] that meets the following: *No Standard*.

[37] FIPS – Federal Information Processing Standard
[38] PUB – Publication
[39] NIST – National Institute of Standards and Technology

FCS_COP.1(1) Cryptographic Operation (AES Data Encryption/Decryption)
Hierarchical to: No other components.
Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(1)

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [CBC] mode and cryptographic key sizes [128 bits, 256 bits] that meet the following: AES as specified in ISO[40] 18033-3, [CBC as specified in ISO 10116].

FCS_COP.1(2) Cryptographic Operation (Signature Generation and Verification)
Hierarchical to: No other components.
Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(2)

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*]]

that meet the following:

[

- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS[41] #1 v2.1 Signature Schemes RSASSA[42]-PSS[43] and/or RSASSA-PKCS1v1_5; ISO/IEC[44] 9796-2, Digital signature scheme 2 or Digital Signature scheme 3

].

FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)
Hierarchical to: No other components.
Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(3)

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

FCS_COP.1(4) Cryptographic operation (Keyed Hash Algorithm)
Hierarchical to: No other components.
Dependencies: FCS_CKM.1 Cryptographic key generation
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1(4)

[40] ISO – International Organization for Standardization
[41] PKCS – Public Key Cryptography Standard
[42] RSASSA – RSA Signature Scheme with Appendix
[43] PSS – Probabilistic Signature Scheme
[44] IEC – International Electrotechnical Commission

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [HMAC-SHA-1, HMAC-SHA-256] and cryptographic key sizes [*160 bits, 256 bits*] and message digest sizes [160, 256] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC[45] Algorithm 2".

FCS_HTTPS_EXT.1 HTTPS Protocol
Hierarchical to: No other components.
Dependencies: FCS_TLS_EXT.1

FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC[46] 2818.

FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3
The TSF shall establish the connection only if [the peer initiates handshake].

FCS_TLSC_EXT.2 TLS Client Protocol with Authentication
Hierarchical to: FCS_TLSC_EXT.1 TLS Client Protocol
Dependencies: FCS_COP.1(1) Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1(2) Cryptographic operation (Signature Verification)
FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_TLSC_EXT.2.1
The TSF shall implement [TLS 1.2 (RFC 5246)] supporting 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 RFC 5246

].

FCS_TLSC_EXT.2.2
The TSF shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.2.3
The TSF shall only establish a trusted channel if the peer certificate is valid.

FCS_TLSC_EXT.2.4
The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [none] and no other curves.

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

FCS_TLSS_EXT.1 TLS Server Protocol
Hierarchical to: No other components
Dependencies: FCS_CKM.1 Cryptographic Key Generation
FCS_COP.1(1) Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1(2) Cryptographic operation (Signature Verification)

[45] MAC – Message Authentication Code
[46] RFC – Request For Comment

FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_TLSS_EXT.1.1

The TSF shall implement [TLS 1.2 (RFC 5246)] supporting 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

  ].

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 RSA key establishment with key size [2048 bits]].

FCS_RBG_EXT.1 Random Bit Generation
Hierarchical to: No other components.
Dependencies: No dependencies.

FCS_RBG_EXT.1.1

The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [Hash_DRBG].

FCS_RBG_EXT.1.2

The deterministic RBG[47] shall be seeded by at least one entropy source that accumulates entropy from [[*4*] software-based noise sources, [*1*] hardware-based noise source] with 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.

[47] RBG – Random Bit Generator

7.2.3 Class FIA: Identification and Authentication

FIA_PMG_EXT.1 Password Management
Hierarchical to: No other components.
Dependencies: No dependencies.
FIA_PMG_EXT.1.1

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“`”, ”~”, ”_”, ”+”, ”-“, ”=”, ”{“, ”}”, “|”, “\”, “:”, “”””, ”;”, ”’”, “<”, “>”, “?”, “,”, “.”, “/”, “[“, ”]”]
b) Minimum password length shall be settable by the Security Administrator, and shall support passwords of 15 characters or greater.

FIA_UIA_EXT.1 User identification and authentication
Hierarchical to: No other components.
Dependencies: FTA_TAB.1 Default TOE Access Banners.
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;
- [*Load the NTO Webstart launch page; Load the NTO Java Console login page, Load the NTO Web Console login page*].

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.

FIA_UAU_EXT.2 Password-based Authentication Mechanism
Hierarchical to: No other components
Dependencies: No other components
FIA_UAU_EXT.2.1

The TSF shall provide a local password-based authentication mechanism, [none] to perform administrative user authentication.

FIA_UAU.7 Protected authentication feedback
Hierarchical to: No other components.
Dependencies: FIA_UAU.1 Timing of authentication
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.

FIA_X509_EXT.1 X.509 Certificate Validation
Hierarchical to: No other components
Dependencies: No other components
FIA_X509_EXT.1.1

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

- RFC 5280 certificate validation and certificate path validation.

- The certificate path must terminate with a trusted CA[48] certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [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[49] 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
  - Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
  - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
  - OCSP[50] 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

The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 X.509 Certificate Authentication
Hierarchical to: No other components
Dependencies: No other components

FIA_X509_EXT.2.1

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

FIA_X509_EXT.2.2

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

FIA_X509_EXT.3 X.509 Certificate Requests
Hierarchical to: No other components
Dependencies: No other components

FIA_X509_EXT.3.1

The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key 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.

[48] CA – Certificate Authority
[49] OID – Object Identifier
[50] OCSP – Online Certificate Status Protocol

7.2.4 Class FMT: Security Management

FMT_MOF.1(1)/TrustedUpdate Management of security functions behaviour

Hierarchical to: **No other components.**

Dependencies: **FMT_SMR.1 Security roles**
FMT_SMF.1 Specification of Management Functions

FMT_MOF.1.1(1)/TrustedUpdate

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

FMT_MTD.1 Management of TSF data

Hierarchical to: **No other components.**

Dependencies: **FMT_SMF.1 Specification of management functions**
FMT_SMR.1 Security roles

FMT_MTD.1.1

The TSF shall restrict the ability to manage the TSF data to Security Administrators.

FMT_MTD.1/AdminAct Management of TSF data

Hierarchical to: **No other components.**

Dependencies: **FMT_SMF.1 Specification of management functions**
FMT_SMR.1 Security roles

FMT_MTD.1.1/AdminAct

The TSF shall restrict the ability to modify, delete, generate/import the cryptographic keys to Security Administrators.

FMT_SMF.1 Specification of Management Functions

Hierarchical to: **No other components.**

Dependencies: **FIA_UIA_EXT.1 User Identification and Authentication**
FCS_COP.1(2) Cryptographic operation (for cryptographic signature)
FPT_TUD_EXT.1 Trusted Update

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 cryptographic functionality]

FMT_SMR.2 Restrictions on security roles

Hierarchical to: **No other components.**

Dependencies: **FIA_UID.1 Timing of identification**

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.

7.2.5 Class FPT: Protection of the TSF

FPT_APW_EXT.1 Protection of Administrator Passwords
Hierarchical to: No other components
Dependencies: No dependencies.
FPT_APW_EXT.1.1
The TSF shall store passwords in non-plaintext form.
FPT_APW_EXT.1.2
The TSF shall prevent the reading of plaintext passwords.

FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric keys)
Hierarchical to: No other components
Dependencies: No dependencies.
FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

FPT_STM.1 Reliable time stamps
Hierarchical to: No other components.
Dependencies: No dependencies.
FPT_STM.1.1
The TSF shall be able to provide reliable time stamps.

FPT_TST_EXT.1 TSF Testing
Hierarchical to: No other components
Dependencies: No other components
FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [during initial start-up (on power on), at the conditions [*execution of DRBG or NDRNG[51] function, generation of asymmetric RSA keypair*]] to demonstrate the correct operation of the TSF:
[
- *Firmware Integrity Test*
- *Firmware load test with 2048-bit RSA private key*
- *AES ECB KAT[52]*
- *HMAC KAT with SHA-1, SHA-256*
- *SHA KAT with SHA-1, SHA-256*
- *NIST SP[53]800-90A Hash DRBG KAT*
- *RSA sign/verify KAT*
- *Continuous RNG test for DRBG*
- *Continuous RNG test for NDRNG*
- *Pairwise Consistency Test (PCT) for RSA keypairs*
- *DRBG Health Checks: instantiate, uninstantiate, generate, and reseed*

].

[51] NDRNG – Non-Deterministic Random Number Generator
[52] KAT – Known Answer Test
[53] SP – Special Publication

FPT_TUD_EXT.1 Trusted Update

Hierarchical to: **No other components**

Dependencies: **FCS_COP.1(1) Cryptographic operation (for cryptographic signature), or FCS_COP.1(3) Cryptographic operation (for cryptographic hashing)**

FPT_TUD_EXT.1.1

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

FPT_TUD_EXT.1.2

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

FPT_TUD_EXT.1.3

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

7.2.6 Class FTA: TOE Access

FTA_SSL_EXT.1 TSF-initiated session locking
Hierarchical to: No other components.
Dependencies: No dependencies.
FTA_SSL_EXT.1.1

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

FTA_SSL.3 TSF-initiated termination
Hierarchical to: No other components.
Dependencies: No dependencies.
FTA_SSL.3.1

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

FTA_SSL.4 User-initiated termination
Hierarchical to: No other components.
Dependencies: No dependencies.
FTA_SSL.4.1

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

FTA_TAB.1 Default TOE access banners
Hierarchical to: No other components.
Dependencies: No dependencies.
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.

7.2.7 Class FTP: Trusted Path/Channels

FTP_ITC.1 Inter-TSF trusted channel
Hierarchical to: No other components.
Dependencies: No other components.

FTP_ITC.1.1

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

FTP_ITC.1.2

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

FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [*sending audit data*].

FTP_TRP.1 Trusted path
Hierarchical to: No other components.
Dependencies: No other components.

FTP_TRP.1.1

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

The TSF shall permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3

The TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

8. TOE Summary Specification

This section presents information to detail how the TOE meets the functional requirements described in previous sections of this ST.

8.1 TOE Security Functionality

Each of the security requirements and the associated descriptions correspond to the security functions. Hence, each function is described by how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions satisfy the necessary requirements.

Table 8 – Mapping of TOE Security Functionality to Security Functional Requirements

TOE Security Function	SFR ID[54]	Description
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected audit trail storage
Cryptographic Support	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic key Destruction
	FCS_COP.1(1)	Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1(2)	Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1(3)	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1(4)	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_TLSC_EXT.2	TLS Client Protocol with authentication
	FCS_TLSS_EXT.1	TLS Server Protocol
Identification and Authentication	FIA_PMG_EXT.1	Password Management
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UIA_EXT.1	User Identification and Authentication

[54] ID – Identifier

TOE Security Function	SFR ID[54]	Description
	FIA_X509_EXT.1	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
Security Management	FMT_MOF.1(1)/TrustedUpdate	Management of security functions behavior
	FMT_MTD.1	Management of TSF data
	FMT_MTD.1/AdminAct	Management of TSF data
	FMT_SMF.1	Specification of management functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF testing
	FPT_TUD_EXT.1	Trusted Update
TOE Access	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 banners
Trusted path/channels	FTP_ITC.1	Inter-TSF Trust Channel
	FTP_TRP.1	Trusted Path

8.1.1 Security Audit

The Security Audit function provides the TOE with the functionality of generating audit records. As administrators manage and configure the TOE, their activities are tracked and recorded as audit records and are stored in the local logfile buffer. The resulting audit records can be examined to determine which security relevant activities took place and who (i.e., which user) is responsible for those activities. The TOE generates audit records for all the required events specified above in Table 8, the start-up and shut-down of the audit functions, and administrative actions. The administrative actions include administrative logins and logouts, security related configuration changes, the generation, import of, or deleting of cryptographic keys, and resetting passwords.

The audit records include the following fields:

- Date and Time Generated
- Severity Level
- Tag
- Message

- Source IP Address (external Syslog server only)
- Source Name (external Syslog server only)
- Date and Time Received (external Syslog server only)
- Origin (external Syslog server only)
- Facility (external Syslog server only)

The "Tag" field is a sequential Tag ID that resets to 0 on power up. When the "Tag" ID is set from 0-2, this indicates the startup of the audit function and that the system is ready to begin logging. The system starts up with an audit record indicating the ATIP Syslog initialization (0), an audit record indicating the system configuration is read (1), and then an audit record indicating that the server is ready (2) is displayed.

The shutdown of the audit function is implied by the shutdown of the system. Logs are displayed for the following types of TOE shutdown:

- Menu "File-Restart"
- Menu "File-Power Down"
- Install Software (upgrade)
- Revert software
- Enable Console encryption: (RMI[55] TLS)

The "Message" field identifies the user (for user-initiated events), subject identity, event type, configuration changes, and outcome (success/fail, alarm set/clear, state change, etc).

The TOE stores 2 MB of audit records in the local log file. As the 2 MB limit is reached, it will delete the oldest 1MB. Only authorized administrators can access the Syslog viewer to view the local audit records and the audit records cannot be deleted or modified via the Syslog viewer. When audit events are generated, they are simultaneously sent to the external Syslog server and the local logfile buffer. The persistent logfile cannot be deleted via the management interfaces and users do not have access to the filesystem to delete or modify the logfile. The TOE securely transfers audit data to an external Syslog server via TLS v1.2 encrypted communications.

TOE Security Functional Requirements Satisfied: FAU_GEN.1, FAU_GEN.2, FAU_STG_EXT.1.

8.1.2 Cryptographic Support

The TOE provides cryptographic algorithms for TLS and HTTPS connections, firmware upgrade signature verification, and local password hashing using the Ixia NTO Java Crypto Library 1.0.1-1. This includes the following CAVP[56] validated cryptographic algorithms:

- AES (Certificate #4747)
- RSA (Certificate #2592)
- SHA (Certificate #3891)
- DRBG (Certificate #1629)
- HMAC (Certificate #3162)
- Component Test (Certificate #1383)

[55] RMI – Remote Method Invocation
[56] CAVP – Cryptographic Algorithm Validation Program

The TOE provides a 256-bit SP800-90A Hash DRBG that is seeded with a minimum of 256 bits of entropy from one hardware and four software noise sources. Hardware entropy is gathered from the RDRAND instruction on the Intel Core i7-3555LE CPU[57] chip located within the NTO 7303 and Vision ONE appliances. The RDRAND instruction relies on the underlying built-in DRNG[58] of the Intel Core i7-3555LE CPU chip which uses thermal noise within the silicon to output a random stream of bits. Software entropy is collected from the following four software noise sources (Linux kernel events) of the NTO 7303 and Vision ONE appliances:

- Keyboard activity: Contains keyboard press and release codes.
- Touchscreen/mouse activity: Contains an event type (pressing or releasing), a code for which mouse button was pressed, and a value for the type of movement.
- Completion of disk I/O[59] operations.
- Interrupt events: Contains the Interrupt Request Channel (IRQ) number.

RSA 2048 with SHA-256 digital signatures are used for firmware upgrades, and RSA 2048 with SHA-256 are used for the TLS certificates. TLS connections (client-side and server-side) rely on AES 128-bit and 256-bit CBC mode for encryption and decryption services. The TOE acts as both a sender (Syslog TLS client) and recipient (TLS server) for RSA-based key establishment. RSA decryption errors throw generic exceptions which are logged but do not reveal the actual error.

The HMAC-SHA-1 cryptographic algorithm uses the SHA-1 hash function with a cryptographic key size of 160 bits and 160-bit message digest size in accordance with the ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”. The HMAC-SHA-256 cryptographic algorithm uses the SHA-256 hash function with a cryptographic key size of 256 bits and 256-bit message digest size in accordance with the ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.

The TOE zeroizes asymmetric and symmetric keys stored in various memory types. The persistently stored plaintext keys in non-volatile flash memory are as follows:

- TLS server private and public RSA keys (used for NTO Java Console and NTO Web Console).
- Syslog server trusted root CA public RSA key
- Syslog client TLS private and public RSA keys
- Ixia public key (used for firmware upgrade and licensing – cannot be zeroized)

The ephemeral plaintext keys stored within volatile memory that are zeroized are as follows:

- TLS session keys and MAC keys

The TOE supports the generation of 2048-bit certificate keypairs. Default TLS certificate key pairs are included within certificate keystores in non-volatile flash memory within the NTO file system. When a TLS certificate keypair is generated, imported, or deleted, the TOE zeroizes the non-volatile flash memory via a shell script (called by Java). The shell script uses the Linux ‘`dd`’ command-line utility to zeroize the non-volatile memory via a single direct overwrite (consisting of zeroes) followed by a read-verify.

The read-verify process of the shell script involves removing the zeros from the non-volatile memory and then writing the non-zero content to a temporary file. The shell script then checks that the temporary file is empty, which provides verification that the non-volatile memory was zeroized. If the read-verification of the overwritten data fails, the process is repeated again. If the process fails three times, the NTO system shuts down.

[57] CPU – Central Processing Unit
[58] DRNG – Digital Random Number Generator
[59] I/O – Input/Output

TLS session keys and MAC keys are zeroized on disconnection of the TLS sessions. The TOE uses a Java function to zeroize volatile memory via a single direct overwrite (consisting of zeroes) of the entire key array followed by a read-verify. To verify that the key array has been zeroized, the read-verification process compares each entry of the key array to "0". If the read-verification of the overwritten data fails, the process is repeated again. If the process fails three times, the NTO system shuts down.

8.1.2.1 TLS Client Protocol with Authentication

The TOE enforces client-side TLS v1.2 for connections to the external Syslog server. The client-side TLS connections 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 RFC 5246

The TOE supports 2-way certificate authentication with X.509v3 certificates for Syslog communication. Only DNS names are supported as acceptable reference identifiers. For Syslog certificate verification, the TOE compares the reference identifiers to the identifier in the presented Syslog server's TLS certificate. Certificate pinning is not supported. By default, the TOE does not support ciphers with elliptic curves in the evaluated configuration and therefore does not present the Supported Elliptic Curves extension for TLS connections.

8.1.2.2 TLS Server Protocol

The TOE supports server-side TLS v1.2 for secure connections from the remote management workstation to the NTO Web Console (HTTPS) and NTO Java Console (RMI over TLS) management interfaces. The NTO Web Console and NTO Java Console includes a link to launch the NTO ATIP GUI, which establishes an independent TLS v1.2 session within the current NTO Web Console or NTO Java Console user session. The ATIP GUI can only be launched via the launch link and cannot be accessed directly via URL. Both the NTO Web Console/NTO Java Console and ATIP GUI TLS sessions run in parallel.

The server-side TLS v1.2 connections 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

The TOE only accepts TLS v1.2 requests and denies any other SSL or TLS connection requests. Authorized administrators can import TLS certificates that are signed by an external CA or trust the default TLS certificate provided by the TOE. Non-RSA certificates are not supported.

The TOE acts as a server during remote administration TLS connections and does not enforce mutual certificate-based authentication. The TOE only establishes connections with clients (remote workstations) over HTTPS (via the NTO Web Console) and TLS (via the NTO Java Console) if the client/peer initiates the handshake. The client is then successfully authenticated when the TOE verifies a submitted username and password against stored values (as described in section 8.1.3).

TOE Security Functional Requirements Satisfied: FCS_CKM.1, FCS_CKM.2, FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_HTTPS_EXT.1, FCS_TLSC_EXT.2, FCS_TLSS_EXT.1, FCS_RBG_EXT.1.

8.1.3 Identification and Authentication

Authorized administrators can configure passwords to be at least a minimum password length of fifteen (15) characters. Valid passwords can be composed of any combination of upper and lower-case letters, numbers, and special characters. The special characters can be any of the following printable characters: ! @ # $ % ^ & * ( ) ` ~ _ + - = { } | \ : ” ” < > ? , . / [ ]

The TOE obscures password feedback during entry with bullets when users log in to the NTO Web Console and NTO Java Console management interfaces.

The management interfaces can be accessed locally via a direct connection (using a crossover cable) to the NTO appliance management port, or remotely via secure TLS v1.2 connections. The NTO Webstart launch page (accessed by typing the NTO IP address in a supported browser) includes an NTO Web Console launch button, NTO Java Console launch button, links to startup and user NTO guides, limited NTO system status information, and installation links. The NTO Web Console can also be accessed directly via a URL address (https://<IP_Addr>:8000/console/NtoUI/index.html). Besides loading the NTO Web Console and NTO Java Console login pages, the NTO Webstart launch page, and the login banner, no TOE functionality is available before identification and authentication occurs.

The NTO Java Console is a Java Swing-based GUI that uses RMI over TLS v1.2 for communication between the console and NTO server. The NTO Web Console is a JavaScript-based HTTPS GUI that uses REST[60]ful API methods to manage the NTO server.

The TOE supports local authentication for each management interface. Authentication is successful when the NTO server verifies the submitted username and password against stored values. Passwords are hashed using SHA-1.

The TOE uses X.509 certificates to authenticate to a Syslog server using TLS v1.2. To do this, a mutually trusted CA must be used. The CA certificate must be installed on both the Syslog server and the TOE. The TOE supports the generation of 2048-bit TLS certificate keypairs. After a TLS certificate keypair is generated by the TOE, the private key is used to generate a certificate signing request that is then exported and signed by the CA using OpenSSL or Microsoft AD Certificate Services. Certificates are generated for the Syslog server and for the TOE (acting as the client). Once the Syslog server and TOE client certificates have been generated, the trusted root certificate of the CA signing the client certificate must be added to the Syslog server’s trust store, and both the trusted root certificate of the CA signing the Syslog server certificate and the TOE client certificate must be imported into the TOE. The Syslog server supported by the TOE is syslog-ng.

The TOE supports the generation of Certificate Request Messages as specified by RFC 2986 and validates the chain of certificates from the root CA upon receiving the CA Certificate Response. Certificate Request Messages provided by the TOE include the following information in the request:

- Public key
- Common Name (CN)
- Organization (O)
- Organizational Unit (OU)
- Country (C)

[60] REST – Representational State Transfer

The TOE ensures that the X.509 certificates adhere to RFC 5280 Section 6.3 (certificate validation and certificate path validation) and that the certificate path terminates with a trusted CA certificate. The TOE validates a certificate path and treats a certificate as a CA certificate when certificates include the basicConstraints extensions and that the CA flag is set to "TRUE" for all CA certificates. The TOE validates the revocation status of Syslog TLS certificates using a CRL that is retrieved (downloaded) from the CA when the TLS connection is initiated. When a Syslog (TLS v1.2) connection cannot be established during the validity check of a certificate used in establishing a trusted channel, a syslog event is sent to the local logfile buffer and the connection is aborted. The TOE validates the extendedKeyUsage field by ensuring that server certificates presented for TLS have the Server Authentication purpose in the extended key usage field. The TOE also validates the revocation status of imported Syslog TLS client and server certificates using the CRL that is retrieved from the CA when a user attempts to save the certificate. If the certificate is revoked during the validity check of the imported certificate, the certificate is not saved and a syslog event is sent to the local logfile buffer.

TOE Security Functional Requirements Satisfied: FIA_PMG_EXT.1, FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7, FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3.

8.1.4 Security Management

Security management specifies how the TOE manages several aspects of the TSF including TSF data and security functions. TSF data includes configuration data of the TSF and audit data. The TOE provides authorized administrators that utilize the NTO Web Console and NTO Java Console with the ability to easily manage the security functions and TSF data of the TOE. Besides loading with the NTO Webstart launch page, the NTO Web Console and NTO Java Console login dialogs, and the login banner, no TOE functionality is available before identification and authentication occurs.

The TOE maintains the Administrator role. The Administrator role has "Security Administrator" privileges discussed throughout the ST and is responsible for all Security Administrator restricted management functions. The Administrator role can manage the TOE locally and remotely and provides Security Administrator management functions including configuration of the TOE access banner and session inactivity timeouts, initiation of manual TOE updates (and verification of TOE updates), and generation, importation, or deletion of cryptographic keys.

TOE Security Functional Requirements Satisfied: FMT_MOF.1(1)/TrustedUpdate, FMT_MTD.1, FMT_MTD.1/AdminAct, FMT_SMF.1, FMT_SMR.2.

8.1.5 Protection of the TSF

Local passwords are hashed with SHA-1 and then stored within the flash memory on the NTO server filesystem. All persistently stored private keys listed in section 8.1.2 are stored in plaintext keystores within the non-volatile flash memory of the TOE filesystem. Filesystem keystores are inaccessible to users and cannot be viewed via any management interface. The plaintext keys within the volatile memory (see section 8.1.2) cannot be viewed via the management interfaces and are zeroized after use.

The TOE provides reliable time stamps for the Security Audit functionality, to track the inactivity of administrative sessions, and for cryptographic functions. An external NTP server can be used to synchronize the NTO clock and provide reliable time stamps for syslog messages.

The TOE relies on an RSA 2048-bit digital signature mechanism for performing software upgrades. Upgrade files are encrypted using an RSA 2048-bit private key. A SHA-256 hash value is calculated for the encrypted upgrade file and the hash is then encrypted using the same RSA 2048-bit private key before being packaged with the encrypted upgrade file as a digital signature.

Digitally signed upgrade tar files are acquired by contacting Ixia Technical Support and only authorized administrators can perform manual software upgrades via the NTO Java Console. When the administrator performs the software upgrade, the TOE performs a Firmware Load test. The Firmware Load test first verifies that the packaged SHA-256 hash value exists and decrypts the SHA-256 hash value using the RSA 2048-bit public key that is hardcoded in a keystore within the TOE filesystem. The test then calculates a SHA-256 hash value for the encrypted upgrade file and compares the result with the packaged SHA-256 hash value. If the hash values do not match, the digital signature is not verified and the software upgrade cannot proceed. If the packaged SHA-256 hash value matches the calculated SHA-256 hash value, the digital signature has been verified and the test decrypts the upgrade file using the hardcoded RSA 2048-bit public key. The TOE software then extracts the upgrade components from the tar file and performs the software upgrade. After the software is installed, the TOE is automatically restarted and requires the administrator to log in again for the upgrade process to be complete. If the newly installed software fails for any reason, the TOE software reverts to the previous version without loss of functionality. The TOE version can be verified via the NTO Web Console and NTO Java Console.

The TOE performs a suite of FIPS power-up and conditional self-tests to verify its correct operation. If any of the self-tests fail, the TOE enters into a critical error state and the appliance must be rebooted by an administrator to run the tests again. The following is a description of each of the start-up tests:

- Firmware Integrity Test: The module verifies the integrity of the entire TOE firmware/software image using a 32-bit CRC[61] during the first phase of the boot process. If the CRC is verified (the newly-computed CRC matches the stored CRC), the test is passed and the boot process proceeds. If the test fails, the module enters a critical error state and halts the boot process. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that the software/firmware integrity test uses an EDC [62] or Approved authentication technique to test the integrity of software/firmware components when the TOE is powered up.
- AES ECB encrypt/decrypt KAT: The AES ECB KAT takes a known 256-bit key and plaintext value, which is encrypted and compared to the expected ciphertext value to test that the encryption operation is working correctly. If the values differ, the test is failed. The AES ECB KAT then reverses this process by taking the ciphertext value and key, performing decryption, and comparing the result to the known plaintext value to test that the decrypt operation is working correctly. If the values differ, the test is failed. If they are the same, the test is passed. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that a cryptographic algorithm test using a KAT shall be conducted for all cryptographic functions of each Approved cryptographic algorithm implemented by the TOE.
- HMAC KAT with SHA-1, SHA-256: The HMAC implementation creates a MAC using known input data and known key. This MAC value is then compared to the expected MAC value to test that the HMAC and hash operations are working correctly. If the values differ, the test fails. If they are the same, the test passes. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that a cryptographic algorithm test using a KAT shall be conducted for all cryptographic functions of each Approved cryptographic algorithm implemented by the TOE.

[61] CRC – Cyclic Redundancy Check
[62] EDC – Error Detection Code

- SHA KAT with SHA-1, SHA-256: The SHA implementation is further tested in a SHA KAT. Again, a known input data is used and a hash is created of the input data. This hash is compared to the expected hash to check that the hash operation is working correctly. If the values differ, the test fails. If they are the same, the test passes. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that a cryptographic algorithm test using a KAT shall be conducted for all cryptographic functions of each Approved cryptographic algorithm implemented by the TOE.
- NIST SP800-90A Hash DRBG: Known values are used to seed and initialize the DRBG. A block of random data is then generated by the DRBG and compared to a value pre-generated using the same known values to test that the DRBG is working correctly. If the random data blocks are the same, the test is passed. Otherwise, it is failed. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that a cryptographic algorithm test using a KAT shall be conducted for all cryptographic functions of each Approved cryptographic algorithm implemented by the TOE.
- RSA sign gen/verify KAT: A known private 2048-bit key (with SHA-256) is used to sign a known block of data, and the resultant value is compared with the expected ciphertext to check that the encrypt operation is working correctly. If they differ, the test fails. If they are the same, then the public key is used to decrypt the ciphertext and the output is compared to the original data to test that the decrypt operation is working correctly. If they are the same, the test passes. Otherwise, it is failed. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules (section 4.9.1) which requires that a cryptographic algorithm test using a KAT shall be conducted for all cryptographic functions of each Approved cryptographic algorithm implemented by the TOE.

In addition, the following conditional tests are performed during normal operation of the TOE:

- PCT for RSA keypairs: This test is activated whenever an asymmetric RSA keypair is generated by the module to verify that the RSA key agreement functions are working correctly. The RSA private key is used to sign a block of data. The resulting signature is compared to the original data before it was signed. If the two values are equal, then the test fails. If the two values differ, the RSA public key is used to verify the signature and the resulting value is compared to the original data. If they are the same, the test is passed. Otherwise, it is failed.
- Firmware Load Test: Prior to upgrading the TOE firmware/software image, the module verifies that the upgrade tar files is properly signed by verifying the signature against an externally calculated signature. The verification uses a 2048-bit RSA key with SHA256 digest.
- Continuous Random Number Generator Test for DRBG: This test is activated on the DRBG implementation whenever a fresh random value is requested. The new random number returned from the DRBG will be compared with the previous random number from the same DRBG to determine if stuck-at-constant type of failure is occurring. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules which requires that a continuous random number generator test is performed by the TOE on each RNG according to the rules specified in section 4.9.2 of the FIPS PUB 140-2 standard.
- Continuous Random Number Generator Test for NDRNG: This test is activated on the NDRNG implementation whenever a fresh random value is requested. The new random number returned from the NDRNG will be compared with the previous random number from the same NDRNG to determine if stuck-at-constant type of failure is occurring. This test meets the FIPS PUB 140-2, Security Requirements for Cryptographic Modules which requires that a continuous random number generator test is performed by the TOE on each RNG according to the rules specified in section 4.9.2 of the FIPS PUB 140-2 standard.
- DRBG Health Checks: instantiate, uninstantiate, generate, and reseed:
    - Instantiate: Testing done before the instantiation of a new DRBG to verify that DRBG instantiate operation is working correctly. The DRBG instantiation algorithm is sent fixed values of entropy, nonce, and personalization string. The output is compared with the value that is expected. If the values match, the test passes. Otherwise the test fails. Error testing is done by forcing an error

upon the algorithm. If the algorithm handles the error as expected, the test passes. Otherwise the test fails.

- Generate: Testing done before the first use of the DRBG to verify that DRBG generate operation is working correctly. The DRBG generate function tests both the instantiate and reseed algorithms. KAT is performed for security strength supported and for prediction resistance (if supported). The number of bits requested, additional input (if supported), working internal state, are supplied to the generate function. If the values used during the test produce the expected results and the errors are handled as expected, the test passes. Otherwise it fails.
- Reseed: Testing done before reseeding the DRBG instantiation function or before the generation of a new random number to verify that DRBG reseed operation is working correctly. The DRBG reseed algorithm is sent fixed values of entropy, additional input, C, and V. The output is compared with the value that was expected. If the values match, the test passes. Otherwise the test fails. Error testing is done by forcing an error upon the algorithm. If the algorithm handles the error as expected, the test passes. Otherwise the test fails.
- Uninstantiate: This test is performed whenever the instantiate, generate, or reseed tests are executed to verify that the DRBG uninstantiate operation is working correctly. It demonstrates that error handling is performed correctly and zeroizes the internal state. This test passes if the internal state is zeroized. Otherwise the test fails.

TOE Security Functional Requirements Satisfied: FPT_APW_EXT.1, FPT_SKP_EXT.1, FPT_STM.1, FPT_TST_EXT.1, FPT_TUD_EXT.1

8.1.6 TOE Access

The TOE can be accessed remotely via TLS v1.2 connections or locally through a direct connection from the management workstation to the management port of the NTO appliance. Before an administrative session can be established, the NTO Java Console and NTO Web Console display a login banner (configured by a Security Administrator) warning against unauthorized use of the TOE. The NTO ATIP GUI is a new TLS session within the same administrative user session of the NTO Java Console or NTO Web Console. The NTO Java Console and NTO Web Console include a warning against unauthorized use of the NTO ATIP GUI in their respective login banners before an administrative session can be established.

The NTO Java Console and NTO Web Console allow administrators to terminate their own interactive sessions by logging out. Otherwise, the NTO Java Console and NTO Web Console terminate local inactive sessions based on a specified time period of inactivity configured by a Security Administrator. When the NTO Java Console and NTO Web Console sessions are terminated, any new NTO ATIP GUI sessions that were established within the NTO Java Console and NTO Web Console administrative sessions are also terminated.

TOE Security Functional Requirements Satisfied: FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4, FTA_TAB.1.

8.1.7 Trusted Path/Channels

All secure channels provided by the TOE conform to the TLS requirements in Section 7.2.2. The TOE communicates with authorized external IT entities via the following secure channel:

- TLS v1.2 for connections to the Syslog server

The TOE provides trusted paths via RMI over TLS v1.2 for connections to the NTO Java Console and HTTPS over TLS v1.2 for connections to the NTO Web Console. The trusted paths provided by the TOE conform to the TLS requirements in Section 7.2.2.

TOE Security Functional Requirements Satisfied: FTP_ITC.1, FTP_TRP.1.

9. Rationale

9.1 Conformance Claims Rationale

This Security Target extends Part 2 and conforms to Part 3 of the Common Criteria Standard for Information Technology Security Evaluations, Version 3.1 Release 4. This ST conforms to the ND cPP.

9.1.1 Variance Between the PP and this ST

There is no variance between the ND cPP and this ST.

9.1.2 Security Assurance Requirements Rationale

This ST claims exact conformance to the ND cPP, including the assurance requirements listed in Section 6 of the ND cPP.

9.1.3 Dependency Rationale

The SFRs in this Security Target represent the SFRs identified in the ND cPP v1.0. As such, the ND cPP v1.0 SFR dependency rationale is deemed acceptable since the PP itself has been validated.

10. Acronyms and Terms

This section describes the acronyms and terms used throughout the document.

10.1 Acronyms

Table 9 – Acronyms

Acronym	Definition
AFM	Advanced Feature Module
AES	Advanced Encryption Standard
API	Application Programming Interface
ATIP	Application and Threat Intelligence Processor
C	Country
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining Mode
CC	Common Criteria
CEM	Common Evaluation Methodology
CFP	C Form-Factor Pluggable
CM	Configuration Management
CN	Common Name
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRL	Certificate Revocation List
DNS	Domain Name System
DRBG	Deterministic Random Bit Generator
DRNG	Digital Random Number Generator
DSS	Digital Signature Standard
EAL	Evaluation Assurance Level
EDC	Error Detection Code
FIPS	Federal Information Processing Standard
G	Gigabit
GB	Gigabyte
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code

Acronym	Definition
HTTPS	Hypertext Transfer Protocol over SSL
ID	Identifier
IEC	International Electrotechnical Commission
I/O	Input/Output
IP	Internet Protocol
IRQ	Interrupt Request Channel
ISO	International Organization for Standardization
IT	Information Technology
ITSEF	Information Technology Security Entrepreneurs Forum
JRE	Java Runtime Environment
KAT	Known Answer Test
MAC	Message Authentication Code
Mb	Megabit
ND cPP	collaborative Protection Profile for Network Devices v1.0
NDRNG	Non-Deterministic Random Number Generator
NEBS	Network Equipment-Building System
NIST	National Institute of Standards and Technology
NTO	Network Tool Optimizer
NTP	Network Time Protocol
NVOS	Network Visibility Operating System
O	Organization
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OSP	Organizational Security Policy
OU	Organizational Unit
PCM	Packet Capture Module
PCT	Pairwise Consistency Test
PKCS	Public Key Cryptography Standard
PP	Protection Profile
PSS	Probabilistic Signature Scheme
PUB	Publication
QSFP+	Quad Small Form-Factor Pluggable Enhanced
RADIUS	Remote Authentication Dial-In User Service
RBG	Random Bit Generator

Acronym	Definition
REST	Representational State Transfer
RFC	Request For Comment
RMI	Remote Method Invocation
RSASSA	RSA Signature Scheme with Appendix
RU	Rack Unit
SAR	Security Assurance Requirement
SD	Supporting Document
SFP+	Small Form-Factor Pluggable Enhanced
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SPAN	Switched Port Analyzer
SSL	Secure Sockets Layer
ST	Security Target
TACACS+	Terminal Access Controller Access-Control System Plus
TAP	Test Access Point
Tb	Terabit
Tcl	Tool Command Language
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality

10.2 Terms

Table 10 – Terms

Name	Definition
Administrator	See Security Administrator.
Assurance	Grounds for confidence that a TOE meets the SFRs.
Security Administrator	The terms “Administrator” and “Security Administrator” are used interchangeably in this document at present.
TSF Data	Data for the operation of the TSF upon which the enforcement of the requirements relies.

Prepared by:
Corsec Security, Inc.

13921 Park Center Road, Suite 460
Herndon, VA 20171
United States of America

Phone: +1 703 267 6050
Email: info@corsec.com
http://www.corsec.com