Imperva SecureSphere

Security Target

Version 0.4
12 November 2015

Prepared for:

Imperva Inc.
3400 Bridge Parkway, Suite 200
Redwood Shores, CA 94065
United States

Prepared by:

Leidos Inc. (formerly Science Applications International Corporation)

Common Criteria Testing Laboratory
6841 Benjamin Franklin Drive, Columbia, Maryland 21046

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is Imperva SecureSphere v11.5 Patch 5 running on two or more of the Imperva appliances listed below in Section 1.1, including one or more Management Servers and one or more Gateways. The product is a set of network security appliances for advanced threat detection with Intrusion Prevention System (IPS) capabilities. SecureSphere provides protection from attacks against database, file, Web, and Web Services assets, both within the organization (insider attacks) and from without. Installed on the network as a reverse Hypertext Transfer Protocol (HTTP) proxy, a transparent inline bridge or as an offline network monitor (sniffer), a SecureSphere Gateway monitors application-level protocols for attacks, and reacts by blocking the attacks and/or reporting them to a centralized management server. The focus of this evaluation is on the TOE functionality supporting the claims in the *Protection Profile for Network Devices* (See section 1.2 for specific version information). The security functionality specified in [NDPP] includes protection of communications between TOE components and trusted IT entities, identification and authentication of administrators, auditing of security-relevant events, ability to verify the source and integrity of updates to the TOE, and specifies FIPS-validated cryptographic mechanisms.

The Security Target contains the following additional sections:

-

TSF TOE Security Function(s)

- TOE Description (Section 0)
- Security Problem Definition (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Protection Profile Claims (Section 7)
- Rationale (Section 8).

1.1 Security Target, TOE and CC Identification

ST Title – Imperva SecureSphere Security Target

ST Version – Version 0.4

ST Date – 12 November 2015

TOE Identification –Imperva SecureSphere v11.5 Patch 5 software running on two or more of the Imperva appliances listed below, including one or more Management Servers and one or more Gateways:

Hardware appliance:

- Gateway Appliances: X1010, X10K, X2010, X2510, X4510, X6510, X8510
- MX Management Server Appliances: M110, M160

Virtual Machine Appliances:

- V1000, V2500, V4500 (for Gateway)
- VM150 (for MX)

And optionally a SecureSphere Operations Manager (SOM) Management Server appliance or Virtual Machine Appliance:

- Appliances: M160
- Virtual Machine Appliance: VM150

Note: MX and SOM are two management applications using the same code base. MX, Gateway and SOM are all installed using the same image, during installation user choses the application to install.

TOE Developer – Imperva, Inc.

Evaluation Sponsor – Imperva, Inc.

CC Identification – *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012*

1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- This ST is conformant to the *Protection Profile for Network Devices*, Version 1.1, 8 June 2012 (NDPP) as amended by Errata #3 dated 3 November 2014, and includes the additional optional SFRs: FCS_HTTPS_EXT.1, FCS_SSH_EXT.1, FCS_TLS_EXT.1, and FPT_ITT.1.
- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
    - Part 2 Extended

- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
    - Part 3 Conformant

1.3 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a number in parentheses placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement, (1) and (2).
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …"). Note that 'cases' that are not applicable in a given SFR have simply been removed without any explicit identification.
- The NDPP uses an additional convention – the 'case' – which defines parts of an SFR that apply only when corresponding selections are made or some other identified conditions exist. Only the applicable cases are identified in this ST and they are identified using **bold** text.
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.3.1 Terminology

This section identifies TOE-specific terminology.

Bridge	A layer-two device that forwards frames received from one network segment to another segment, based on their MAC address
Correlated Attack Validation	An Imperva technology that addresses attacks by basing ID decisions on multiple observations.
Dynamic Profiling	An Imperva technology that creates and maintains a comprehensive model (profile) of an application's legitimate protocol structure and dynamics through the examination of live traffic.
TAP	A device that provides a non-intrusive fault-tolerant method of viewing traffic on a network segment.

1.3.2 Abbreviations

This section identifies abbreviations and acronyms used in this ST.

Abbreviation	Description
ADC	Application Defense Center
AES	Advanced Encryption Standard
AWS	Amazon Web Services
CAV	Correlated Attack Validation
CC	Common Criteria
CLI	Command Line Interface
GUI	Graphical User Interface
HSM	Hardware Security Module
HTTP	Hypertext Transfer Protocol
ID	Intrusion Detection
IDS	Intrusion Detection System
IP	Internet Protocol
IPS	Intrusion Prevention System
MX	Management Server
NIC	Network Interface Card
NTP	Network Time Protocol
OOB	Out Of Band
PP	Protection Profile
RFC	Request for Comment
SFR	Security Functional Requirement
SFP	Security Function Policy
SIEM	Security Information and Event Management
SOM	SecureSphere Operations Manager
SPAN	Switch Port Analyzer
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function(s)

2. TOE Description

2.1 Product Description

The TOE is Imperva SecureSphere v11.5 Patch 5 running on two or more of the Imperva appliances listed in Section 1.1, including one or more Management Servers and one or more Gateways. The product is a set of network security appliances for advanced threat detection with Intrusion Prevention System (IPS) capabilities. SecureSphere provides protection from attacks against database, file, Web, and Web Services assets, both within the organization (insider attacks) and from without. Installed on the network as a transparent inline bridge or as an offline network monitor (sniffer), a SecureSphere Gateway monitors application-level protocols for attacks, and reacts by blocking the attacks and/or reporting them to a centralized management server. Though the focus of this evaluation is on the TOE functionality supporting the claims in the Protection Profile for Network Devices this section includes descriptions of additional product security functionality for usage and context purposes.

The product is deployed as one or more Gateway appliances controlled by a MX appliance. In multi-tier management configurations, one or more MX Management Servers may in turn be managed by a SecureSphere Operations Manager (SOM) Management Server.

Administrators connect to the Management Server using a standard Web browser (outside of the Target of Evaluation). They are required to authenticate their identity before being allowed any further action.

The different appliance models all run the same SecureSphere v11.5 Patch 5 software and provide all claimed security functionality, but may differ in throughput and storage capacity. The SecureSphere v11.5 Patch 5 software (including management and/or Gateway components) may alternatively be installed on a Virtual Machine (VM) hosted by a VMware ESXi Hypervisor or on Amazon appliance in AWS environment. The Virtual Machine emulates the SecureSphere appliance hardware. The VMware Hypervisor, and underlying hardware is considered to be outside of the boundaries of the Target of Evaluation. Amazon deployments are not included in the TOE.

Imperva's Dynamic Profiling technology automatically builds a model of legitimate application behavior that is used by the product to identify illegitimate traffic. In addition, attack signatures are preconfigured into the product and can be periodically updated from an external Application Defense Center (ADC). The ADC also provides ADC Insights – these are pre-packaged security policy rules and reports for commonly used applications. This functionality is outside the scope of the TOE.

SecureSphere applies different layers of intrusion detection logic to analyzed network traffic, as depicted below in Figure 1-1. Some of these layers are applicable to all network traffic; some are relevant only for Web traffic and/or database access protocols. In addition, Imperva's Correlated Attack Validation (CAV) technology examines sequences of events and identifies suspicious traffic based on a correlation of multiple analysis layers. Identified malicious traffic is blocked.

SecureSphere supports the following two blocking methods:

- TCP Reset (sniffing topology): SecureSphere can signal protected servers to disconnect malicious users using TCP reset, a special TCP packet that signals TCP peers to close the TCP session. SecureSphere spoofs a TCP reset packet and sends it to the protected server. It is assumed that a standards-conformant server would immediately drop the attacker's session on receipt of the TCP reset packet.

  Note: TCP reset is considered inferior to inline blocking (see below) because it does not actively block the malicious traffic from reaching the server; blocking depends on the server's correct and timely session termination behavior.

- Inline Blocking: the gateway drops the packet, so that it doesn't reach its intended destination, and sends a TCP reset to the server.

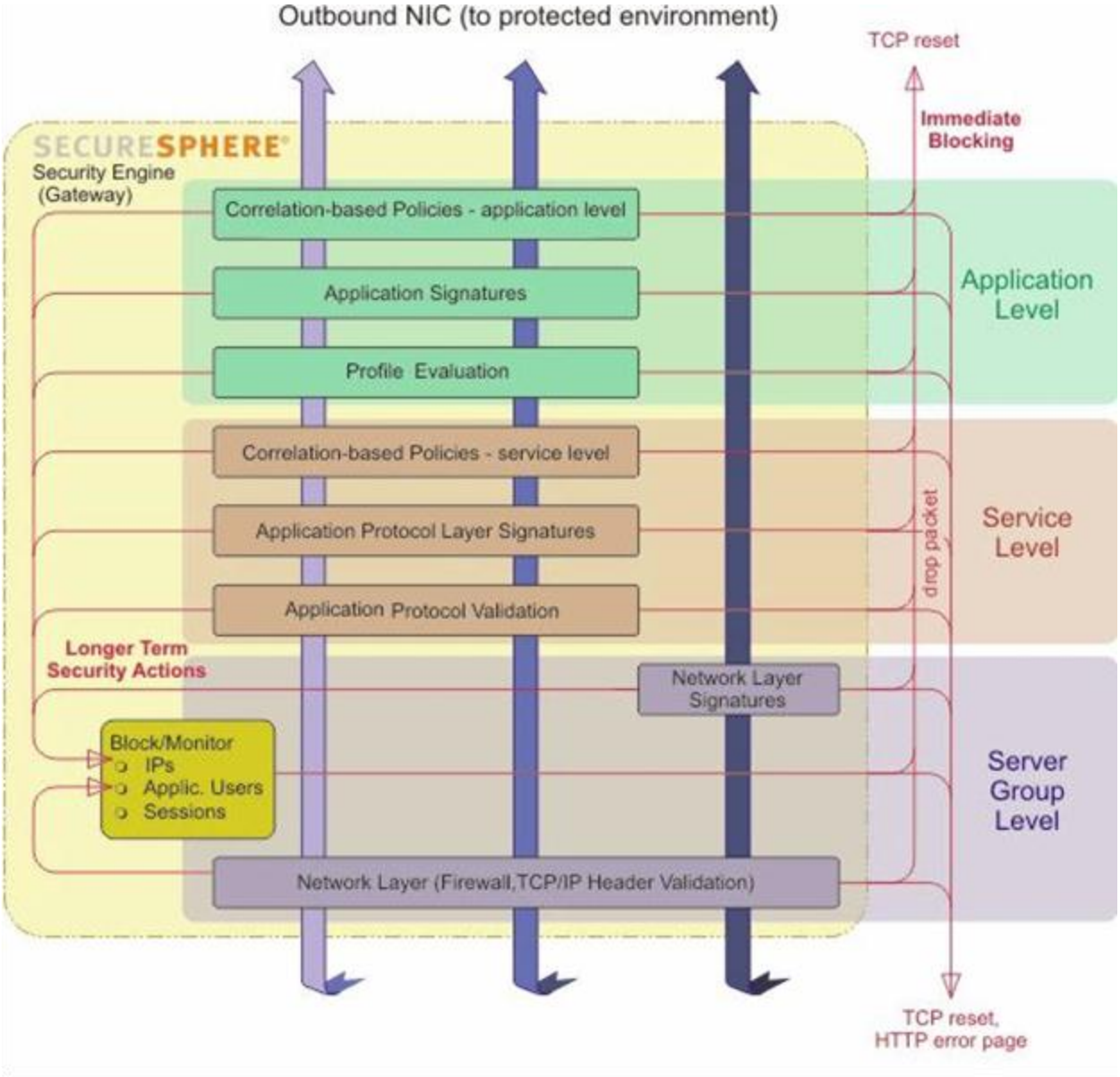


Figure 1-1 Intrusion Analysis and Reaction

The product's application auditing capability is augmented by a discovery and assessment capability that scans databases and file servers for known vulnerabilities and policy violations, identifies sensitive data, and enables automatic aggregation and review of user rights across the organization. The product can also integrate information from external sources such as web vulnerability scanners. This functionality is outside the scope of the TOE. The TOE includes a security event auditing functionality which is included in the TOE.

MX Management Server can communicate with Imperva Update Server to download TOE updates. When MX downloads a new update package, it is hashed using SHA-256. Authorized administrators can download the TOE updates and manually verify the hash prior to installing the updates. The communication channel between the TOE and the Imperva Update Server is protected using TLS.

TLS is also used for secure transmission of audit records to an external syslog server.

MX Management Servers interact with authorized administrators via SSH using OpenSSL; or using a web browser where RSA BSAFE is used to implement HTTP over TLS (HTTPS) to secure the underlying communications. MX

also offers a local console interface for management of the TOE. Secure connections are also established between the distributed TOE components: namely MX Management Servers (including SOMs) and Gateways.

The TOE is operated in FIPS mode. For cryptographic functions, the X1010, X10K, X2010, X2510, X4510, X6510, X8510, V1000, V2500, and V4500 appliances include FIPS Object Module 2.0.1, which is a FIPS 140-2 validated cryptomodule (#1747). The M110, M160, VM150 appliances include BSAFE Crypto-J 6.1.3, which incorporates the FIPS 140-2 validated RSA JCE 6.1 (FIPS 140-2 cert #2057/2058).

The TOE provides several predefined user roles, but only the Administrator corresponding to the protection profile role: Security Administrator can manage all of the TOE security functions. Other roles only have a subset of TOE access capabilities.

2.2 TOE Overview

The Target of Evaluation (TOE) is a combination of Imperva appliances each running Imperva SecureSphere v11.5 Patch 5. More specifically, the TOE consists of

- One[1] MX Management Server appliance; and
- One or more Gateway appliances; and optionally:
- One SecureSphere Operations Manager (SOM) Management Server appliance.

Figure 1-2 - SecureSphere Gateway Appliance

SecureSphere provides protection from attacks against database, file, Web, and Web Services assets, both within the organization (insider attacks) and from without. Installed on the network as a reverse Hypertext Transfer Protocol (HTTP) proxy, a transparent inline bridge or as an offline network monitor (sniffer), a SecureSphere Gateway monitors application-level protocols for attacks, and reacts by blocking the attacks and/or reporting them to a centralized management server.

The TOE is deployed as one Gateway appliance controlled by a MX appliance that in turn is managed by a SecureSphere Operations Manager (SOM) Management Server. The Management Server (MX) and SOM are two different management applications using the same code base. SOM is a manager of MXs and runs M110 and M160 M-Appliances. SOM runs on an M160 M-Appliance.

Administrators connect to the Management/SOM Servers using a standard Web browser (outside of the Target of Evaluation). The TOE authenticates Administrators using a local password based mechanism. Additionally, the TOE can be configured to use the services of a trusted LDAP server in the operational environment.

As noted above, all TOE appliance models run the same software and provide all claimed security functionality.

The TOE generates logs for security relevant events including the events specified in NDPP. The TOE can be configured to store the logs locally so they can be accessed by an administrator and can also be configured to send the logs to a designated external log server. The communication channel with the external syslog server is protected using TLS.

The TOE is operated in FIPS mode and includes cryptographic modules from both RSA BSAFE Crypto-J 6.1.3 FIPS 140-2 validated cryptographic module and OpenSSL 2.0.1 FIPS-certified with certificate number 1747.

[1] Onebox mode (where both the SecureSphere management server and SecureSphere gateway are integrated in a single machine) is not included in the evaluated configuration

SecureSphere Deployment Scenarios include both non-inline (sniffing) and inline gateways. An inline gateway is more invasive but provides better blocking capabilities. A sniffing gateway is totally noninvasive but provides less effective blocking capabilities.

In the inline scenario, the gateway acts as a bridging device between the external network and the protected network segment. The gateway will block malicious traffic inline (i.e. drop packets). A single inline gateway protects one to four network segments. For examples of deployment types, see the TOE guidance documentation.

X1010 and X2010 appliance models have six network interface ports. Two of the ports are used for management: one to connect to the management server and the other is optional. The other four ports are part of two bridges that are used for inline inspection of up to two different protected network segments.

X2510, X4510, X6510, X8510 and X10K appliance models have 2 fixed network interface ports and 2 optional network cards, each card contains two to four network ports. The two fixed ports are used for management: one to connect to the management server and the other is optional. The other optional two cards (each with two to four ports) are part of one to four bridges that are used for inline inspection of up to four different protected network segments.

The MX Management Server Appliances: M110, M160 each have an RJ45 Connector Serial Port, two USB ports, and two management ports. The appliance models differ only in size of memory and hard drive. While the physical form factor of each appliance differs, the underlying hardware shares a similar architecture.

A sniffing gateway is a passive sniffing device. It connects to corporate hubs and switches and taps the traffic sent to and from protected servers, using a SPAN (Switch Port Analyzer) mirror port on the switch, or a dedicated TAP device. Traffic is copied to it instead of passing directly through it. Transmission Control Protocol (TCP) resets are transmitted over a "blocking" Network Interface Card (NIC). For an example of this type of deployment, see the TOE guidance documentation.

2.3 TOE Architecture

The section describes the TOE physical and logical boundaries.

2.3.1 Physical Boundaries

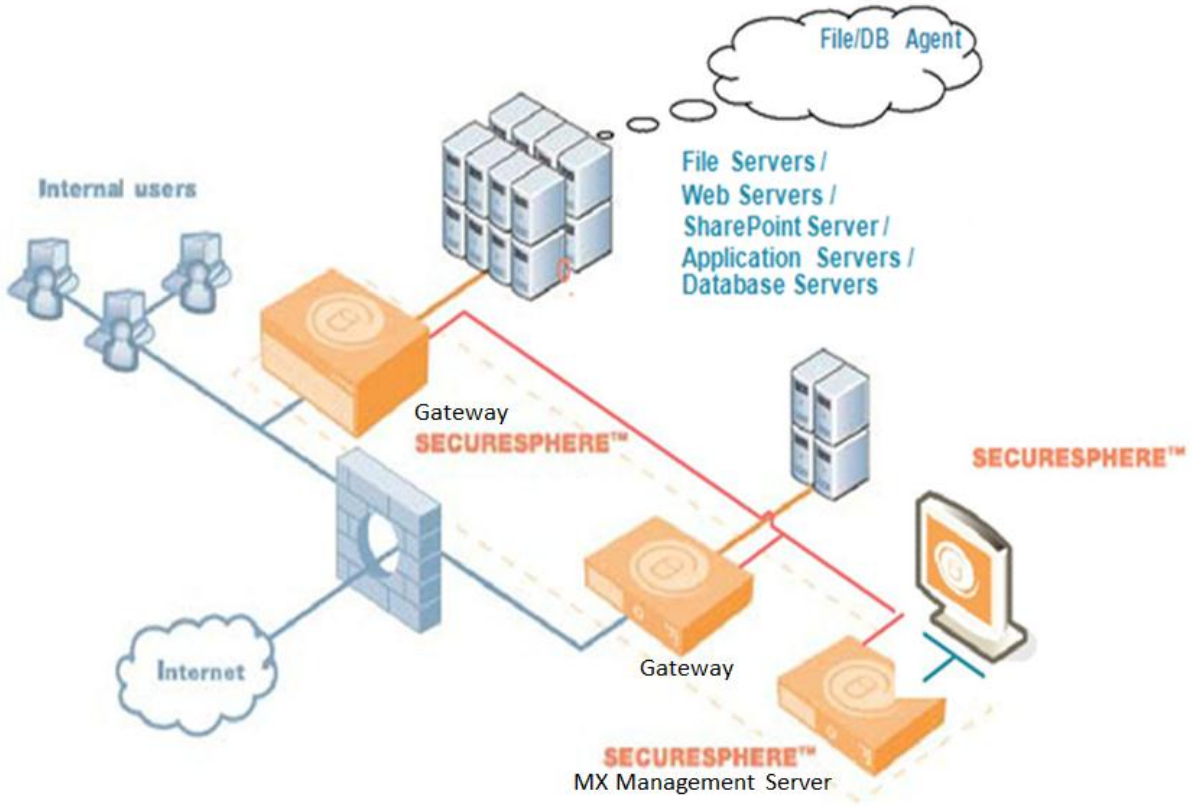
As explained above, a given Imperva configuration includes one or more Gateway appliances controlled by a MX (Management Server) appliance. In multi -tier management configurations, one or more MX Management Servers may in turn be managed by a SecureSphere Operations Manager (SOM) Management Server. Each Imperva appliance is a self-contained hardware appliance or VM designed to interact with its environment via network connections.

Figure 1-4 below depicts the TOE in its operational environment. SecureSphere Gateways are installed in front of the protected resources. They are connected to the Management Server using dedicated out of band (OOB) management network interfaces, and the communication between the gateways and the Management Server is protected using HTTPS. SOM Management Servers (not shown) are in the TOE and would be connected to the Management Server and communications protected using HTTPS. The operational environment also requires LDAP, TOE Update, syslog, and NTP servers.

Descriptions of the interconnecting lines in the figure are as follows:

- The Blue Line to the monitor represents the TLS protected communications between the administrator workstation and the MX Management Server.
- The Red Lines represent the TLS protected communications between the TOE components and between instances of a TOE.
- The Orange Lines represent the Internet data filtered and processed by the TOE.

Figure 1-4 Physical Scope and Boundaries of the TOE



The virtual (VM) appliances are delivered as an installation disk (or ISO image). They require that the minimum following hardware and software be installed on the host system:

- VMware ESXi 5.x with virtual hardware version 9.0 and newer
- Dual core or higher number of cores, Intel based server
- IvyBridge supported Microprocessor (http://en.wikipedia.org/wiki/List_of_Intel_Xeon_microprocessors#Ivy_Bridge-based_Xeons) or newer generation of Intel based CPUs: 3rd Generation Intel Core processors, Intel Xeon processor E3-1200 v2 product family, Next Generation Intel Xeon processors, Intel Xeon processor E5 v2 and E7 v2 families or newer Intel Xeon processors.
- 250 GB Hard Drive
- Hypervisor-supported network interface card
- If ESXi is in cluster, the EVC level must be set to L5 (IvyBridge) or higher
- The main reason for using IvyBridge CPUs for Common Criteria is because of the need to use RDRAND command, any IvyBridge CPU is compatible with this requirement and therefore all IvyBridge CPUs can be used as ESX servers.

The VM appliances also require the following minimums:

	VM150	**V1000**	**V2500**	**V4500**
CPU	2	2	2	4
Memory	4GB	4 GB	4 GB	8 GB
Disk Space	160GB	160 GB	160 GB	160 GB

All of the VM Machines in the CCTL test configuration were tested on an Ivy Bridge supported Intel Core i5-3350P processor @ 3.10 GHz with VMware ESXi v5.1.0 with virtual hardware version 9.

All appliance hardware and software is included in the TOE, with the following exceptions.

- Hardware Security Module (HSM) and SSL Accelerator Cards: SecureSphere Gateway appliances may be purchased with an internal HSM or SSL accelerator PCI card that offloads key storage and cryptographic operations used for network traffic deciphering from the appliance CPU. The cards are not included in the evaluated configuration.
- The TOE monitors network traffic between clients and servers in real-time, provides analyses of that traffic for suspected intrusions, and provides a reaction capability. Database auditing allows you to record selected user database queries for audit purposes. Web and file server queries and responses can also be selectively recorded. In addition, monitored databases can be actively scanned to identify potential vulnerabilities. The Imperva Agent software is installed on the client machines to support these features and may be used in the evaluated configuration but the agents are considered to be outside the scope of the TOE and the functionality was not evaluated or tested.
- The VMware ESXi Hypervisor virtualization software and the hardware it is installed on.

2.3.1.1 Software Requirements

In order for an MX Management Client to connect via web-based, remote access, the following software is required on the client machine(s):

- Browser: Microsoft Internet Explorer; Chrome, Firefox; or Apple Safari
- Adobe Flash Player

The TOE also requires that the administrator client machine have SSHv2 client software in order to connect to the CLI remotely.

Use of the syslog, TOE Updates, and optional external authentication methods require LDAP, TOE Update and syslog servers. The TOE also requires an NTP Server in the operational environment in order to synchronize its clock with that of the external time server.

2.3.2 Logical Boundaries

This section summarizes the security functions provided by the TOE:

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

2.3.2.1 Security audit

The TOE is designed to be able to generate logs for security relevant events including the events specified in NDPP. The TOE can be configured to store the logs locally so they can be accessed by an administrator and can also be configured to send the logs to a designated external log server.

2.3.2.2 Cryptographic support

The TOE is operated in FIPS mode and includes both NIST-validated RSA B-Safe and NIST-validated OpenSSL cryptographic modules. The modules provide key management, random bit generation, encryption/decryption, digital signature and cryptographic hashing and keyed-hash message authentication features in support of higher level cryptographic protocols, including SSH, TLS and HTTP over TLS.

2.3.2.3 User data protection

The TOE is designed to ensure that it does not inadvertently reuse data found in network traffic.

2.3.2.4 Identification and authentication

The TOE requires users (i.e., administrators) to be successfully identified and authenticated before they can access any security management functions available in the TOE. The TOE offers a network accessible GUI (HTTP over TLS) and console available locally and remotely via SSH for interactive administrator sessions.

The TOE supports the local (i.e., on device) definition of administrators with usernames and passwords. Additionally, the TOE can be configured to use the services of a trusted LDAP server in the operational environment.

2.3.2.5 Security management

The TOE provides remote access to a CLI using SSHv2 and a GUI to access the wide range of security management functions. Security management commands are limited to administrators and are available only after they have provided acceptable user identification and authentication data to the TOE.

The TOE also provides the ability to manage the TOE locally via direct serial console connection. The direct serial console is used mainly for initial configuration and then the TOE is designed to be managed and monitored using the Web GUI from a remote HTTPS/TLS client; or using the CLI from an SSHv2 client. The TOE provides the Administrator role which corresponds to the NDPP Security Administrator.

2.3.2.6 Protection of the TSF

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

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. The TOE includes its own time clock to ensure that reliable time information is available (e.g., for log accountability) but requires an NTP Server in the operational environment in order to synchronizes its clock with that of the external time server.

The TOE uses HTTPS to protect communications between distributed TOE components (FPT_ITT.1).

The TOE includes functions to perform self-tests so that it might detect when it is failing. It also includes mechanisms so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE.

2.3.2.7 TOE access

The TOE can be configured to display an informative banner that will appear prior to an administrator being permitted to establish an interactive session. The TOE subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session will be terminated. Administrators can also terminate their own sessions by logging out.

2.3.2.8 Trusted path/channels

The TOE protects interactive communication with remote administrators using SSH or HTTP over TLS. TLS ensures both integrity and disclosure protection.

The TOE uses TLS to ensure that any authentication operations, and exported audit records, are sent only to the configured Syslog or authentication servers so they are not subject to inappropriate disclosure or modification. TLS is also used to ensure TOE updates are transmitted securely..

2.4 TOE Documentation

Imperva offers a series of documents that describe the installation process for the TOE, as well as guidance for subsequent use and administration of the system security features.

- Imperva SecureSphere Configuring Common Criteria Compliance User Guide, v11.5, November 2015
- Imperva SecureSphere v11.5, Admin Guide Version 11.5, January 2015
- Imperva SecureSphere Operations Manager (SOM) User Guide Version 11.5, August 2015

3. Security Problem Definition

This security target includes by reference the Security Problem Definition (composed of organizational policies, threat statements, and assumptions) from the NDPP.

In general, the NDPP has presented a Security Problem Definition appropriate for network infrastructure devices, and as such is applicable to the Imperva TOE.

4. Security Objectives

Like the Security Problem Definition, this security target includes by reference the Security Objectives from the NDPP. The NDPP security objectives for the operational environment are reproduced below, since these objectives characterize technical and procedural measures each consumer must implement in their operational environment.

In general, the NDPP has presented a Security Objectives statement appropriate for network infrastructure devices, and as such is applicable to the Imperva TOE.

4.1 Security Objectives for the Operational Environment

OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

5. IT Security Requirements

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

The SFRs have all been drawn from the *Protection Profile for Network Devices, Version 1.1, 8 June 2012* (NDPP), as amended by Errata #3.

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

The SARs are the set of SARs specified in NDPP.

5.1 Extended Requirements

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

- FAU_STG_EXT.1: External Audit Trail Storage
- FCS_CKM_EXT.4: Cryptographic Key Zeroization
- FCS_HTTPS_EXT.1: Extended: HTTP Security (HTTPS)
- FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
- FCS_SSH_EXT.1: Explicit: SSH
- FCS_TLS_EXT.1: Extended: Transport Layer Security (TLS)
- FIA_PMG_EXT.1: Password Management
- FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
- FIA_UIA_EXT.1: User Identification and Authentication
- FPT_APW_EXT.1: Extended: Protection of Administrator Passwords
- FPT_SKP_EXT.1: Extended: Protection of TSF Data (for reading of all symmetric keys)
- FPT_TST_EXT.1: TSF Testing
- FPT_TUD_EXT.1: Extended: Trusted Update
- FTA_SSL_EXT.1: TSF-initiated Session Locking

5.2 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by the Fidelis XPS.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User identity association
	FAU_STG_EXT.1: External Audit Trail Storage
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM_EXT.4: Cryptographic Key Zeroization
	FCS_COP.1(1): Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1(2): Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3): Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1(4): Cryptographic Operation (for keyed-hash message authentication)
	FCS_HTTPS_EXT.1: Extended: HTTP Security (HTTPS)
	FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
	FCS_SSH_EXT.1: Explicit: SSH
	FCS_TLS_EXT.1: Extended: Transport Layer Security (TLS)
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management
	FIA_UAU.7: Protected Authentication Feedback
	FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
	FIA_UIA_EXT.1: User Identification and Authentication
FMT: Security management	FMT_MTD.1: Management of TSF Data (for general TSF data)
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1: Extended: Protection of Administrator Passwords
	FPT_ITT.1: Basic Internal TSF Data Transfer Protection
	FPT_SKP_EXT.1: Extended: 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: Extended: Trusted Update
FTA: 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
FTP: Trusted path/channels	FTP_ITC.1: Trusted Channel
	FTP_TRP.1: Trusted Path

Table 1 TOE Security Functional Components

5.2.1 Security audit (FAU)

5.2.1.1 Audit Data Generation (FAU_GEN.1)

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;

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

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 2**.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	
FAU_GEN.2	None.	
FAU_STG_EXT.1	None.	
FCS_CKM.1	None.	
FCS_CKM_EXT.4	None.	
FCS_COP.1(1)	None.	
FCS_COP.1(2)	None.	
FCS_COP.1(3)	None.	
FCS_COP.1(4)	None.	
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session. Establishment/Termination of a HTTPS session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures
FCS_RBG_EXT.1	None.	
FCS_SSH_EXT.1	Failure to establish an SSH session. Establishment/Termination of an SSH session.	Reason for failure
	Establishment/Termination of an SSH session.	Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_TLS_EXT.1	Failure to establish a TLS session. Establishment/Termination of a TLS session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures
FDP_RIP.2	None.	
FIA_PMG_EXT.1	None.	
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	
FMT_MTD.1	None.	
FMT_SMF.1	None.	
FMT_SMR.2	None.	
FPT_APW_EXT.1	None.	
FPT_ITT.1	None.	
FPT_SKP_EXT.1	None.	
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FPT_TST_EXT.1	None.	
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by	No additional information.

Requirement	Auditable Events	Additional Audit Record Contents
	the session locking mechanism.	
FTA_SSL.4	The termination of an interactive session.	No additional information.
FTA_TAB.1	None.	
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.	Identification of the claimed user identity.

Table 2 Auditable Events

5.2.1.2 User identity association (FAU_GEN.2)

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

5.2.1.3 External Audit Trail Storage (FAU_STG_EXT.1)

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 implementing the [***TLS***] protocol.

5.2.2 Cryptographic support (FCS)

5.2.2.1 Cryptographic Key Generation (for asymmetric keys) (FCS_CKM.1)

FCS_CKM.1.1 Refinement: The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with [

- ***NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for finite field-based key establishment schemes;***
- ***NIST Special Publication 800-56B, 'Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography' for RSA-based key establishment schemes***]

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.2.2.2 Cryptographic Key Zeroization (FCS_CKM_EXT.4)

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

5.2.2.3 Cryptographic Operation (for data encryption/decryption) (FCS_COP.1(1))

FCS_COP.1.1(1) Refinement: The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm [AES operating in [***CBC***]] and cryptographic key sizes 128-bits and 256-bits that meets the following:

- FIPS PUB 197, 'Advanced Encryption Standard (AES)'
- [***NIST SP 800-38A***].

5.2.2.4 Cryptographic Operation (for cryptographic signature) (FCS_COP.1(2))

FCS_COP.1.1(2) Refinement: The TSF shall perform cryptographic signature services in accordance with a [

(1) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater~~, or~~]

that meets the following:

Case: RSA Digital Signature Algorithm

- FIPS PUB 186-2 or FIPS PUB 186-3, "Digital Signature Standard"

5.2.2.5 Cryptographic Operation (for cryptographic hashing) (FCS_COP.1(3))

FCS_COP.1.1(3) Refinement: The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-512***] and message digest sizes [***160, 256, 512***] bits that meet the following: FIPS Pub 180-3, 'Secure Hash Standard.'

5.2.2.6 Cryptographic Operation (for keyed-hash message authentication) (FCS_COP.1(4))

FCS_COP.1.1(4) Refinement: The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC-[***SHA-1***], key size [**160 bits**], and message digest sizes [***160***] bits that meet the following: FIPS Pub 198-1, 'The Keyed-Hash Message Authentication Code', and FIPS Pub 180-3, 'Secure Hash Standard.'

5.2.2.7 Extended: HTTP Security (HTTPS) (FCS_HTTPS_EXT.1)

FCS_HTTPS_EXT.1.1

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

FCS_HTTPS_EXT.1.2

The TSF shall implement HTTPS using TLS as specified in FCS_TLS_EXT.1.

5.2.2.8 Extended: Cryptographic Operation (Random Bit Generation) (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [***NIST Special Publication 800-90 using [CTR_DRBG (AES)]***] seeded by an entropy source that accumulated entropy from [***a software-based noise source***].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

5.2.2.9 Explicit: SSH (FCS_SSH_EXT.1)

FCS_SSH_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [***5656***].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [***password-based***][2].

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

FCS_SSH_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, [***no other algorithms***].

FCS_SSH_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses [***SSH_RSA***] and [***no other public key algorithms***] as its public key algorithm(s).

[2] Marked as a selection per NIAP Technical Decision TD0032 (https://www.niap-ccevs.org/Documents_and_Guidance/view_td.cfm?td_id=34)

FCS_SSH_EXT.1.6 The TSF shall ensure that data integrity algorithms used in SSH transport connection is [***hmac-sha1***].

FCS_SSH_EXT.1.7 The TSF shall ensure that diffie-hellman-group14-sha1 and [***no other methods***] are the only allowed key exchange methods used for the SSH protocol.

5.2.2.10 Extended: Transport Layer Security (TLS) (FCS_TLS_EXT.1)

FCS_TLS_EXT.1.1

The TSF shall implement one or more of the following protocols [***TLS 1.0 (RFC 2246), TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)*** **]** supporting the following ciphersuites:

Mandatory Ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA,

Optional Ciphersuites:

[***None***

].

5.2.3 User data protection (FDP)

5.2.3.1 Full Residual Information Protection (FDP_RIP.2)

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***allocation of the resource to***] all objects.

5.2.4 Identification and authentication (FIA)

5.2.4.1 Password Management (FIA_PMG_EXT.1)

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

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

5.2.4.2 Protected Authentication Feedback (FIA_UAU.7)

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

5.2.4.3 Extended: Password-based Authentication Mechanism (FIA_UAU_EXT.2)

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [***[and access to external LDAP Server]***] to perform administrative user authentication.

5.2.4.4 User Identification and Authentication (FIA_UIA_EXT.1)

FIA_UIA_EXT.1.1 The TSF shall allow responses to 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;
- [***[Enter User Credentials in the login GUI]***].

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

5.2.5 Security management (FMT)

5.2.5.1 Management of TSF Data (for general TSF data) (FMT_MTD.1)

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

5.2.5.2 Specification of Management Functions (FMT_SMF.1)

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 update the TOE, and to verify the updates using the [***published hash***] capability prior to installing those updates;
- **[*Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1;***
- ***Ability to configure the cryptographic functionality***].

5.2.5.3 Restrictions on Security Roles (FMT_SMR.2)

FMT_SMR.2.1 The TSF shall maintain the roles:

- Authorized 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

- Authorized Administrator role shall be able to administer the TOE locally;
- Authorized Administrator role shall be able to administer the TOE remotely;

are satisfied.

5.2.6 Protection of the TSF (FPT)

5.2.6.1 Extended: Protection of Administrator Passwords (FPT_APW_EXT.1)

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

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

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

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

5.2.6.3 Basic Internal TSF Data Transfer Protection

FPT_ITT.1.1 Refinement: The TSF shall protect TSF data from disclosure and detect its modification when it is transmitted between separate parts of the TOE through the use **of** [***TLS, TLS/HTTPS***].

5.2.6.4 Reliable Time Stamps (FPT_STM.1)

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

5.2.6.5 TSF Testing (FPT_TST_EXT.1)

FPT_TST_EXT.1.1 The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

5.2.6.6 Extended: Trusted Update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1 The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3 The TSF shall provide a means to verify firmware/software updates to the TOE using a [***published hash***] prior to installing those updates.

5.2.7 TOE access (FTA)

5.2.7.1 TSF-initiated Termination (FTA_SSL.3)

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

5.2.7.2 User-initiated Termination (FTA_SSL.4)

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

5.2.7.3 TSF-initiated Session Locking (FTA_SSL_EXT.1)

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

5.2.7.4 Default TOE Access Banners (FTA_TAB.1)

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

5.2.8 Trusted path/channels (FTP)

5.2.8.1 Trusted Channel (FTP_ITC.1)

FTP_ITC.1.1 Refinement: The TSF shall use [***TLS***] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [***authentication server, [TOE Update Server]***] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification 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 [**transmitting audit records to an audit server, obtaining TOE updates, and external authentication functions**].

5.2.8.2 Trusted Path (FTP_TRP.1)

FTP_TRP.1.1 Refinement: The TSF shall use [***TLS/HTTPS, SSH***] **to** provide a communication path between itself and 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 detection of modification of the communicated data.

FTP_TRP.1.2 Refinement: 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 administrative actions.

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are included by reference from the NDPP.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1 Basic functional specification
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.1 Labelling of the TOE
	ALC_CMS.1 TOE CM coverage
ATE: Tests	ATE_IND.1 Independent testing - conformance
AVA: Vulnerability assessment	AVA_VAN.1 Vulnerability survey

Table 3 Assurance Components

Consequently, the assurance activities specified in NDPP apply to the TOE evaluation.

6. TOE Summary Specification

This chapter describes the security functions:

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

6.1 Security audit

The TOE is designed to be able to generate log records for security relevant events as they occur. The events that can cause an audit record to be logged include starting and stopping the audit function, any use of an administrator command via the Web GUI or console, as well as all of the events identified in **Table 2** (which corresponds to the audit events specified in NDPP). Note that the only protocol (i.e., HTTPS, TLS, SSH) failures auditable by the TOE are authentication failures for user-level connections.

The logged audit records identify the date and time, the nature or type of the triggering event, an indication of whether the event succeeded or failed, and the identity of the user responsible for the event. The logged audit records also include event-specific content that includes at least all of the content required in **Table 2**.

The TOE includes an internal log implementation that can be used to store and review audit records locally. The local audit logs are stored on the MX Management Server database. The SOM administrator can pre-select System Event types that will be automatically forwarded from the MX server to the SOM for storage and audit review by the SOM administrator. System Events are also generated by the SOM (e.g. for SOM administrator logins and SOM user account management) and are stored locally on the SOM server. The TOE does not provide any interface for modifying audit records. Audit records can only be purged by an authorized Administrator via the SecureSphere GUI management interface.

By default, the Management Server retains up to 100,000 System Event records, and purges the oldest records when this configurable threshold is exceeded. An authorized Administrator can modify this threshold, or specify a time period for which System Event records must be retained.

The TOE can be configured to send generated audit records to an external Syslog server using TLS. When configured to send audit records to a syslog server, audit records are also written to the external syslog as they are written to the local audit log.

The Security audit function is designed to satisfy the following security functional requirements:

- FAU_GEN.1: The TOE can generate audit records for events including starting and stopping the audit function, administrator commands, and all other events identified in **Table 2**. Furthermore, each audit record identifies the date/time, event type, outcome of the event, responsible subject/user, as well as the additional event-specific content indicated in **Table 2**.
- FAU_GEN.2: The TOE associates each auditable event with the identity of the user that caused the event.
- FAU_STG_EXT.1: The TOE can be configured to export audit records to an external Syslog server and can be configured to use TLS for communication with the Syslog server.

6.2 Cryptographic support

The TOE provides a FIPS mode of operation, which must be enabled in the evaluated configuration. The TOE includes NIST-validated cryptographic algorithms providing supporting cryptographic functions. The TOE uses the RSA Crypto-J version 6.1.3 (FIPS 140-2 cert #2057/2058) and FIPS 140-2 OpenSSL version FIPS 2.0.1 (cert# 1747) cryptomodules for all of the cryptographic functionality. The following functions have been certified in accordance with the identified standards.

Functions	**Standards**	**Certificates (crypto-J / openSSL)**
Asymmetric key generation		
• Domain parameter generation (key size 2048 bits)	NIST Special Publication 800-56A NIST Special Publication 800-56B	RSA #1154/ RSA (Certs. #960, #1086, #1145, #1205, #1237, #1273, #1477, #1535 and #1581);
Encryption/Decryption		
• AES CBC (128 and 256 bits)	FIPS PUB 197 NIST SP 800-38A	AES #2249 AES (Certs. #1884, #2116, #2234, #2342, #2394, #2484, #2824, #2929 and #3090)
Cryptographic signature services		
• RSA Digital Signature Algorithm (rDSA) (modulus 2048)	FIPS PUB 186-2 FIPS PUB 186-3	RSA #1154 RSA (Certs. #960, #1086, #1145, #1205, #1237, #1273, #1477, #1535 and #1581);
Cryptographic hashing		
• SHA-1 (digest sizes 160 bits) • SHA-256 (digest sizes 256 bits) • SHA-512 (digest sizes 512 bits)	FIPS Pub 180-3	SHS #1938 SHS (Certs. #1655, #1840, #1923, #2019, #2056, #2102, #2368, #2465 and #2553)
Keyed-hash message authentication		
• HMAC-SHA-1 (key size 160 bits and digest size 160 bits)	FIPS Pub 198-1 FIPS Pub 180-3	HMAC #1378 HMAC (Certs. #1126, #1288, #1363, #1451, #1485, #1526, #1768, #1856 and #1937)

Random bit generation		
• CTR-DRBG(AES) with one independent software-based noise source of 256 bits of non-determinism	NIST Special Publication 800-90A	DRBG # 273 DRBG (Certs. #157, #229, #264, #292, #316, #342, #485, #540 and #607)

Table 4 Cryptographic Functions

The TOE implements a random number generator for RSA key establishment schemes; and for finite-based key establishment (conformant to NIST SP 800-56A and to NIST SP 800-56B).

The Gateway appliances (including virtual appliances) implement a software-based deterministic random bit generator that complies with NIST SP 800-90, using CTR_DRBG (AES) seeded with 256 bits of entropy. On the X10K, X2510, and X8510 appliances, the entropy source is the RDRAND instruction provided by Intel Ivy Bridge-based processors, which is assumed to provide 0.5 bits of entropy per bit sample. The same entropy source is also used on virtual gateway appliances, which require an Ivy Bridge-based processor on the hosting hardware. On X1010, X2010, X4510, and X6510 appliances, the entropy source is an Infineon SLB96xx Trusted Platform Module (TPM) processor, which is assumed to provide 1 bit of entropy per bit sample (i.e., full entropy).

The Management Server appliances (including the virtual Management Server appliance) implement a software-based deterministic random bit generator that complies with NIST SP 800-90, using HMAC_DRBG seeded with 256 bits of entropy. On M110 and M160 appliances, the entropy source is an Infineon SLB96xx Trusted Platform Module (TPM) processor, which is assumed to provide 1 bit of entropy per bit sample (i.e., full entropy).

The TOE is designed to zeroize secret and private keys when they are no longer required by the TOE. The TOE uses the RSA Crypto-J and FIPS 140-2 OpenSSL cryptomodule functions for the zeroization of all ephemeral sensitive data. The TOE itself zeroizes the following secret and private keys when they are no longer required by the TOE.

#	Key/ CSP Name	Generation/ Algorithm	Description	Storage	Zeroization
CSP1	RSA private keys	RSA(2048 bits)	Identity certificates for the security appliance itself and also used in TLS negotiations.	Key Store on Disk RAM (plain text)	Overwriting file with a string of the original length of the sensitive data into the same location in the file; then delete.
CSP2	CA Certificates	RSA(2048 bits)	Trusted CAs	Trust Store on Disk RAM (plain text)	Overwriting file with a string of the original length of the sensitive data into the same location in the file; then delete.
CSP3	Domain and DB Credentials, Proxy Credentials	Secret (plain text)	Usernames and passwords for protected machines and their databases	Database (RSA-2048) RAM (plain text) Gateway Disk (RSA-2048) Transit (TLS with secrets generated by RSA-2048 private keys)	Overwrite with a fixed string of zeroes; then delete.
CSP5	SIEM Credentials	Secret	Used for sending syslog messages	See CSP3	See CSP3
CSP6	External Machines Certificates	Various, can be shared secrets of any kind	Public Keys of machines for integration authentication	See CSP3	See CSP3
CSP7	Machine	Secret	admin login	Linux Hash saved on	Overwriting file

#	Key/ CSP Name	Generation/ Algorithm	Description	Storage	Zeroization
	credentials			disk	with a string of the original length of the sensitive data into the same location in the file; then delete.
CSP8	Database Credentials	RSA(2048)	SecureSphere Database Access	Disk (RSA-2048) RAM (plain text)	Overwriting file with a string of the original length of the sensitive data into the same location in the file; then delete.

Table 5 Key/CSP Zeroization Summary

Administrator passwords for locally defined users are stored as SHA-512 hash in a database located on the MX Management Server.

When the TOE downloads a new software release package these are hashed using SHA-256. Administrators verify the hash manually before installing the downloaded updates.

The TOE uses RSA B-SAFE and OpenSSL FIPS Object Module algorithms: AES (CBC) 128, 256 bit ciphers, in conjunction with HMAC-SHA-1 and RSA signature verification with 2048 bit key sizes. The implementations are in accordance with FIPS PUB 186-3, “Digital Signature Standard”, FIPS Pub 180-3, 'Secure Hash Standard', and FIPS Pub 198-1, 'The Keyed-Hash Message Authentication Code'.

The TOE supports SSHv2 with AES (CBC) 128 or 256 bit ciphers, in conjunction with HMAC-SHA-1. The TOEs SSH implementation complies with RFCs 4251, 4252, 4253, and 4254, 5656; supports RSA public key algorithm; and supports diffie-hellman-group14-sha1 key exchange method. Both public-key and password based authentication can be configured. The TOE manages a packet counter for each SSH session such that packets larger than the 256K bytes packet limit are dropped.

The TOE’s HTTPS protocol complies with RFC 2818 and is implemented using TLS 1.0 (RFC 2246), TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)] supporting the following ciphersuite: TLS_RSA_WITH_AES_128_CBC_SHA .

The Cryptographic support function is designed to satisfy the following security functional requirements:

- FCS_CKM.1: See table above.
- FCS_CKM_EXT.4: See table above.
- FCS_COP.1(1): See table above.
- FCS_COP.1(2): See table above.
- FCS_COP.1(3): See table above.
- FCS_COP.1(4): See table above.
- FCS_HTTPS_EXT.1: The TOE supports HTTPS web-based secure administrator sessions.
- FCS_RBG_EXT.1: See table above.
- FCS_SSH_EXT.1: The TOE supports SSH-based secure administrator sessions.
- FCS_TLS_EXT.1: The TOE supports HTTP over TLS web-based secure administrator sessions.

6.3 User data protection

The TOE is designed to ensure that it does not inadvertently reuse data found in network traffic. Previous information is made unavailable to next user during construction of the new packet. The TOE does not include any

padding or gaps within packets, and therefore all information that the next user receives is the information intended for it. The next user will only receive the information written in the packet for the new user.

The User data protection function is designed to satisfy the following security functional requirements:

- FDP_RIP.2: The TOE always overwrites resources when allocated for use in objects.

6.4 Identification and authentication

The TOE is designed to require users to be identified and authenticated before they can access any of the TOE functions. Administrators manage the TOE remotely using SSHv2 to access the CLI; or using a web-based GUI accessed via HTTPS. Administrators can also connect to the TOE locally using a directly connected console. However the TOE is not intended to be managed locally. For each session, the user is required to log in prior to successfully establishing a session through which TOE functions can be exercised. After a user enters their credentials but prior to a session being established, the TOE displays an advisory warning banner.

In order to log in, the user must provide an identity and also authentication data that matches the provided identity. Local and remote access to the CLI and GUI all support password authentication. SSHv2 also supports public key authentication methods. Users can be defined locally within the TOE with a user identity, password, and user role. Alternately, users can be defined within the TOE but have their authentication data (password) defined in an external LDAP server configured to be used by the TOE. Locally defined users are authenticated directly by the TOE, while remotely defined users are authenticated by the external server and the result is enforced by the TOE. In either case, any resulting session is dependent upon successful authentication and established sessions are associated with the role(s) (see Section 6.5) assigned to the user.

When logging in, the TOE will not echo passwords so that passwords are not inadvertently displayed to the user and any other users that might be able to view the login display.

Note also that should a user have their session terminated (e.g., due to inactivity), they are required to successfully authenticate, by reentering their identity and authentication data, in order to establish a new session.

Password requirements can be configured for local user accounts and are enforced for both SSH and TLS connections. Passwords can be composed of upper and lower case letters, numbers and special characters: "!@#$%^&*()< > ? . _ + = - [ ] { } \ | : ; , / ` ~".

Also, new passwords have to satisfy a configurable minimum password length. The administrator can specify a minimum password length of 15 characters with an upper bound of 255 characters.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_PMG_EXT.1: The TOE implements a rich set of password composition constraints as described above.
- FIA_UAU.7: The TOE does not echo passwords as they are entered.
- FIA_UAU_EXT.2: The TOE provides a local password-based authentication mechanism and can be configured to use an external LDAP authentication server.
- FIA_UIA_EXT.1: The TOE only displays the warning banner prior to a user being identified and authenticated. Note that a user enters their credentials, the banner is displayed, and then the session is established.

6.5 Security management

The TOE provides a GUI to access the wide range of security management functions. Security management commands are limited to administrators and are available only after they have provided acceptable user identification and authentication data to the TOE. The TOE controls user access to the TOE and resources based on user role. Users are given permission to access a set of commands and resources based on their user role.

The TOE also provides the ability to manage the TOE locally via direct serial console connection. The direct serial console is used mainly for initial configuration and then the TOE is designed to be managed and monitored using the Web GUI from a remote HTTPS/TLS client or via SSHv2 remote connection to the CLI. The TOE provides the Administrator role which corresponds to the NDPP Security Administrator.

The TOE includes pre-defined user roles, of which only the user role: Administrator is considered a 'Security Administrator' as defined in the NDPP. Users with the Administrator role are capable of managing the security functions of the TOE. The TOE includes other pre-defined roles that represent logical subsets of the Administrator role. Only users with the Administrator role can manage all aspects of the TOE.

The TOE includes the following security management functions: functions to configure the TOE banners; to enable/disable FIPS mode; configure ciphersuites; to configure TLS secure connections for external user LDAP authentication; to configure user session timeout, and to manage and verify updates of the TOE software. The security management functions required by the PP are accessible via the GUI, except configuration of the advisory banner and session timeout, which are done via direct serial connection during initial configuration or remotely via SSHv2.

The Security management function is designed to satisfy the following security functional requirements:

- FMT_MTD.1: The TOE restricts the access to manage TSF data that can affect the security functions of the TOE to Administrators.
- FMT_SMF.1: The TOE includes the functions necessary to configure TOE banners, to enable/disable FIPS mode, to configure TLS secure connections for external user LDAP authentication, user session timeout, and to manage and verify updates of the TOE software and firmware.
- FMT_SMR.2: The TOE includes predefined roles of which only the Administrator role has access to all security management functions of the TOE, which corresponds to the required 'Security Administrator'.

6.6 Protection of the TSF

While the administrative interface is function rich, the TOE is designed specifically to prevent access to locally-stored cryptographically protected passwords and does not disclose any keys stored in the TOE. The TOE protects user passwords as follows. User and administrator passwords are hashed with SHA-512. Keys are encrypted using the default Java algorithm, which is PBEWithMD5AndTripleDES. The TOE does not offer any functions that will disclose to any users a stored cryptographic key or password. See Section 2 for more information about stored keys and passwords.

The TOE protects TSF data from disclosure and detects its modification when it is transmitted between separate parts of the TOE through the use of TLS/HTTPS. Communication between TOE components uses HTTP over TLS for TOE configuration, management and monitoring.

The TOE is a hardware appliance or a virtual appliance image installed on a hardware appliance that includes a hardware-based real-time clock. The TOE's embedded OS manages the clock and exposes administrator clock-related functions. The clock is used for audit record time stamps, measuring session activity for termination, and for cryptographic operations based on time/date. The TOE requires an NTP Server in the operational environment in order to synchronize its clock with that of the external time server.

The TOE includes FIPS validated OpenSSL and RSA BSAFE and as such runs the same suite of self tests included with these packages during initial start-up (on power on) to demonstrate the correct operation of the cryptographic procedures of the TOE. Please see the publically available FIPS policies for OpenSSL and RSA BSAFE for more details.

MX and Gateway software updates can be downloaded from the TOE Update Server using a secure TLS connection. A software update installation package is hashed by Imperva using SHA-256 hash. Administrators with proper credentials are able to download the update package and can verify the hash prior to installing the update. If the verification fails, it is assumed that the download was corrupted and the administrator is instructed not to install the

package. There are no automatic installation procedures available. The TOE provides functions to query and upgrade the TOE versions.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_APW_EXT.1: The TOE does not offer any functions that will disclose to any user a plain text password. Passwords and keys are stored in cryptographically protected form within the TOE.
- FPT_ITT.1: The TOE protects TSF data from disclosure and detect its modification when it is transmitted between separate parts of the TOE through the use of TLS/HTTPS.
- FPT_SKP_EXT.1: The TOE does not offer any functions that will disclose to any users a stored cryptographic key.
- FPT_STM.1: The TOE includes its own hardware clock.
- FPT_TST_EXT.1: The TOE runs the self-tests included in the FIPS validated OpenSSL and RSA BSAFE packages to ensure that the TOE is functioning properly.
- FPT_TUD_EXT.1: The TOE provides functions to query and upgrade the TOE versions. SHA-256 hash is used to ensure the integrity of each upgrade prior to performing the upgrade.

6.7 TOE access

The TOE can be configured by an administrator to display advisory banners prior to allowing an administrator to establish an administrative user session. The banner will be displayed when accessing the TOE locally, via SSHv2 or via the GUI.

The TOE can be configured by an administrator to set an interactive session timeout value (any integer value greater than zero in minutes; -1 specifies a value of infinity). The default timeout is fifteen minutes for local and remote sessions for both the CLI and GUI. A remote session that is inactive (i.e., no commands issuing from the remote client) for the defined timeout value will be terminated. A local session that is similarly inactive for the defined timeout period will be terminated. Should a user have their session terminated (e.g., due to inactivity), they are required to successfully authenticate, by reentering their identity and authentication data, in order to establish a new session.

The TOE provides functions to logout (or terminate) both local and remote user sessions as directed by the user.

The TOE access function is designed to satisfy the following security functional requirements:

- FTA_SSL.3: The TOE terminates remote sessions that have been inactive for an administrator-configured period of time.
- FTA_SSL.4: The TOE provides the function to logout (or terminate) both local and remote user sessions as directed by the user.
- FTA_SSL_EXT.1: The TOE terminates local sessions that have been inactive for an administrator-configured period of time.
- FTA_TAB.1: The TOE can be configured to display administrator-defined advisory banners before establishing an administrative user session.

6.8 Trusted path/channels

The TOE can be configured to export audit records to an external Syslog server. The TOE uses TLS to protect communications between itself and components in the operational environment including the TOE Update Server, Syslog and authentication servers (LDAP).

To support secure remote administration, the TOE includes implementations of TLS and SSH. An authorized administrator can establish secure remote connections with the TOE using HTTP over TLS or SSHv2. To successfully establish an interactive administrative session, the administrator must be able to provide acceptable user

credentials (e.g., user id and password), after which they will be able to access the Administrative functions. The TOE also supports public key-based authentication for users connecting to the CLI via remote SSHv2.

The secure protocols are supported by NIST-validated cryptographic mechanisms included in the TOE implementation.

The Trusted path/channels function is designed to satisfy the following security functional requirements:

- FTP_ITC.1: The TOE can be configured to use TLS to ensure that any authentication operations, and exported audit records, are sent only to the configured Syslog or authentication servers so they are not subject to inappropriate disclosure or modification. TLS is also used to ensure TOE updates are transmitted securely.
- FTP_TRP.1: The TOE provides SSH and TLS/HTTPS to support secure remote administration. Administrators can initiate a remote session that is secured (from disclosure and modification) using NIST-validated cryptographic operations, and all remote security management functions requires the use of this secure channel.

7. Protection Profile Claims

The ST conforms to the *Protection Profile for Network Devices, Version 1.1, 8 June 2012* (NDPP), as amended by Errata #3 – with the optional FPT_ITT, HTTPS and TLS requirements.

As explained in Section 3, Security Problem Definition, the Security Problem Definition of the NDPP has been included by reference into this ST.

As explained in Section 4, Security Objectives, the Security Objectives of the NDPP have been included by reference into this ST.

The following table identifies all the Security Functional Requirements (SFRs) in this ST. Each SFR is reproduced from the NDPP and operations completed as appropriate.

Requirement Class	Requirement Component	Source
FAU: Security audit	FAU_GEN.1: Audit Data Generation	NDPP
	FAU_GEN.2: User identity association	NDPP
	FAU_STG_EXT.1: External Audit Trail Storage	NDPP
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation (for asymmetric keys)	NDPP
	FCS_CKM_EXT.4: Cryptographic Key Zeroization	NDPP
	FCS_COP.1(1): Cryptographic Operation (for data encryption/decryption)	NDPP
	FCS_COP.1(2): Cryptographic Operation (for cryptographic signature)	NDPP
	FCS_COP.1(3): Cryptographic Operation (for cryptographic hashing)	NDPP
	FCS_COP.1(4): Cryptographic Operation (for keyed-hash message authentication)	NDPP
	FCS_HTTPS_EXT.1: Extended: HTTP Security (HTTPS)	NDPP
	FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)	NDPP
	FCS_SSH_EXT.1: Explicit: SSH	NDPP
	FCS_TLS_EXT.1: Extended: Transport Layer Security (TLS)	NDPP
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection	NDPP
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management	NDPP
	FIA_UAU.7: Protected Authentication Feedback	NDPP
	FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism	NDPP
	FIA_UIA_EXT.1: User Identification and Authentication	NDPP
	FMT_MTD.1: Management of TSF Data (for general TSF data)	NDPP
	FMT_SMF.1: Specification of Management Functions	NDPP
	FMT_SMR.2: Restrictions on Security Roles	NDPP
FPT: Protection of the TSF	FPT_SKP_EXT.1: Extended: Protection of TSF Data (for reading of all symmetric keys)	NDPP
	FPT_APW_EXT.1: Extended: Protection of Administrator Passwords	NDPP
	FPT_ITT.1: Basic Internal TSF Data Transfer Protection	NDPP
	FPT_STM.1: Reliable Time Stamps	NDPP
	FPT_TST_EXT.1: TSF Testing	NDPP
	FPT_TUD_EXT.1: Extended: Trusted Update	NDPP
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination	NDPP
	FTA_SSL.4: User-initiated Termination	NDPP
	FTA_SSL_EXT.1: TSF-initiated Session Locking	NDPP
	FTA_TAB.1: Default TOE Access Banners	NDPP
FTP: Trusted path/channels	FTP_ITC.1: Trusted Channel	NDPP
	FTP_TRP.1: Trusted Path	NDPP

Table 6 SFR Protection Profile Sources

8. Rationale

This security target includes by reference the NDPP Security Problem Definition, Security Objectives, and Security Assurance Requirements. The security target makes no additions to the NDPP assumptions. NDPP security functional requirements have been reproduced with the Protection Profile operations completed. Operations on the security requirements follow NDPP application notes and assurance activities. Consequently, NDPP rationale applies but is incomplete. The TOE Summary Specification rationale below serves to complete the rationale required for the security target.

8.1 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The security functions work together to satisfy all of the security functional requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 7 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

	Security audit	Cryptographic support	User data protection	Identification and authentication	Security management	Protection of the TSF	TOE access	Trusted path/channels
FAU_GEN.1	X							
FAU_GEN.2	X							
FAU_STG_EXT.1	X							
FCS_CKM.1		X						
FCS_CKM_EXT.4		X						
FCS_COP.1(1)		X						
FCS_COP.1(2)		X						
FCS_COP.1(3)		X						
FCS_COP.1(4)		X						
FCS_HTTPS_EXT.1		X						
FCS_RBG_EXT.1		X						
FCS_SSH_EXT.1		X						
FCS_TLS_EXT.1		X						
FDP_RIP.2			X					
FIA_PMG_EXT.1				X				
FIA_UAU.7				X				
FIA_UAU_EXT.2				X				
FIA_UIA_EXT.1				X				
FMT_MTD.1					X			
FMT_SMF.1					X			
FMT_SMR.2					X			
FPT_APW_EXT.1						X		
FPT_ITT.1						X		
FPT_SKP_EXT.1						X		
FPT_STM.1						X		
FPT_TST_EXT.1						X		
FPT_TUD_EXT.1						X		
FTA_SSL.3							X	
FTA_SSL.4							X	
FTA_SSL_EXT.1							X	
FTA_TAB.1							X	
FTP_ITC.1								X
FTP_TRP.1								X

Table 7 Security Functions vs. Requirements Mapping