Cisco Identity Services Engine (ISE) Security Target

Revision 1.0

January 2014

Table of Contents

List of Tables

DOCUMENT INTRODUCTION

Prepared By:
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Cisco Identity Services Engine (ISE) v1.2. This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements. Administrators of the TOE will be referred to as administrators, Authorized Administrators, TOE administrators, semi-privileged, privileged administrators, and security administrators in this document.

1 SECURITY TARGET INTRODUCTION

The Security Target contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- IT Security Requirements [Section 5]
- TOE Summary Specification [Section 6]
- Rationale [Section 7]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 3, Chapter 4.

1.1 ST and TOE Reference

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

Table 1: ST and TOE Identification

ST Title	Cisco Identity Services Engine (ISE) v1.2 Security Target
ST Version	1.0
Publication Date	January 2014
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Identity Services Engine (ISE) v1.2 with patch 5
TOE Models	ISE 3400 series: 3415, 3495, and Virtual Machine
TOE Software Version	ISE v1.2, with patch 5 (1.2.0.899-5), running on Cisco Application Deployment Engine (ADE) Release 2.0 operating system (ADE-OS)
ST Evaluation Status	In Evaluation
Keywords	AAA, Audit, Authentication, Encryption, NAC, Profiling, Network Device

1.2 Acronyms and Abbreviations

The following acronyms and abbreviations are used in this Security Target:

Table 2: Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
ACL	Access Control List
AES	Advanced Encryption Standard
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CLI	Command Line Interface

Acronyms / Abbreviations	Definition
CM	Configuration Management
DH	Diffie-Hellman
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standard
HMAC	Hashed Message Authentication Code
HTTPS	Hyper-Text Transport Protocol Secure
IKE	Internet Key Exchange
IP	Internet Protocol
IPSec	IP Security
IT	Information Technology
NAC	Network Access Control
NTP	Network Time Protocol
OS	Operating System
OSP	Organizational Security Policies
PP	Protection Profile
pp_nd_v1.1	U.S. Government Protection Profile, Security Requirements for Network Devices (NDPP)
PRNG	Pseudo Random Number Generator
RNG	Random Number Generator
SGA	Security Group Access
SGACL	Security Group Access Control List
SGT	Security Group tags
SSHv2	Secure Shell (version 2)
ST	Security Target
TCP	Transport Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TDES	Triple Data Encryption Standard
TLS	Transport Layer Security
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
VPN	Virtual Private Network
WLC	Wireless LAN Controller

1.3 Terminology

The following terms are used in this Security Target:

Table 3: Acronyms

Term	Definition
Endpoints	An endpoint role is a set of permissions that determine the tasks that the device can perform or services that can be accessed on the Cisco ISE network. Endpoints can be users, personal computers, laptops, IP phones,

Term	Definition
	printers, or any other device supported on the ISE network
Inline Posture node	A gate-keeping node that is positioned behind the network access devices. Inline Posture enforces access policies after a user has been authenticated and granted access. There can be or two maximum nodes instances running as Inline Posture node. The Inline Posture node cannot assume any other persona, due to its specialized nature.
Group member	A group member role is a set of permissions that determine the tasks a user (by virtue of being a member of a group) can perform or the services that can be accessed on the ISE network.
Node	A node is an individual instance of ISE. There are two types of nodes, an ISE node that can take on one of three Personas and the Inline Posture node.
Node type	The TOE can be one of two types; an ISE node or an Inline posture node. The node type and persona determine the type of functionality provided by the node.
Persona	The persona of a node determines that service provided by a node. The TOE can be configure as any of the following personas: • Administration – allows the user to perform all of the administrative operations on the TOE. All of the authentication, authorization, auditing, and so on are managed. There can be one or two maximum node instances running the Administration persona and can take any one of the following roles; standalone, primary, or secondary. • Policy Service – provides network access, posture, guest services, client provisioning, and profiling services. This persona evaluates the policies and makes all of the decisions. There can be one or more instance of a node configured as a Policy Service. • Monitoring – functions as the log collector and stores log messages from all of the Administration and Policy Service personas. There can be one or two node instances running the Monitoring persona.
Role	The role identity determines of the TOE is a standalone, primary, or secondary node.
Service	A service is a specific feature that a persona provides, such as network access, posture, security group access, and monitoring
User	A user role is a set of permissions that determine what tasks a user can perform or what services can be accessed on the ISE network. The user identity includes username, password, and group association.

1.4 TOE Overview

The TOE is an identity and access control platform that enables organizations to enforce compliance and security within the network infrastructure. The TOE includes two hardware options: Cisco Identity Services Engine Appliance 3415 (Small) and Cisco Identity Services Engine Appliance 3495 (Large), and the Cisco Identity Services Engine Virtual Machine (ISE VM) on dedicated hardware.

1.4.1 TOE Product Type

The Cisco Identity Services Engine (ISE) v1.2 is a network device identity, authentication, and access control policy platform that enables enterprises to enforce compliance, enhance infrastructure security, and streamline service operations. ISE allows enterprises to gather real-time contextual information from networks, users, and devices. The administrator can then use that information to make proactive governance decisions by tying identity to various network elements including access switches, wireless LAN controllers (WLCs), virtual private network (VPN) gateways, and data center switches.

1.4.2 Supported Non-TOE Hardware/ Software/ Firmware

The TOE supports (in some cases optionally) the following hardware, software, and firmware in its environment:

Table 4: IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Administrative Console	Yes	This console provides the connection to the ISE appliance for administration and management. The console can connect directly to ISE or over the network via a browser or SSHv2 connection. The TOE supports the following browsers: Firefox 18.x Firefox 15.x Firefox 14.x Firefox 9.x Firefox 8.x Firefox 5.x Internet Explorer 8.x Internet Explorer 9.x (IE8 Compatible Mode)
NTP Server(s)	No	The TOE supports communications with up to three NTP servers. Connection with an NTP server is to maintain an accurate time and synchronize time across different time zones. This procedure ensures that the logs provide a reliable timestamp. By having multiple NTP servers configured the time to converge when one of the NTP servers goes down is reduced. Because of the

Component	Required	Usage/Purpose Description for TOE performance
		importance of reliable time in a security product, it is advised that multiple NTP servers are configured for ISE.
Remote Authentication Store	No	The TOE supports local authentication or authentication via a remote authentication store, including LDAP and Active Directory.
Syslog Target	Yes	The TOE must offload syslogs to an external entity, which can be another iteration of ISE or a syslog server that supports TLS-protected transfer.

1.5 TOE DESCRIPTION

This section provides an overview of the Cisco Identity Services Engine (ISE) v1.2 Target of Evaluation (TOE). ISE is a consolidated policy-based access control system that combines authentication, authorization, accounting (AAA), posture, profiler, and guest management in one appliance.

There are two types of license of ISE, Base and Advanced. For the purposes of this evaluation, all claimed functionality is included in both license types. The Base license includes AAA services, guest lifecycle management, compliance reporting and end-to-end monitoring and troubleshooting. The Advanced license expands on the Base license and enables policy decision based on user and device compliance. The Advanced license features include device profiling, posture services, and security group access enforcement capabilities.

There are seven policy models that can be configured to determine how network access is granted to the users requesting access to the network resources. The policies are a set of conditions that must be met in order for access to be granted. The policy models are as follows:

- Authentication Policy – defines the protocols that are used to communicate with the network devices, the identity sources used for authentication, and the failover options.
- Authorization Policy – defines the authorization policies and profiles for specific users and groups of users that have access to the network resources. The policies associate rules with specific user and group identities to create the corresponding profiles. Whenever these rules match the configured attributes, the corresponding authorization profile that grants permission is returned by the policy, network access is authorized accordingly.
- Profiler Policy - provides the unique functionality in discovering, locating, and determining the capabilities of all the attached endpoints (a.k.a identities) on the

network. The profiler collects an attribute or a set of attributes of all the endpoints on the network and classifies them according to their profiles.

- Client Provisioning Policy – like the Profiler policy, the TOE looks at various elements when classifying the type of login session through which users access the internal network, including:
    - Client machine operating system and version
    - Client machine browser type and version
    - Group to which the user belongs
    - Condition evaluation results (based on applied dictionary attributes)

  After the TOE classifies a client machine, it uses client provisioning resource policies to ensure that the client machine is set up with an appropriate agent version, up-to-date compliance modules for antivirus and antispyware vendor support, and correct agent customization packages and profiles, if necessary.
- Posture Policy - allows the administrator to check the state (posture) for all the endpoints that are connecting to the network with the corporate security policies for compliance before clients are granted access to protected areas of the network.
- Guest Management – allows guest (visitors, contractors, consultants, or customers) to perform an HTTP or HTTPS login to access a network whether that network is a corporate intranet or the public Internet. The ISE Guest service allows any user with privileges (sponsor) to create temporary guest accounts and to sponsor guests. When a guest user first attaches to the local network, either through a wireless or wired connection, the user is placed in a segregated network with limited access. The ISE Guest service supports default and customizable guest login portals. The entire process, from user account creation to guest network access, is stored for audit and reporting purposes. It is noted that the guest account is only active for the time specified when the account is created.
- Security Group Access Policy - establishes clouds of trusted network devices to build secure networks. Each device in the ISE SGA cloud is authenticated by its neighbors (peers). Communication between the devices in the SGA cloud is secured with a combination of encryption, message integrity checks, and data-path replay protection mechanisms. The SGA solution uses the device and user identity information that it obtains during authentication to classify, or color, the packets as they enter the network. This packet classification is maintained by tagging packets when they enter the SGA network so that they can be properly identified for the purpose of applying security and other policy criteria along the data path. The tag, also called the security group tag (SGT), allows ISE to enforce access control policies by enabling the endpoint device to act upon the SGT to filter traffic.

1.6 TOE Evaluated Configuration

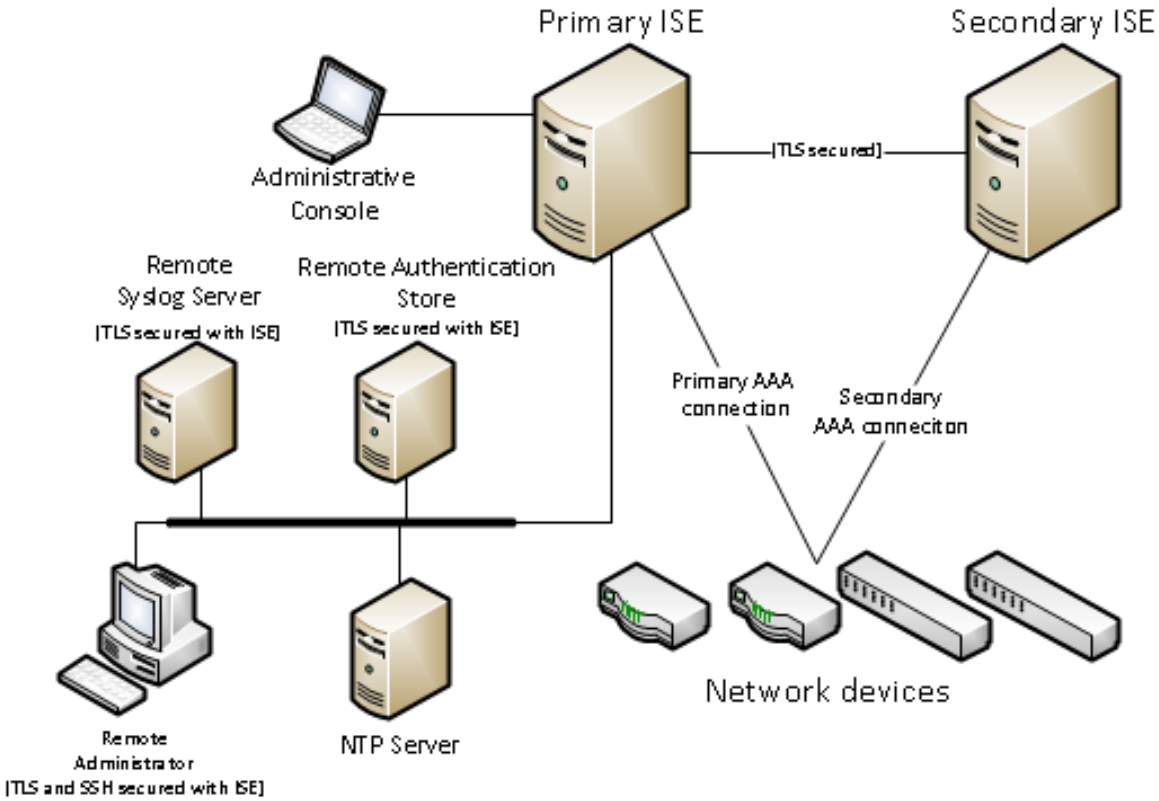
The ISE architecture supports both stand-alone and distributed deployments. In a distributed configuration, one machine assumes the primary role and another "backup" machine assumes the secondary role.

The administrator can deploy ISE nodes with one or more of the Administration, Monitoring, and Policy Service personas, each one performing a different vital part in the overall network policy management topology. Installing ISE with an Administration persona allows the administrator to configure and manage the network from a centralized portal. The administrator can also choose to deploy the ISE platform as an Inline Posture node to perform policy enforcement.

The TOE architecture includes the following components:

- Nodes and persona types
  - ISE node—Administration, Policy Service, Monitoring
  - Inline Posture node—Gatekeeping and access policy enforcer
- Network resources
- Endpoints

Figure 1: Typical TOE Deployment



The evaluated configuration will include one or more ISE instances in a network. A typical deployment will include network devices utilizing the ISE authentication, authorization and accounting (AAA) features, remote administrator, local administrative console, a remote authentication store, and an NTP server. Both the remote administrator and local administrator console capabilities must be supported.

1.7 Physical Scope of the TOE

The Cisco ISE software runs on the Cisco Application Deployment Engine (ADE) Release 2.0 operating system (ADE-OS). The Cisco ADE-OS and Cisco ISE software run on a Cisco ISE 3400 Series appliance or on a dedicated hardware platform with a VMWare hypervisor; the hardware specifications must meet the requirements defined in Table 5 below. All models include the same security functionality.

Table 5: TOE Models

Hardware Model	**Cisco Identity Services Engine Appliance 3415 (Small)**	**Cisco Identity Services Engine Appliance 3495 (Large)**	**Cisco Identity Services Engine Virtual Machine (on dedicated hardware[1]) Required minimum system specifications[2]**
Processor	Cisco UCS C220M3, Single Intel Xeon E5-2609 4 core processor	Cisco UCS C220M3, Dual Intel Xeon E5-2609 4 core processor (8 cores total)	Single Quad-Core; 2.13 GHz or faster
Memory	16 GB	32 GB	4 GB
Hard disk	1x600Gb disk	2x600Gb disk	100 to 600 GB of disk storage (size depends on deployment and tasks) with SCSI controller
RAID	Yes (Software RAID level 0 (single drive striped))	Yes (RAID 1)	N/A
Expansion slots	- Two PCIe slots (on a riser card) ■One full-height profile, half-length slot with x24 connector and x16 lane ■One half-height	- Two PCIe slots (on a riser card) ■One full-height profile, half-length slot with x24 connector and x16 lane ■One half-height	N/A

Hardware Model	Cisco Identity Services Engine Appliance 3415 (Small)	Cisco Identity Services Engine Appliance 3495 (Large)	Cisco Identity Services Engine Virtual Machine (on dedicated hardware[1]) Required minimum system specifications[2]
	profile, half-length slot with x16 connector and x8 lane	profile, half-length slot with x16 connector and x8 lane	
NIC Ports	• One 1-GB Ethernet port (GigE0) for TOE management and network device governance • Three 1-GB Ethernet ports (GigE1, GigE2, GigE3) for network device governance	• One 1-GB Ethernet port (GigE0) for TOE management and network device governance • Three 1-GB Ethernet ports (GigE1, GigE2, GigE3) for network device governance	1-GB Ethernet port required for TOE management and network device governance (two or more NICs are recommended)
Serial/VGA ports	2	2	N/A
USB 2.0 ports	2	2	N/A
Video ports	1	1	N/A
External SCSI ports	None	None	N/A
Hypervisor	None	None	• VMware ESX 4.x • VMware ESXi 4.x; or • VMware ESXi 5.x

1. ISE Virtual Machine's hardware must be dedicated. Likewise, the virtual resources will be dedicated to a single virtual machine running the ISE software.
2. It is recommended that the system specification be comparable with the 3415 or 3495 models in a production environment. This table lists the minimum system specifications for the ISE Virtual Machine to operate as validated by this evaluation.

1.8 Logical Scope of the TOE

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

1. Security audit
2. Cryptographic support
3. User data protection
4. Identification and authentication
5. Secure Management
6. Protection of the TSF

7. TOE access
8. Trusted path/channels

These features are described in more detail in the subsections below.

1.8.1 Security Audit

The TOE's Audit security function supports audit record generation and review. The TOE provides date and time information that is used in audit timestamps. The events generated by the TOE include indication of the logging starting and stopping, cryptographic operations, attempts to log onto the TOE, all commands/ web-based actions executed by the authorized administrative user, and other system events.

The TOE can be configured to send syslog events to other devices, including other iterations of ISE, using a TLS protected collection method. Logs are classified into various predefined categories. The TOE also provides the capability for the administrator to customize the logging output by editing the categories with respect to their targets, severity level, etc. The logging categories help describe the content of the messages that they contain. Access to the logs is restricted only to the authorized administrator, who has no access to edit them, only to copy or delete (clear) them.

The logs can be viewed by using the Operations -> Reports page on the ISE administration interface, then select the log from the left side and individual record (message). The log record includes the category name, the message class, the message code (type of event), the message text (including a date/time stamp, subject (user) associated with the event, outcome of the event, etc) and the severity level associated with the message.

1.8.2 Cryptographic Support

The TOE provides cryptography support for secure communications and protection of information. The TOE relies on FIPS PUB 140-2 validation for testing of cryptographic functions, including self-tests and key zeroization. ISE uses Cisco Common Cryptographic Module (C3M) (FIPS 140-2 Cert#1643) and Cisco Secure Access Control Server (ACS) and FIPS module Network Services (NSS) (FIPS 140-2 Cert#1497). The cryptographic services provided by the TOE include: symmetric encryption and decryption using AES; digital signature using RSA; cryptographic hashing using SHA1 (and other sizes); and keyed-hash message authentication using HMAC-SHA (multiple key sizes). The TOE supports SSH and TLS/HTTPS secure protocols.

1.8.3 User Data Protection

The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Packets that are not the required length use zeros, fixed data based on the amount of padding, or random data, for padding. Residual data is never transmitted from the TOE.

1.8.4 Identification and Authentication

All users wanting to use TOE services are identified and authenticated prior to being allowed access to any of the services. Once a user attempts to access the management functionality of the TOE, the TOE prompts the user for a user name and password. The identification and authentication credentials are confirmed against a local user database or an optional remote authentication store (part of the IT Environment). Other authentication options include public key authentication. For remote password-based authentication to the administration application, an Active Directory identity source (remote authentication store) is required in order to perform the association of the credentials to an ISE Role Based Access Control role. For the SSH public key authentication method, the public keys configured by the EXEC CLI command "crypto key import" command will be used for signature verification. The user information is from the local user database. In all cases only after the administrative user presents the correct identification and authentication credentials will access to the TOE functionality be granted.

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

1.8.5 Security Management

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

- Enable, disable, determine and modify the behavior of the audit trail management
- Configure the cryptographic services
- Update the TOE and verify the updates via a hash comparison
- Enable, disable, determine and modify the behavior of communication of authorized external IT entities with the TOE
- Query, modify, delete, and assign the user attributes
- Specify the time limits of session inactivity

All of these management functions are restricted to the authorized administrator of the TOE, which covers all administrator roles (see table for FMT_SMR.2 in Section 6.1). The Authorized Administrators of the TOE are individuals who manage specific type of administrative tasks. The Authorized Administrators are dependent upon the admin role assigned to them, which limits the network access or tasks they can perform (a role-based access approach).

The primary management interface is the HTTPS Cisco ISE user interface. The Cisco

ISE user interface provides an integrated network administration console from which you can manage various identity services. These services include authentication, authorization, posture, guest, profiler, as well as monitoring, troubleshooting, and reporting. All of these services can be managed from a single console window called the Cisco ISE dashboard. The navigation tabs and menus at the top of the window provide point-and-click access to all other administration features. A Command Line Interface (CLI) is also supplied for additional administration functionality. This interface can be used remotely over SSHv2.

1.8.6 Protection of the TSF

The TOE provides secure transmission when TSF data is transmitted between separate parts of the TOE and is able to detect modification of information and/or operations. The TOE also provides protection of TSF data (authentication data and cryptographic keys). In addition, the TOE internally maintains the date and time. This date and time is used as the time stamp that is applied to TOE generated audit records. This time can be set manually, or an NTP server (or servers) can be used to synchronize the date-timestamp. The TOE is also capable of ensuring software updates are from a reliable source. Finally, the TOE performs testing to verify correct operation.

In order for updates to be installed on the TOE, an administrator must use the provided hash value to confirm the integrity of the product.

1.8.7 TOE Access

The TOE can terminate inactive sessions after an authorized administrator configurable time-period. The TOE also allows users to terminate their own interactive session. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session.

The TOE can also display a Security Administrator specified banner on the CLI and the web-based management interface prior to allowing any administrative access to the TOE.

1.8.8 Trusted Path/Channels

The TOE establishes a trusted path between the ISE and the administrative web-based using TLS/HTTPS, and between the ISE and the CLI using SSH. The TOE also establishes a secure connection for sending syslog data to other IT devices using TLS and other external authentication stores using TLS-protected communications.

1.9 Excluded Functionality

The following functional is excluded from the evaluation.

Table 6: Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations.

These services will be disabled by configuration. The exclusion of this functionality does not affect compliance to the US Government, Security Requirements for Network Devices (pp_nd_v1.1), version 1.1.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 4, dated: September 2012.
The TOE and ST are EAL1 Part 3 conformant; PP dependent.
The TOE and ST are CC Part 2 extended.

2.2 Protection Profile Conformance

This ST claims compliance to the following Common Criteria validated Protection Profiles (PP), US Government, Security Requirements for Network Devices (pp_nd_v1.1), version 1.1, dated 8 June 2012 (from here within referred to as NDPP).

This Security target has adapted the Security Problem Definition, Security Objectives, and Security Functional Requirements (SFRs) from the NDPP with the additions listed below.

2.2.1 Protection Profile Additions

None.

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality of a network device as described in the NDPP.

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent the Assumptions, Threats, and Organization Security Policies specified in the NDPP for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Section 3.

2.3.3 Statement of Security Objectives Consistency

The Security Objectives included in the Security Target represent the Security Objectives specified in the NDPP for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Objectives are included in the Security Target Section 4.

2.3.4 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDPP for which conformance is claimed verbatim. All concepts covered by the Protection Profile's Statement of Security Requirements are included in the Security Target Section 5. Additionally, the Security Assurance Requirements included in the Security Target are identical to the Security Assurance Requirements included in the NDPP.

3 SECURITY PROBLEM DEFINITION

This chapter identifies the following:

- Significant assumptions about the TOE's operational environment.
- IT related threats to the organization countered by the TOE.
- Environmental threats requiring controls to provide sufficient protection.
- Organizational security policies for the TOE as appropriate.

This document identifies assumptions as A.assumption with "assumption" specifying a unique name. Threats are identified as T.threat with "threat" specifying a unique name. Organizational Security Policies (OSPs) are identified as P.osp with "osp" specifying a unique name.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 7: TOE Assumptions

Assumption	Assumption Definition
Reproduced from the Security Requirements for NDPP	
A.NO_GENERAL_PURPOSE	It is assumed that 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.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is Enhanced-Basic.

Table 8: Threats

Threat	Threat Definition
Reproduced from the Security Requirements for NDPP	
T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE

Threat	Threat Definition
	executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

3.3 Organizational Security Policies

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

Table 9: Organizational Security Policies

Policy Name	Policy Definition
Reproduced from the Security Requirements for NDPP	
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

4 SECURITY OBJECTIVES

This Chapter identifies the security objectives of the TOE and the IT Environment. The security objectives identify the responsibilities of the TOE and the TOE's IT environment in meeting the security needs.

- This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the TOE

The following table, Security Objectives for the TOE, identifies the security objectives of the TOE. These security objectives reflect the stated intent to counter identified threats and/or comply with any security policies identified. An explanation of the relationship between the objectives and the threats/policies is provided in the rationale section of this document.

Table 10: Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
Reproduced from the Security Requirements for NDPP	
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

4.2 Security Objectives for the Environment

All of the assumptions stated in Section 3.1 are considered to be security objectives for the environment. The following are the NDPP non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements

on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 11: Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
Reproduced from the Security Requirements for NDPP	
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 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from *US Government, Security Requirements for Network Devices (pp_nd_v1.1), version 1.1, dated 8 June 2012* and Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012* and all international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Completed assignments are shown in **bold text in brackets []**, while completed selections are show in ***bold italics in brackets* []**. Where operations were competed in the NDPP itself, the formatting used in the NDPP has been retained.
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3)
- The Extended SFRs are identified by having a label '_EXT' as part of the requirement name for TOE SFRs
- Refinements are shown in **bold text**, with the removed text **~~bold strike-through~~**.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE that are specified in the NDPP. The TOE Security Functional Requirements that appear in the following table are described in more detail in the following subsections.

Table 12: Security Functional Requirements

Functional Component	
Requirement Class	**Requirement Component**
Security Functional Requirements Drawn from Security Requirements for NDPP	
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)

Functional Component	
	FCS_RBG_EXT.1: Cryptographic operation (random bit generation)
	FCS_TLS_EXT.1: TLS
	FCS_HTTPS_EXT.1: HTTPS
	FCS_SSH_EXT.1: SSH
FDP: User data protection	FDP_RIP.2: Full residual information protection
FIA: Identification and authentication	FIA_PMG_EXT.1: Password management
	FIA_UIA_EXT.1: User identification and authentication
	FIA_UAU_EXT.2: Password-based authentication mechanism
	FIA_UAU.7: Protected authentication feedback
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_ITT.1: Basic internal TSF data transfer protection
	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)
	FPT_APW_EXT.1: Protection of Administrator Passwords
	FPT_STM.1: Reliable time stamps
	FPT_TUD_EXT.1: Trusted update
	FPT_TST_EXT.1: TSF testing
FTA: TOE Access	FTA_SSL_EXT.1: TSF-initiated session locking
	FTA_SSL.3: TSF-initiated termination
	FTA_SSL.4: User-initiated termination
	FTA_TAB.1: Default TOE access banners
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel
	FTP_TRP.1: Trusted path

5.2.1 Security Audit (FAU)

5.2.1.1 FAU_GEN.1: Audit data generation

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

a) Start-up 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 13*].

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 13*].

Table 13: 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_EXT.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_RBG_EXT.1	None.	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_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.
FCS_SSH_EXT.1	Failure to establish an SSH Session. Establishment/Termination of an SSH Session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FDP_RIP.2	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 address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FMT_MTD.1	None.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.

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

5.2.1.2 FAU_GEN.2: User identity association

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

5.2.1.3 FAU_STG_EXT.1: External audit trail storage

FAU_STG_EXT.1.1 The TSF shall be able to [***transmit the generated audit data to an external IT entity***] using a trusted channel implementing the [***TLS***] protocol.

5.2.2 Cryptographic Support (FCS)

5.2.2.1 FCS_CKM.1: Cryptographic key generation (for asymmetric keys)

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; or 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 FCS_CKM_EXT.4: Cryptographic key zeroization

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 FCS_COP.1(1): Cryptographic operation (for data encryption/decryption)

FCS_COP.1.1(1) **Refinement:** The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES operating in* [**CBC mode**]] and cryptographic key sizes 128-bits, 256-bits, and [***no other key sizes***] that meets the following:

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

5.2.2.4 FCS_COP.1(2): Cryptographic operation (for cryptographic signature)

FCS_COP.1.1(2) **Refinement:** The TSF shall perform **cryptographic signature services** in accordance with a [***RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater***] that meets the following:

- **FIPS PUB 186-2, "Digital Signature Standard"**

5.2.2.5 FCS_COP.1(3): Cryptographic operation (for cryptographic hashing)

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

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

FCS_COP.1.1(4) **Refinement:** The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm HMAC-[***SHA-1, SHA-256***]**, key size** [**160, 256 bits**]**, and message digest sizes** [***160, 256***] **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 FCS_RBG_EXT.1: Cryptographic operation (random bit generation)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [***FIPS Pub 140-2 Annex C: X9.31 Appendix 2.4 using AES***] seeded by an entropy source that accumulated entropy from [***a software-based noise source; a TSF-hardware-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.

Application Note: The ISE appliance form-factors use both a software-based and TSF-hardware-based noise source, while the ISE VM uses a software-based noise source.

5.2.2.8 FCS_SSH_EXT.1: SSH

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

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.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than **[262,144]** 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).

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 is the only allowed key exchange method used for the SSH protocol.

5.2.2.9 FCS_TLS_EXT.1: TLS

FCS_TLS_EXT.1.1 The TSF shall implement one or more of the following protocols [***TLS1.0 (RFC 2246)***] supporting the following ciphersuites:

Mandatory Ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_DHE_RSA_WITH_AES_128_CBC_SHA

TLS_DHE_RSA_WITH_AES_256_CBC_SHA

Optional Ciphersuites:

[***None***]**.**

5.2.2.10 FCS_HTTPS_EXT.1: HTTPS

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.

5.2.3 User Data Protection (FDP)

5.2.3.1 FDP_RIP.2: Full residual information protection

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 FIA_PMG_EXT.1: Password management

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 be settable by the Security Administrator, and support passwords of 15 characters or greater;*

5.2.4.2 FIA_UIA_EXT.1: User identification and authentication

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

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

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

5.2.4.3 FIA_UAU_EXT.2: Password-based authentication mechanism

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

5.2.4.4 FIA_UAU.7: Protected authentication feedback

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

5.2.5 Security Management (FMT)

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

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

5.2.5.2 FMT_SMF.1: Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to update the TOE, and to verify the updates using [**published hash**] capability prior to installing those updates;*
- *[**Ability to configure the cryptographic functionality**]*

5.2.5.3 FMT_SMR.2: Restrictions on Security roles

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

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

5.2.6.2 FPT_APW_EXT.1 Extended: Protection of Administrator Passwords

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

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

5.2.6.3 FPT_ITT.1: Basic internal TSF data transfer protection (disclosure)

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*]**.

5.2.6.4 FPT_STM.1: Reliable time stamps

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

5.2.6.5 FPT_TUD_EXT.1: Trusted update

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 the 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.

5.2.6.6 FPT_TST_EXT.1: TSF testing

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.

5.2.7 TOE Access (FTA)

5.2.7.1 FTA_SSL_EXT.1: TSF-initiated session locking

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

5.2.7.2 FTA_SSL.3: TSF-initiated termination

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

5.2.7.3 FTA_SSL.4: User-initiated termination

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

5.2.7.4 FTA_TAB.1: Default TOE Access Banners

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/Channel (FTP)

5.2.8.1 FTP_ITC.1: Inter-TSF trusted channel (prevention of disclosure)

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*]** 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 [*all authentication functions,* **[syslogs sent to peer ISE or other devices]**].

5.2.8.2 FTP_TRP.1: Trusted path (prevention of disclosure)

FTP_TRP.1.1 **Refinement:** The TSF shall **use [*SSH, HTTPS*]** provide **a trusted** 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*.

Application Note: The SSH is used to protect the CLI administration, and the HTTPS is used to protect the web-based administration.

5.3 Extended Components Definition

This Security Target includes Security Functional Requirements (SFR) that are not drawn from existing CC Part 2. The Extended SFRs are identified by having a label '_EXT' as part of the

requirement name for TOE SFRs. The structure of the extended SFRs is modelled after the SFRs included in CC Part 2. The structure is as follows:

A. Class – The extended SFRs included in this ST are part of the identified classes of requirements.

B. Family – The extended SFRs included in this ST are part of several SFR families

C. Component – The extended SFRs are not hierarchical to any other components, though they may have identifiers terminating on other than "1". The dependencies for each extended component are identified in the TOE SFR Dependencies section of this ST below.

Extended Requirements Rationale:

FAU_STG_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement to export audit records outside the TOE.

FCS_CKM_EXT.4:

This SFR was taken from NDPP – where it is defined as a requirement for immediate zeroization when keys and CSPs are no longer required.

FCS_HTTPS_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement specific to HTTPS.

FCS_RBG_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement specific to random bit generation.

FCS_SSH_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement specific to SSH.

FCS_TLS_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement specific to TLS.

FIA_PMG_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement for specific password composition and aging constraints. Note that "Security Administrator" has been replaced with "Authorized Administrator".

FIA_UAU_EXT.2:

This SFR was taken from NDPP – where it is defined as a requirement allowing local and other authentication mechanisms.

FIA_UIA_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement combining both identification and authentication requirements.

FPT_SKP_EXT.1:

This SFR was taken from NDPP –where it is defined as a requirement specifically disallowing access to pre-shared keys, symmetric keys, and private keys.

FPT_APW_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement specifically disallowing access to passwords.

FPT_TST_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement for TSF self tests during initialization.

FPT_TUD_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement for secure TOE update capabilities. Note that "Security Administrator" has been replaced with "Authorized Administrator".

FTA_SSL_EXT.1:

This SFR was taken from NDPP – where it is defined as a requirement for behavior after local terminal session inactivity. Note that "Security Administrator" has been replaced with "Authorized Administrator".

5.4 TOE SFR Dependencies Rationale

The following table provides dependency rational for SFRs that were drawn from the NDPP.

Table 14: SFR Dependency Rationale (from NDPP)

SFR	Dependency	Rationale
FAU_GEN.1	FPT_STM.1	Met by FPT_STM.1
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	Met by FAU_GEN. Met by FIA_UIA_EXT.1
FAU_STG_EXT.1	FAU_GEN.1	Met by FAU_GEN.1
FCS_CKM.1	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	Met by FCS_COP.1(2), (3), and (4) Met by FCS_CKM.4
FCS_CKM_EXT.4	FDP_ITC.1 or FDP_ITC.2 or	Met by FCS_CKM.1

SFR	Dependency	Rationale
	FCS_CKM.1	
FCS_COP.1(1)	FDP_ITC.1 or 2 or FCS_CKM.1 FCS_CKM.4	Functional component FCS_COP.1 depends on the following functional components: FCS_CKM.1 Cryptographic key generation, and FCS_CKM.4 Cryptographic key destruction Cryptographic modules must be FIPS PUB 140-2 compliant. If the cryptographic module is indeed compliant with this FIPS PUB, then the dependencies of key generation, key destruction and secure key values will have been satisfied in becoming FIPS PUB 140-2 compliant. For more information, refer to section 4.7 of FIPS PUB 140-2. Met by FCS_CKM.1 and FCS_CKM_EXT.4
FCS_COP.1(2)	FDP_ITC.1 or 2 or FCS_CKM.1 FCS_CKM.4	Met by FCS_CKM.1 and Met by FCS_CKM_EXT.4
FCS_COP.1(3)	FDP_ITC.1 or 2 or FCS_CKM.1 FCS_CKM.4	Met by FCS_CKM.1 and Met by FCS_CKM_EXT.4
FCS_COP.1(4)	FDP_ITC.1 or 2 or FCS_CKM.1 FCS_CKM.4	Met by FCS_CKM.1 and Met by FCS_CKM_EXT.4
FCS_RBG_EXT.1	No dependencies	N/A
FCS_HTTPS_EXT.1	FCS_TLS_EXT.1	Met by FCS_TLS_EXT.1
FCS_TLS_EXT.1	FCS_COP.1	Met by FCS_COP.1
FCS_SSH_EXT.1	FCS_COP.1	Met by FCS_COP.1
FDP_RIP.2	No dependencies	N/A
FIA_PMG_EXT.1	No dependencies	N/A
FIA_UIA_EXT.1	No dependencies	N/A
FIA_UAU_EXT.2	No dependencies	N/A
FIA_UAU.7	FIA_UAU.1	Met by FIA_UIA_EXT.1
FMT_MTD.1	FMT_SMF.1 FMT_SMR.2	Met by FMT_SMF.1 Met by FMT_SMR.2
FMT_SMF.1	No dependencies	N/A
FMT_SMR.2	FIA_UID.1	Met by FIA_UIA_EXT.1
FPT_ITT.1	No dependencies	N/A
FPT_STM.1	No dependencies	N/A
FPT_SKP_EXT.1	No dependencies	N/A
FPT_APW_EXT.1	No dependencies	N/A
FPT_TUD_EXT.1	No dependencies	N/A

SFR	Dependency	Rationale
FPT_TST_EXT.1	No dependencies	N/A
FTA_SSL_EXT.1	No dependencies	N/A
FTA_SSL.3	No dependencies	N/A
FTA_SSL.4	No dependencies	N/A
FTA_TAB.1	No dependencies	N/A
FTP_ITC.1	No dependencies	N/A
FTP_TRP.1	No dependencies	N/A

5.5 Security Assurance Requirements

5.5.1 SAR Requirements

The TOE assurance requirements for this ST are the same as specified in the NDPP, Section 4.3. These constitute an Evaluation Assurance Level (EAL) of EAL1 and are summarized in the table below.

Table 15: Assurance Measures

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: Labeling 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

5.5.2 Security Assurance Requirements Rationale

This Security Target claims conformance to the NDPP which essentially is an EAL1 conformance claim. This target was chosen to ensure that the TOE has a low to moderate level of assurance in enforcing its security functions when instantiated in its intended environment which imposes no restrictions on assumed activity on applicable networks.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. The table below lists the details.

Table 16: Assurance Measures

Component	How requirement will be met
ADV_FSP.1	A description of the TOE security functional interfaces (TSFIs) (SFR-enforcing and SFR-supporting TSFIs) that includes the purpose, method of use, and parameters is documented in the Cisco development evidence. The description of the security functions in the Cisco development evidence shows a correspondence between the interfaces and the security functions defined in the ST.
AGD_OPE.1	The administrative guidance is detailed to provide descriptions of how administrative users of the TOE can securely administer the TOE using those functions and interfaces detailed in the guidance.
AGD_PRE.1	Cisco documents the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	Cisco performs configuration management on configuration items of the TOE. Each configuration is uniquely identified and labeled with its unique reference.
ALC_CMS.1	Cisco uniquely identifies configuration items and each release of the TOE has a unique reference. The Configuration Management documentation contains a configuration item list.
ATE_IND.1	Cisco will help meet the independent testing by providing the TOE to the evaluation facility.
AVA_VAN.1	Cisco will provide the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

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

Table 17: How TOE SFRs are Met

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td colspan="2">Security Functional Requirements Drawn from NDPP</td></tr>
<tr><td>FAU_GEN.1</td><td>The TOE generates and stores audit records locally on the TOE whenever an audited event occurs. The types of events that cause audit records to be generated include, cryptography related events, identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, “Auditable Events Table”). Each of the events is specified in the syslog in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred. Additionally, the startup and shutdown of the audit functionality is audited.
The ability to change logging settings is provided on the Administration > System > Logging > Local Log Settings page.
Following is a sample record: <181>Dec 17 20:17:36 acsview-srv8 CSCOacs_Administrative_Audit 0000003218 1 0 2011-12-17 20:17:36.615 -08:00 0000003936 51001 NOTICE Administrator-Login: Administrator authentication succeeded, ConfigVersionId=3, AdminInterface=GUI, AdminIPAddress=171.69.74.79, AdminSession=058C95A67C4078537C028354A377C11E, AdminName=acsadmin, Each record contains the following fields:
• Category Name—The logging category to which a message belongs (acsview-srv8 in the above record)
• Message Class—The group to which a message belongs (CSCOacs_Administrative_Audit in the above record)
• Message Code—A unique message code identification number associated with a message (0000003218 in the above record)
• Message Text—Name of the message (Administrator-Login in the above record)
• Severity—The severity level associated with a message (NOTICE in the above record)
• Timestamp – The time associated with the message (2011-12-17 20:17:36.615 in the above record)
Note that success or failure is indicated in the individual events, where relevant. The record above indicates that the authentication was successful.
<table>
<tr><th>Auditable Event</th><th>Rationale</th></tr>
<tr><td>Success and failure of encrypted communications</td><td>Attempts of secure encrypted communications/connections (SSH, TLS/HTTPS). The communications include the remote administrator establishing a session and the TOE sending syslog data. The identity of the non-TOE entity is included in the log record.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><td>All use of the user identification and authentication mechanism.</td><td>Events will be generated for attempted identification/ authentication (including whether it was successful or failed), and the username attempting to authenticate will be included in the log record, along with the origin or source of the attempt.</td></tr>
<tr><td>Changes to the time.</td><td>Changes to the time are logged, including old and new values for time, as well as origin of attempt</td></tr>
<tr><td>Initiation of an update to the TOE.</td><td>TOE updates are logged as configuration changes.</td></tr>
<tr><td>Termination of a remote session.</td><td>Termination of a remote session (due to inactivity) is logged (as a terminated cryptographic path).</td></tr>
<tr><td>Termination of an interactive session.</td><td>Termination of an Interactive session (due to logging off) is logged (as the session ending).</td></tr>
<tr><td>Initiation, termination and failures in trusted channels.</td><td>Requests for encrypted session negotiation are logged (including whether successful or failed). Similarly, when an established cryptographic channel or path is terminated or fails a log record is generated.
Also the initiator and target of any failed attempts to establish a trusted channels are identified.</td></tr>
<tr><td>Initiation, termination and failures in trusted paths.</td><td>Requests for encrypted session negotiation are logged (including whether successful or failed). Similarly, when an established cryptographic channel or path is terminated or fails a log record is generated.
The records include the claimed user identity.</td></tr>
<tr><td>Management functions</td><td>The use of the security management functions are logged, along with the origin or source of the attempt.</td></tr>
</table>
The TOE also sends audit logs to other entities (including other ISE nodes) using TLS protected syslog. ISE is configured by default to listen for UDP, TCP, and TLS-protected TCP. To configure this transfer to use TLS, the administrator must configure the secondary ISE box to send syslogs to the primary ISE via the “System” -> “Logging” tab, and set it to use “Secure Syslog” for the “Target Type”.

One can obtain reports on the log collection status for all Cisco ISE nodes. Log collection errors are noted by alarms via the dashboard</td></tr>
<tr><td>FAU_GEN.2</td><td>The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For</td></tr>
</table>

TOE SFRs	How the SFR is Met
	example, a human user, user identity or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. Refer to the Guidance documentation for configuration syntax and information.
FAU_STG_EXT.1	The TOE stores its own syslog events locally on the platform, and can offload events to other entities (including other ISE nodes) over TLS protected syslog. TCP syslog buffers events in a local file that is limited to a total of 100MB. The limit is specified as a file size, not a specific number of events. Overwriting is handled by wrapping to the beginning of the file (overwriting the oldest events). On the TOE, the local log files rotate after a certain size threshold is reached. The number of days of local log files is configurable, with the default of keeping records only up to last 7 days. From the Administration > System > Logging > Local Log Settings page an administrator is able to configure the storage period for logs in days and delete the existing log file. The administrator may delete all of the rolled over log files by the "Delete Local Logs Now" selection in the administration application. After the configured storage period of time has passed for logs the events exceeding the age are deleted. The administrators that are able to view the logs (at Operations > Reports > Catalog) are Super Admin, Monitoring Admin, or Helpdesk Admin. The administrator can also set the reports on peer ISE nodes, which is where the TOE stores remote syslog records that are received, to be maintained for a set number of days or delete them immediately if space becomes an issue using commands at the CLI.
FCS_CKM.1	The TOE implements a FIPS-approved Deterministic Random Bit Generator for Diffie-Hellman key establishment (conformant to NIST SP 800-56A), and for RSA key establishment schemes (conformant to NIST SP 800-56B). The TOE does not implement elliptic-curve-based key establishment schemes. For Diffie-Hellman Key Establishment, the TOE implements all sections of SP 800-56A, as outlined in Annex A.2 below. The TOE does not perform any operation marked as “Shall Not” or “Should not” in SP 800-56A. Additionally, the TOE does not omit any operation marked as “Shall.” For RSA Key Establishment, the TOE implements the all sections of SP 800-56B, as outlined in Annex A.2 below. The TOE does not perform any operation marked as “Shall Not” or “Should not” in SP 800-56B. Additionally, the TOE does not omit any operation marked as “Shall.”
FCS_CKM_EXT.4	The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs) in that none of the symmetric keys, pre-shared keys, or private keys are stored in plaintext form. Plaintext forms of keys are only stored in volatile memory (RAM), so zeroization is not required. This requirement applies to the secret keys used for symmetric encryption, private keys, and CSPs used to generate keys, which are zeroized immediately after use, or on system shutdown, etc. See the table in Annex A.1, below for more information.
FCS_COP.1(1)	The TOE provides symmetric encryption and decryption capabilities using AES, as defined in FIPS PUB 197, in CBC mode (128, 256 bits) as described in NIST SP 800-38A. Please see CAVP certificate #1475 and 1759 for validation details. These key sizes are used for both TLS and SSH.
FCS_COP.1(2)	The TOE will provide cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048. The ISE product can be configured

TOE SFRs	How the SFR is Met
	to generate key sizes of 1024 bit, but administrative guidance for the evaluated configuration instructs administrators to only use keys with size 2048. Please see CAVP certificate # 722 and 876 for validation details.
FCS_COP.1(3)	The TOE provides cryptographic hashing services using SHA-1 and SHA-256. Please see CAVP certificate # 1334 and 1544 for validation details. SHA-1 and SHA-256 are used for generating certificate signing requests or generating self-signed certificates on the TOE. SHA-1 is used for TLS and SSH.
FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 and HMAC-SHA-256, as specified in FIPS Pub 198-1,"The Keyed-Hash Message Authentication Code," and FIPS 180-3, "Secure Hash Standard." as part of the RADIUS Key Wrap functionality. Please see CAVP certificate # 868 and 1034 for validation details. Note that HMAC-SHA-1 is used for SSH connections, while HMAC-SHA-1, HMAC-SHA-256 are used for TLS.
FCS_RBG_EXT.1	The TOE implements a random bit generator (RBG) based on the AES-256 block cipher, as specified in FIPS Pub 140-2 Annex C: X9.31 Appendix 2.4. The appliance form factor TOE uses the Emulex Pilot III BMC chips. The RBG for the ISE appliance form-factor is seeded with a hardware-based noise source that uses a ring oscillator jitter based architecture. The ISE Virtual Machine model uses only software-based noise sources from /dev/urandom (special Linux device file).
FCS_SSH_EXT.1	The TOE implements SSHv2. There is no SSHv1 or telnet implementation on the TOE. SSH connections will be dropped if the TOE receives a packet larger than 262,144 bytes. Large packets are detected by the SSH implementation, and dropped internal to the SSH process. The TOE implementation of SSHv2 supports the following public key algorithm for authentication, RSA Signature Verification. The TOE also supports RSA public-keys and password-based authentication for administrative users accessing the TOE through SSHv2. The TOE implementation of SSHv2 supports the following encryption algorithms, AES-CBC-128, AES-CBC-256 to ensure confidentiality of the session. The TOE's implementation of SSHv2 supports hashing algorithms HMAC-SHA1 to ensure the integrity of the session. Note that the TOE complies with RFCs 4251, 4252, 4253, and 4254 with the exceptions of the following instances where the additional FCS_SSH_EXT requirements narrow the requirement: • FCS_SSH_EXT.1.4 requires only AES be used, while RFC 4253 lists 3des-cbc as REQUIRED. The TOE is capable of supporting the AES algorithms. • FCS_SSH_EXT.1.5 requires only SSH_RSA be used, while RFC 4253 lists ssh-dss as REQUIRED and ssh-rsa as RECOMMENDED. The TOE is capable of using SSH_RSA, and does not use SSH-DSS. • FCS_SSH_EXT.1.7 requires only Diffie-hellman-group14-sha1, while RFC 4253 requires both diffie-hellman-group14-sha1 and diffie-hellman-group1-sha1. The TOE is capable of enforcing group 14 as the only allowed method.
FCS_TLS_EXT.1	The TOE provides TLS 1.0, conformant to RFC 2246 and supports all four of the mandatory ciphersuites. • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_RSA_WITH_AES_256_CBC_SHA • TLS_DHE_RSA_WITH_AES_128_CBC_SHA

TOE SFRs	How the SFR is Met
	• TLS_DHE_RSA_WITH_AES_256_CBC_SHA None of the optional ciphersuites, per the requirement wording, are supported. The TOE only supports standard extensions, methods, and characteristics. TLS 1.0 is used for HTTPS/TLS for management purposes and to establish encrypted sessions with other instances of the TOE and IT entities to send/receive audit data. LDAPS has support for additional extensions to support communication with external authentication stores. The TOE's implementation of RFC 2246 includes all of the must statements, as well as does not violate the must not statements.
FCS_HTTPS_EXT.1	The TOE provides HTTPS, as specified in RFC 2818, to provide a secure interactive interface for remote administrative functions, and to support secure exchange of user authentication parameters during login. HTTPS uses TLS (as specified in FCS_TLS_EXT.1) to securely establish the encrypted remote session, and provide mutual authentication of TLS session endpoints through authentication of digital certificates within PKI. Note that port 80 is exposed on the product, but only as a redirect to port 443. HTTP connections are not allowed.
FDP_RIP.2	The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Packets that are not the required length use zeros for padding. Residual data is never transmitted from the TOE. Once packet handling is completed, its content is zeroized (overwritten with 0x00, fixed data based on the amount of padding, or random data) before the memory buffer which previously contained the packet is reused.
FIA_PMG_EXT.1	The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". Minimum password length is settable by the Authorized Administrator, with a default of six characters and can be configured for minimum password lengths of 15 characters or greater. It is configured via the Administration menu in the web-based, on the Admin Actions tab, under Password Policy.
FIA_UIA_EXT.1	The TOE requires all users to be successfully identified and authenticated before allowing any services and/or TSF mediated actions to be performed per the authentication policy. A pre-authentication banner is also displayed at both the CLI and GUI. Access to the web-based interface (via HTTPS), the CLI (SSH), and the console, all require at a minimum username and password be provided and successfully verified prior to access being granted. A successful login requires a correct username and password pair be confirmed, as existing in the local user database or a remote authentication store. The administrator can optionally configure stronger cryptographic protection, authentication and authorization for the CLI using SSH public key authentication. In the case of SSH public key authentication the client uses its private key to digitally sign. The ISE SSH server verifies the signature as the means of authentication.
FIA_UAU_EXT.2	The TOE can be configured to require local authentication and/or remote authentication via a remote authentication store as defined in the authentication policy. The process for authentication is the same for administrative access whether administration is occurring via the HTTPS web-based interface or via SSHv2 at the CLI. At initial login in the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password or public-key associated with the user account. The TOE then either grants administrative access (if the combination of username and

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>password or public-key is correct) or indicates that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.

The table below summarizes the authentication mechanisms that are supported at each interface.
<table>
<tr><th>Interface</th><th>Authentication Mechanism</th></tr>
<tr><td>Web-Based (GUI)</td><td>• local password-based
• remote password-based</td></tr>
<tr><td>Remote SSH (CLI)</td><td>• SSH public key
• local password-based
• remote password-based</td></tr>
<tr><td>Local Console (CLI)</td><td>• local password-based
• remote password-based</td></tr>
</table></td></tr>
<tr><td>FIA_UAU.7</td><td>When a user enters their password at the ISE web-based interface only ‘*’ characters are displayed and at the CLI nothing is displayed so that the user password is obscured. Also, the error displayed for the user does not give clues about which part of the credentials entered for authentication failed.</td></tr>
<tr><td>FMT_MTD.1</td><td>The TOE restricts access to the management functions to the authorized administrator. As noted in FMT_SMR.2, the TOE supports two levels of administrative users, the CLI-admin (local console or SSHv2 accessible) and the web-based admin user. The same functionality is available on the TOE via the web-based interface and CLI, with the exception that only the CLI-admin can start and stop the ISE application and reload (update) or shutdown the ISE appliance via the CLI.
None of the administrative functions of the product are available prior to administrator log-in.</td></tr>
<tr><td>FMT_SMF.1</td><td>The TOE provides all the capabilities necessary to securely manage the TOE, the services provided by the TOE. The management functionality of the TOE is provided through the TOE CLI or HTTPS web-based interface. The specific management capabilities available from the TOE are identified in the text of FMT_SMF.1.</td></tr>
<tr><td>FMT_SMR.2</td><td>Cisco ISE provides role-based access control (RBAC) policies that ensure security by restricting administrative privileges. RBAC policies are associated with default admin groups to define roles and permissions. A standard set of permissions (for menu as well as data access) is paired with each of the predefined admin groups, and is thereby aligned with the associated role and job function.

RBAC restricts system access to authorized users through the use of roles that are then associated with admin groups. Each admin group has the ability to perform certain tasks with permissions that are defined by an RBAC policy. Policies restrict or allow a person to perform tasks that are based on the admin group (or groups) to which that person is assigned. A user can be assigned to multiple roles, which provides them with privileges for each role to which they are assigned.
A specialized administrator role has the ability to customize permissions and admin groups and to create custom policies. The default Cisco ISE RBAC policies cannot be modified, however.

An individual who manages or performs a specific type of administrative task using the Cisco ISE user interface is considered an admin (or administrator). Administrators are dependent upon the admin role assigned to them, which limits</td></tr>
</table>

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
the network access or tasks they can perform (a role-based access approach). Using the Cisco ISE user interfaces (CLI and web-based), administrator roles can perform the following tasks:
• Change admin or user passwords
• Manage deployments, helpdesk operations, monitoring and troubleshooting nodes, and network devices
• Manage Cisco ISE services policies and admin access, Cisco ISE administrator accounts and roles, Cisco ISE administrative functions, and Cisco ISE system configuration and operations

The TOE supports two categories of administrative user, the CLI-admin and the web-based admin user.

The CLI-admin user and the web-based admin user can perform the following ISE system-related tasks:
• Backup and restore the Cisco ISE application data
• Display any system, application, or diagnostic logs on the Cisco ISE appliance
• Apply Cisco ISE software patches, maintenance releases, and upgrades

Following are the default roles for the web-based admin and their capabilities.
Web-based Admin Group Role Descriptions
<table>
<tr><td>Helpdesk Admin</td><td>This role provides access for querying all monitoring and troubleshooting operations and within the Cisco ISE administrative console, and can perform the following tasks:
• Run all reports
• Run all troubleshooting flows
• View the Cisco ISE dashboard and livelogs
• View alarms
This role cannot create, update, or delete reports, troubleshooting flows, live authentications, or alarms.</td></tr>
<tr><td>Identity Admin</td><td>This role provides access for managing all of the internal user identities that use the Cisco ISE administrative console across the Cisco ISE network. This role has read and write permissions on identities, endpoints, and identity groups (user identity groups and endpoint identity groups).</td></tr>
<tr><td>Monitoring Admin</td><td>This role provides access to all monitoring and troubleshooting operations within the Cisco ISE administrative console, and can perform the following tasks:
• Manage all reports (run, create, and delete)
• Run all troubleshooting flows
• View the Cisco ISE dashboard and livelogs
• Manage alarms (create, update, view, and delete)</td></tr>
<tr><td>Network Device Admin</td><td>This role provides access for Cisco ISE administrators that manage only the Cisco ISE network device repository and perform tasks such as adding, updating, or deleting devices. This role has the following permissions:
• Read and write permissions on network devices
• Read and write permissions on NDGs and all network resources object types</td></tr>
<tr><td>Policy Admin</td><td>This role provides access for Cisco ISE policy administrators who are responsible for creating and managing the policies for all Cisco ISE services across the network that are related to authentication, authorization, posture, profiler, and client</td></tr>
</table>
</td></tr>
</table>

TOE SFRs	How the SFR is Met
	provisioning. This role has the following permissions: • Read and write permissions on all the elements used in policies, such as authorization profiles, NDGs, and conditions • Read and write permissions on identities, endpoints, and identity groups (user identity groups and endpoint identity groups) • Read and write permissions on services policies
	RBAC Admin: This role provides full access (read and write permissions) to perform all activities under the Operations tab and partial access to some menu items under the Administration tab. This role has the following permissions: • View the authentication details • Enable or disable endpoint protection service • Create, edit, and delete alarms; generate and view reports; and use Cisco ISE to troubleshoot problems in your network • Read permissions on administrator account settings and admin group settings • View permissions on admin access and data access permissions along with the RBAC policy page.
	Super Admin: This role provides access to every Cisco ISE administrative function. This role is assigned to the default administrator account, and has create, read, update, delete, and eXecute (CRUDX) permissions on all Cisco ISE resources. Note The super admin user cannot modify the default system-generated RBAC policies and permissions. To do this, you must create new RBAC policies with the necessary permissions based on your needs, and map these policies to any admin group.
	System Admin: This role provides access for Cisco ISE administrators who are responsible for Cisco ISE configuration and operations. This role provides full access (read and write permissions) to perform all activities under the Operations tab and partial access to some menu items under the Administration tab. This role has the following permissions: • Read permissions on administrator account settings and administrator group settings • Read permissions on admin access and data access permissions along with the RBAC policy page. • Read and write permissions for all options under the Administration > System menu. • View the authentication details • Enable or disable endpoint protection service • Create, edit, and delete alarms; generate and view reports; and use Cisco ISE to troubleshoot problems in your network
	Only the CLI-admin user can perform the following Cisco ISE system-related tasks: • Start and stop the ISE application software • Reload or shutdown the ISE appliance Because only the CLI-admin user can perform these services, the CLI-admin user credentials must be protected. It is noted that only a user assigned these privileges can access the ISE CLI.

TOE SFRs	How the SFR is Met
	The ability to administer the TOE locally is provided through a console connection to the appliance models or dedicated hardware hosting the Virtual Machine instance. The ability to administer the TOE remotely is provided via SSH protected access to the ISE CLI or TLS protected access to the web-based interface. The 'Authorized Administrator' specified in the SFRs is synonymous/equivalent to the entire set of TOE default administrative levels/administrators.
FPT_ITT.1	The communication between instances of the ISE appliance to share logging events and configuration data is protected (from disclosure and modification) via TLS session.
FPT_SKP_EXT.1 and FPT_APW_EXT.1	The TOE by default encrypts all locally defined user passwords using MD5 hashing for CLI passwords, and AES-ECB encryption for GUI credentials. In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators. The TOE stores all private keys in a secure directory that is not readily accessible to administrators. All pre-shared and symmetric keys are stored in encrypted form to prevent access. TOE is designed specifically to not disclose any keys stored in the TOE. All pre-shared and symmetric keys are stored in encrypted form using AES encryption to additionally obscure access.
FPT_STM.1	The TOE provides a source of date and time information, used in audit timestamps and in validating service requests. This function can be configured from the Administration > System > Settings > System Time page by a Super Admin or System Admin role only. The clock function is reliant on the system clock provided by the underlying hardware. The TOE can optionally be set to receive time from an NTP server or multiple NTP servers. If an NTP server is used, the TOE supports signature verification of the timestamp from the time server. . This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions. The time information is also used to set system time, determining AAA timeout, and administrative session timeout
FPT_TUD_EXT.1	The TOE has specific versions that can be queried by an administrator from the CLI using the "show version" command, or from the administration GUI, lower left "Help" > About Identity Services Engine. When updates are made available by Cisco, an administrator (specifically the Super Admin or System Admin) can obtain and install those updates. The cryptographic checksums (i.e., public hashes), as published on http://www.cisco.com/cisco/web/support/index.html, are used to verify software/firmware update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to actually update the applicable TOE components. The administrator must verify the correctness of the hash using the upgrade verification mechanism on the TOE, which lists the SHA256 hash of the package to be installed. Information on how to check for updates and confirm their checksum will be given in the Operational Guidance. Logs for update actions are located in Operations > Reports > Catalog > Server Instance Report.
FPT_TST_EXT.1	As ISE uses a FIPS 140-2 validated cryptographic library, the TOE runs a suite of self-tests during initial start-up to verify its correct operation. These tests check the integrity of the code, and the correct operation of each cryptographic algorithm and method used (i.e. AES-CBC, SHA-1, etc.) If any of the tests fail,

TOE SFRs	How the SFR is Met
	the administrative web-based will not be accessible, and the security administrator will have to log into the CLI to determine which test failed and why. If the tests pass successfully the FIPS badge is displayed on the web-based screen and the web-based will be accessible for login by the security administrator. During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for all the cryptographic modules perform the POST for the corresponding component (hardware or software). These tests include: • AES Known Answer Test • RSA Signature Known Answer Test (both signature/verification) • Power up bypass test • RNG Known Answer Test • Diffie Hellman test • HMAC Known Answer Test • SHA-1/256/512 Known Answer Test • Triple-DES Known Answer Test • Software Integrity Test If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen, and saved in the crashinfo file. All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic. These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected.
FTA_SSL_EXT.1, FTA_SSL.3, and FTA_SSL.4	An administrator can configure maximum inactivity times for both local and remote administrative sessions. When a session is inactive (i.e., no session input) for the configured period of time the TOE will terminate the session, requiring the administrator to log in again to establish a new session when needed. At the CLI, once the administrator establishes a new session, they have the option of seeing data from their previous sessions. This is selected after successful authentication and only gives access to that user's previous sessions. The ability to configure these settings is limited to the Super Admin or System Admin. It is configured via the Administration > System > Admin Access > Settings > Session Timeout page. Each administrator logged onto the TOE can manually terminate her session using the "LogOut" link in the web-based or the "exit" or "forceout <username>" commands at the CLI.
FTA_TAB.1	The TOE displays a privileged Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE. The TOE also displays a banner at the web-based interface that is accessed via HTTPS. The local console access to the TOE takes the administrator to the CLI, where the administrative banner is displayed.
FTP_ITC.1	The TOE protects communications with devices to which is sends syslogs, including other iterations of ISE, using TLS. This protects the data from disclosure by encryption and by checksums that verify that data has not been modified. The TOE also protects communications with external authentication stores in the following manner:

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><th>External Authentication Store</th><th>Protection Mechanism</th></tr>
<tr><td>LDAP Server(s)</td><td>TLS</td></tr>
<tr><td>Active Directory Directory Services (acting as the Secure LDAP server)</td><td>TLS</td></tr>
</table>
</td></tr>
<tr><td>FTP_TRP.1</td><td>All remote administrative communications take place over a secure encrypted SSHv2 (CLI) session or TLS (web-based GUI) session. Both SSHv2 and TLS sessions are protected using AES encryption. The remote users are able to initiate both TLS and SSHv2 communications with the TOE.</td></tr>
</table>

7 RATIONALE

This section describes the rationale for the Security Objectives and Security Functional Requirements as defined within this Security Target (and as based on the NDPP). The following matrix is the typical display that is drawn from the information presented in Sections 2 and 3 of the NDPP.

7.1 Rationale for TOE Security Objectives

The security objectives rationale shows how the security objectives correspond to threats and organizational security policies and provides a justification of that tracing.

Table 18: Threat/Objectives/Policies Mappings

	T.UNAUTHORIZED_ACCESS	T.UNAUTHORIZED_UPDATE	T.ADMIN_ERROR	T.UNDETECTED_ACTIONS	T.USER_DATA_REUSE	T.TSF_FAILURE	P.ACCESS BANNER
O.PROTECTED_COMMUNICATIONS	X						
O.VERIFIABLE_UPDATES		X					
O.SYSTEM_MONITORING				X			
O.DISPLAY_BANNER							X
O.TOE_ADMINISTRATION			X				
O.RESIDUAL_INFORMATION_CLEARING					X		
O.SESSION_LOCK	X						
O.TSF_SELF_TEST						X	

Table 19: Threat/Policies/TOE Objectives Rationale

Objective	Rationale
Security ObjectivesDrawn from NDPP	
O.PROTECTED_COMMUNICATIONS	This security objective is necessary to counter the threat: T.UNAUTHORIZED_ACCESS to ensure the communications with the TOE is not compromised
O.VERIFIABLE_UPDATES	This security objective is necessary to counter the threat T.UNAUTHORIZED_UPDATE to ensure the end user has not installed a malicious update, thinking that it was legitimate.
O.SYSTEM_MONITORING	This security objective is necessary to counter the T.UNDETECTED_ACTIONS to ensure activity is monitored so the security of the TOE is not compromised.
O.DISPLAY_BANNER	This security objective is necessary to address the Organization

Objective	Rationale
	Security Policy P.ACCESS_BANNER to ensure an advisory notice and consent warning message regarding unauthorized use of the TOE is displayed before the session is established.
O.TOE_ADMINISTRATION	This security objective is necessary to counter the T.ADMIN_ERROR that ensures actions performed on the TOE are logged so that indications of a failure or compromise of a TOE security mechanism are known and corrective actions can be taken.
O.RESIDUAL_INFORMATION_CLEARING	This security objective is necessary to counter the threat T.USER_DATA_REUSE so that data traversing the TOE could inadvertently be sent to a user other than that intended by the sender of the original network traffic.
O.SESSION_LOCK	This security objective is necessary to counter the threat: T.UNAUTHORIZED_ACCESS to ensure accounts cannot be compromised and used by an attacker that does not otherwise have access to the TOE.
O.TSF_SELF_TEST	This security objective is necessary to counter the threat T.TSF_FAILURE to ensure failure of mechanisms do not lead to a compromise in the TSF.

7.2 Rationale for the Security Objectives for the Environment

The security objectives for the environment rationale shows how the security objectives for the environment correspond to assumptions and provides a justification of that tracing.

Table 20: Assumptions/Environment Objectives Mappings

	OE.NO_GENERAL_PURPOSE	OE.PHYSICAL	OE.TRUSTED_ADMIN
A.NO_GENERAL_PURPOSE	X		
A.PHYSICAL		X	
A.TRUSTED_ADMIN			X

Table 21: Assumptions/Threats/Objectives Rationale

Environment Objective	Rationale
OE.NO_GENERAL_PURPOSE	This security objective is necessary to address the assumption A.NO_GENERAL_PURPOSE by ensuring there are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) capabilities on the TOE.
OE.PHYSICAL	This security objective is necessary to address the assumption A.PHYSICAL by ensuring the TOE and the data it contains is

Environment Objective	Rationale
	physically protected from unauthorized access.
OE.TRUSTED_ADMIN	This security objective is necessary to address the assumption A.TRUSTED_ADMIN by ensuring the administrators are non-hostile and follow all administrator guidance.

7.3 Rationale for requirements/TOE Objectives

The security requirements are derived according to the general model presented in Part 1 of the Common Criteria. Specifically, the tables below illustrate the mapping between the security requirements and the security objectives and the relationship between the threats, policies and IT security objectives. The functional and assurance requirements presented in this Protection Profile are mutually supportive and their combination meets the stated security objectives.

Table 22: Security Objective to Security Requirements Mappings

	O.PROTECTED_COMMUNICATIONS	O.VERIFIABLE_UPDATES	O.SYSTEM_MONITORING	O.DISPLAY_BANNER	O.TOE_ADMINISTRATION	O.RESIDUAL_INFORMATION_CLEARING	O.SESSION_LOCK	O.TSF_SELF_TEST
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	X						
FCS_COP.1(3)	X	X						
FCS_COP.1(4)	X							
FCS_RBG_EXT.1	X							
FCS_HTTPS_EXT.1	X							
FCS_TLS_EXT.1	X							
FCS_SSH_EXT.1	X							

FDP_RIP.2						X		
FIA_PMG_EXT.1					X			
FIA_UIA_EXT.1					X			
FIA_UAU_EXT.2					X			
FIA_UAU.7					X			
FMT_MTD.1					X			
FMT_SMF.1					X			
FMT_SMR.2					X			
FPT_ITT.1	X							
FPT_STM.1			X					
FPT_SKP_EXT.1	X							
FPT_APW_EXT.1	X							
FPT_TUD_EXT.1		X						
FPT_TST_EXT.1								X
FTA_SSL_EXT.1					X		X	
FTA_SSL.3					X		X	
FTA_SSL.4					X			
FTA_TAB.1				X				
FTP_ITC.1	X							
FTP_TRP.1	X							

Table 23: Objectives to Requirements Rationale

Objective	Rationale
Security Functional Requirements Drawn from Security Requirements for NDPP	
O.PROTECTED_COMMUNICATIONS	The SFRs FCS_CKM.1, FCS_CKM_EXT.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_RBG_EXT.1, FCS_SSH_EXT.1, FCS_TLS_EXT.1, FCS_HTTPS_EXT.1, FPT_SKP_EXT.1, FPT_APW_EXT.1, FPT_ITT.1, FTP_ITC.1, FTP_TRP.1 meet this objective by ensuring the communications between the TOE and endpoints are secure by implementing the encryption protocols as defined in the SFRs and as specified by the RFCs.
O.VERIFIABLE_UPDATES	The SFRs, FPT_TUD_EXT.1, FCS_COP.1(2), FCS_COP.1(3) meet this objective by ensuring the update was downloaded via secure communications, is from a trusted source, and the update can be verified by cryptographic mechanisms prior to installation.
O.SYSTEM_MONITORING	The SFRs, FAU_GEN.1, FAU_GEN.2, FAU_STG_EXT.1, FPT_STM.1 meet this objective by auditing actions on the TOE. The audit records identify the user associated with the action/event, whether the action/event was successful or failed, the type of action/event, and the date/time the action/event occurred. The audit logs are transmitted securely to a remote syslog server. If connectivity to the remote syslog server is lost, the TOE will block

Objective	Rationale
	new permit actions.
O.DISPLAY_BANNER	The SFR, FTA_TAB.1 meets this objective by displaying a advisory notice and consent warning message regarding unauthorized use of the TOE.
O.TOE_ADMINISTRATION	The SFRs, FIA_UIA_EXT.1, FIA_PMG_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7, FMT_MTD.1, FMT_SMF.1, FMT_SFR.2, FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4 meet this objective by ensuring the TOE supports a password-based authentication mechanism with password complexity enforcement such as, strong passwords, password life-time constraints, providing current password when changing the password, obscured password feedback when logging in, and passwords are not stored in plaintext. The objective is further met by ensuring restrictive default values are enforced on the SFPs (authorization and flow control), that only Authorized Administrators to override the default values, that the TOE provides the management and configuration features to securely manage the TOE and that those functions are restricted to the authorized administrator, and the implementation of session termination after an administrative configurable inactivity time period whereas the user must be re-authenticated,
O.RESIDUAL_INFORMATION_CLEARING	The SFR, FDP_RIP.2 meets this objective by ensuring no left over user data from the previous transmission is included in the network traffic.
O.SESSION_LOCK	The SFRs, FTA_SSL_EXT.1, FTA_SSL.3 meet this objective by terminating a session due to meeting/exceeding the inactivity time limit.
O.TSF_SELF_TEST	The SFR, FPT_TST_EXT.1 meets this objective by performing self-test to ensure the TOE is operating correctly and all functions are available and enforced.

ANNEX A: ADDITIONAL INFORMATION

A.1 Key Protection and Zeroization

The following table describes the key zeroization referenced by FCS_CKM_EXT.4 provided by the TOE.

Table 24: TOE Key Zeroization

Name	Description	Zeroization
Diffie-Hellman Shared Secret	The value is zeroized after it has been given back to the consuming operation. The value is overwritten by 0's.	Automatically after completion of DH exchange, by calling a specific API within the two crypto modules, when module is shutdown, or reinitialized. Overwritten with: 0x00
Diffie Hellman private exponent	The function returns the value to the TOE and then calls the function to perform the zeroization of the generated key pair . These values are automatically zeroized after generation and once the value has been provided back to the actual consumer.	Zeroized upon completion of DH exchange, by calling a specific API within the two crypto modules, when module is shutdown, or reinitialized. Overwritten with: 0x00
ISE server certificate	The certificate is used for TLS, HTTPS client connections, secure transport between ISE nodes, and secure connections to authentication stores.	Generation of a new certificate. Overwritten with: 0x00
SSH Private Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents) via API call. This overwrites the key with all 0's.	Generation of a new key Overwritten with: 0x00
SSH Session Key	The results zeroized by overwriting the values with 0x00. This is done when a session is ended.	Automatically when the SSH session is terminated. Overwritten with: 0x00

A.2 800-56 Compliance

The TOE is compliant to [NIST SP 800-56A] and [NIST SP 800-56B] as described in Table 25 and Table 26 below.

Table 25: NIST SP 800-56A Compliance

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
5.1 Cryptographic Hash Functions	None.	None.	Yes	N/A
5.2 Message Authentication Code (MAC) Algorithm	None.	None.	Yes	N/A
5.2.1 MacTag Computation	None.	None.	Yes	N/A
5.2.2 MacTag Checking	N/A, no shall statements	None.	Yes	N/A
5.2.3 Implementation Validation Message	None.	None.	Yes	N/A
5.3 Random Number Generation	None.	None.	Yes	N/A
5.4 Nonces	None.	"a random nonce **should** be used"	Yes	N/A
5.5 Domain Parameters	None.	None.	Yes	N/A
5.5.1 Domain Parameter Generation	N/A, no shall statements	"If the appropriate security strength does not have an FFC parameter set, then Elliptic Curve Cryptography **should** be used"	Yes	N/A
5.5.1.1 FFC Domain Parameter Generation	None.	None.	Yes	N/A
5.5.1.2 ECC Domain Parameter Generation	N/A, no ECC in use.	None.	Yes	N/A
5.5.2 Assurances of Domain Parameter Validity	None.	None.	Yes	N/A
5.5.3 Domain	None.	None.	Yes	N/A

[1] This column does not include "should/should not" statements that relate to the "owner", "recipient", "application", or "party" as they are outside of the scope of the TOE.

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
Parameter Management				
5.6 Private and Public Keys	N/A, no shall statements	None.	Yes	N/A
5.6.1 Private/Public Key Pair Generation	N/A, no shall statements	None.	Yes	N/A
5.6.1.1 FFC Key Pair Generation	None.	None.	No	N/A
5.6.1.2 ECC Key Pair Generation	N/A, no ECC in use.	None.	Yes	N/A
5.6.2 Assurances of the Arithmetic Validity of a Public Key	None.	None.	Yes	N/A
5.6.2.1 Owner Assurances of Static Public Key Validity	None. Static key is not supported.	None.	Yes	N/A
5.6.2.2 Recipient Assurances of Static Public Key Validity	None. Static key is not supported.	None.	Yes	N/A
5.6.2.3 Recipient Assurances of Ephemeral Public Key Validity	None.	None.	Yes	N/A

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
5.6.2.4 FFC Full Public Key Validation Routine	None.	None.	Yes	N/A
5.6.2.5 ECC Full Public Key Validation Routine	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.6.2.6 ECC Partial Public Key Validation Routine	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.6.3 Assurances of the Possession of a Static Private Key	None. Static key is not supported.	None.	Yes	N/A
5.6.3.1 Owner Assurances of Possession of a Static Private Key	None. Static key is not supported.	None.	Yes	N/A
5.6.3.2 Recipient Assurance of Owner's Possession of a Static Private Key	None. Static key is not supported.	None.	Yes	N/A
5.6.3.2.1 Recipient Obtains Assurance through a Trusted Third Party	N/A, no shall statements	None.	Yes	N/A
5.6.3.2.2 Recipient Obtains Assurance Directly from the Claimed Owner	None. Static key is not supported.	None.	Yes	N/A
5.6.4 Key Pair Management	N/A, no shall statements	None.	Yes	N/A
5.6.4.1 Common Requirements on Static and Ephemeral Key Pairs	None.	None.	Yes	N/A
5.6.4.2 Specific Requirements on Static Key Pairs	None. Static key is not supported.	None.	Yes	N/A
5.6.4.3 Specific Requirements on Ephemeral Key Pairs	None.	"An ephemeral key pair **should** be generated as close to its time of use as possible"	Yes	N/A
5.7 DLC Primitives	None.	None.	Yes	N/A
5.7.1 Diffie-Hellman	N/A, no shall statements	None.	Yes	N/A

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
Primitives				
5.7.1.1 Finite Field Cryptography Diffie-Hellman (FFC DH) Primitive	N/A, no shall statements	None.	Yes	N/A
5.7.1.2 Elliptic Curve Cryptography Cofactor Diffie-Hellman (ECC CDH) Primitive	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.7.2 MQV Primitives	N/A, no shall statements	None.	Yes	N/A
5.7.2.1 Finite Field Cryptography MQV (FFC MQV) Primitive	N/A, no shall statements	None.	Yes	N/A
5.7.2.1.1 MQV2 Form of the FFC MQV Primitive	N/A, no shall statements	None.	Yes	N/A
5.7.2.1.2 MQV1 Form of the FFC MQV Primitive	N/A, no shall statements	None.	Yes	N/A
5.7.2.2 ECC MQV Associate Value Function	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.7.2.3 Elliptic Curve Cryptography MQV (ECC MQV) Primitive	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.7.2.3.1 Full MQV Form of the ECC MQV Primitive	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.7.2.3.2 One-Pass Form of the ECC MQV Primitive	N/A, no ECC in use. N/A, no shall statements	None.	Yes	N/A
5.8 Key Derivation Functions for Key Agreement Schemes	In TLS the MAC key is used for traffic protection as well as key confirmation.	None.	No	Only applicable if If Key Confirmation (KC) or implementation validation testing are to be performed as

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
				specified in Section 8.
5.8.1 Concatenation Key Derivation Function (Approved Alternative 1)	See above.	None.	Yes	Only applicable if If Key Confirmation (KC) or implementation validation testing are to be performed as specified in Section 8.
5.8.2 ASN.1 Key Derivation Function (Approved Alternative 2)	See above.	None.	Yes	Only applicable if If Key Confirmation (KC) or implementation validation testing are to be performed as specified in Section 8.
6. Key Agreement	None.	None.	Yes	N/A
6.1 Schemes Using Two Ephemeral Key Pairs, C(2)	N/A, no shall statements	None.	Yes	N/A
6.1.1 Each Party Has a Static Key Pair and Generates an Ephemeral Key Pair, C(2, 2)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.1.1.1 dhHybrid1, C(2, 2, FFC DH)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.1.1.2 Full Unified Model, C(2, 2, ECC CDH)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.1.1.3 MQV2, C(2, 2, FFC MQV)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.1.1.4 Full MQV, C(2, 2, ECC MQV)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.1.1.5 Rationale for Choosing a C(2, 2) Scheme	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
6.1.2 Each Party Generates an Ephemeral Key Pair; No Static Keys are Used, C(2, 0)	None.	None.	Yes	N/A
6.1.2.1 dhEphem, C(2, 0, FFC DH)	None.	None.	Yes	N/A
6.1.2.2 Ephemeral Unified Model, C(2, 0, ECC CDH)	N/A, no ECC in use.	None.	Yes	N/A
6.1.2.3 Rationale for Choosing a C(2, 0) Scheme	N/A, no shall statements	None.	Yes	N/A
6.2 Schemes Using One Ephemeral Key Pair, C(1)	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)
6.2.1 Initiator Has a Static Key Pair and Generates an Ephemeral Key Pair; Responder Has a Static Key Pair, C(1, 2)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.2.1.1 dhHybridOneFlow, C(1, 2, FFC DH)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.2.1.2 One-Pass Unified Model, C(1, 2, ECC CDH)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.2.1.3 MQV1, C(1, 2, FFC MQV)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.2.1.4 One-Pass MQV, C(1, 2, ECC MQV)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.2.1.5 Rationale for Choosing a C(1, 2) Scheme	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)
6.2.2 Initiator Generates Only an Ephemeral Key Pair; Responder Has Only a Static Key Pair, C(1, 1)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.2.2.1 dhOneFlow, C(1,	None.	None.	Yes	N/A, TOE uses C(2,0)

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
1, FFC DH)				
6.2.2.2 One-Pass Diffie-Hellman, C(1, 1, ECC CDH)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.2.2.3 Rationale in Choosing a C(1, 1) Scheme	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)
6.3 Scheme Using No Ephemeral Key Pairs, C(0, 2)	None.	None.	Yes	N/A, TOE uses C(2,0)
6.3.1 dhStatic, C(0, 2, FFC DH)	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)
6.3.2 Static Unified Model, C(0, 2, ECC CDH)	N/A, no ECC in use.	None.	Yes	N/A, TOE uses C(2,0)
6.3.3 Rationale in Choosing a C(0, 2) Scheme	N/A, no shall statements	None.	Yes	N/A, TOE uses C(2,0)
7. DLC-Based Key Transport	None.	None.	Yes	N/A, TOE uses C(2,0)
8. Key Confirmation	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.1 Assurance of Possession Considerations when using Key Confirmation	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.2 Unilateral Key Confirmation for Key Agreement Schemes	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.3 Bilateral Key Confirmation for Key Agreement	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
Schemes				agreement schemes is not specified, since neither party has a static key pair
8.4 Incorporating Key Confirmation into a Key Agreement Scheme	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.1 C(2, 2) Scheme with Unilateral Key Confirmation Provided by U to V	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.2 C(2, 2) Scheme with Unilateral Key Confirmation Provided by V to U	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.3 C(2, 2) Scheme with Bilateral Key Confirmation	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.4 C(1, 2) Scheme with Unilateral Key Confirmation Provided by U to V	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.5 C(1, 2) Scheme with Unilateral Key Confirmation	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement

Section	Exceptions to Shall/Shall Not Statement(s)	Should (Not) Statements[1]	TOE Compliant?	Rationale
Provided by V to U				schemes is not specified, since neither party has a static key pair
8.4.6 C(1, 2) Scheme with Bilateral Key Confirmation	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.7 C(1, 1) Scheme with Unilateral Key Confirmation Provided by V to U	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.8 C(0, 2) Scheme with Unilateral Key Confirmation Provided by U to V	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.9 C(0, 2) Scheme with Unilateral Key Confirmation Provided by V to U	N/A, no shall statements	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair
8.4.10 C(0, 2) Scheme with Bilateral Key Confirmation	None.	None.	Yes	Key confirmation for the C(2, 0) key agreement schemes is not specified, since neither party has a static key pair

Table 26: NIST SP 800-56B Compliance

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
5 Cryptographic Elements	None.	None.	Yes	N/A
5.1 Cryptographic Hash Functions	None.	None.	Yes	N/A
5.2 Message Authentication Code (MAC) Algorithm	None.	None.	Yes	N/A
5.2.1 MacTag Computation	None.	None.	Yes	N/A
5.2.2 MacTag Checking	N/A, no shall statements	None.	Yes	N/A
5.2.3 Implementation Validation Message	None.	None.	Yes	N/A
5.3 Random Bit Generation	None.	None.	Yes	N/A
5.4 Prime Number Generators	Only approved prime number generation methods shall be employed in this Recommendation.	None.	No	We are ANSI X9.31 compliant. However, the requirements in this SP have recently changed.
5.5 Primality Testing Methods	None.	None.	Yes	N/A
5.6 Nonces	None.	"When using a nonce, a random nonce **should** be used."	Yes	N/A
5.7 Symmetric Key-Wrapping Algorithms	N/A for TLS and SSH.	None.	Yes	N/A
5.8 Mask Generation Function (MGF)	None.	None.	Yes	N/A
5.9 Key Derivation Functions for Key Establishment Schemes	None.	None.	Yes	ISE uses other allowable methods and the protocols as referenced in

[2] This column does not included "should/should not" statements that relate to the "owner", "recipient", "application", or "party" as they are outside of the scope of the TOE.

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
				FIPS 140-2 Annex D
5.9.1 Concatenation Key Derivation Function (Approved Alternative 1)	None.	None.	Yes	N/A
5.9.2 ASN.1 Key Derivation Function (Approved Alternative 2)	None.	None.	Yes	N/A
6 RSA Key Pairs	N/A, no shall statements	None.	Yes	N/A
6.1 General Requirements	None.	"a key pair used for schemes specified in this recommendation **should not** be used for any schemes not specified herein"	Yes	N/A
6.2 Criteria for RSA Key Pairs for Key Establishment	N/A, no shall statements	None.	Yes	N/A
6.2.1 Definition of a Key Pair	None.	None.	Yes	N/A
6.2.2 Formats	N/A, no shall statements	None.	Yes	N/A
6.2.3 Parameter Length Sets	None.	"The MacKey length shall meet or exceed the target security strength, and **should** meet or exceed the security strength of the modulus."	Yes	N/A
6.3 RSA Key Pair Generators	None.	None.	Yes	N/A
6.3.1 RSAKPG1 Family: RSA Key Pair Generation with a Fixed Public Exponent	No shall statements (def of approved key pair generator)	None.	Yes	N/A
6.3.2 RSAKPG2 Family: RSA Key Pair Generation with a Random Public Exponent	No shall statements (def of approved key pair generator)	None.	Yes	N/A
6.4 Assurances of Validity	N/A, no shall statements	None.	Yes	N/A
6.4.1 Assurance of Key Pair Validity	None.	None.	Yes	N/A
6.4.2 Recipient Assurances of	None.	None.	Yes	N/A

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
Public Key Validity				
6.5 Assurances of Private Key Possession	None.	None.	Yes	N/A
6.5.1 Owner Assurance of Private Key Possession	None.	None.	Yes	N/A
6.5.2 Recipient Assurance of Owner's Possession of a Private Key	None.	None.	Yes	N/A
6.6 Key Confirmation	None.	None.	Yes	N/A
6.6.1 Unilateral Key Confirmation for Key Establishment Schemes	Unilateral Key Confirmation is done for both TLS and SSH, however it varies slightly from that outlined here.	None.	Yes	N/A
6.6.2 Bilateral Key Confirmation for Key Establishment Schemes	N/A, no shall statements	None.	Yes	N/A
6.7 Authentication	N/A, no shall statements	None.	Yes	N/A
7 IFC Primitives and Operations	N/A, no shall statements	None.	Yes	N/A
7.1 Encryption and Decryption Primitives	N/A, no shall statements	None.	Yes	N/A
7.1.1 RSAEP	N/A, no shall statements	None.	Yes	N/A
7.1.2 RSADP	N/A, no shall statements	"Care **should** be taken to ensure that an implementation of RSADP does not reveal even partial information about the value of k."	Yes	N/A
7.2 Encryption and Decryption Operations	N/A, no shall statements	None.	Yes	N/A
7.2.1 RSA Secret Value Encapsulation (RSASVE)	N/A, no shall statements	"Care **should** be taken to ensure that an implementation does not reveal information about the encapsulated secret value Z." "the observable behavior of the I2BS routine should not	Yes	N/A

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
		reveal even partial information about the byte string Z."		
7.2.2 RSA with Optimal Asymmetric Encryption Padding (RSA-OAEP)	None.	"Care should be taken to ensure that the different error conditions that may be detected in Step 5 above cannot be distinguished from one another by an opponent, whether by error message or by process timing." "A single error message **should** be employed and output the same way for each type of decryption error. There **should** be no difference in the observable behavior for the different RSA-OAEP decryption errors." "care should be taken to ensure that even if there are no errors, an implementation does not reveal partial information about the encoded message EM" "the observable behavior of the mask generation function **should not** reveal even partial information about the MGF seed employed in the process"	Yes	N/A
7.2.3 RSA-based Key-Encapsulation Mechanism with a Key-Wrapping Scheme (RSA-KEM-KWS)	N/A, no shall statements	"Care **should** be taken to ensure that the different error conditions in Steps 2.2, 4, and 6 cannot be distinguished from one another by an opponent, whether by error message or timing." "A single error message **should** be employed and output the same way for each error type. There **should** be no difference in timing or other behavior for the different errors. In addition, care **should** be taken to ensure that even if there are no errors, an implementation does not reveal partial information about the shared secret	Yes	N/A

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
		Z." "care **should** be taken to ensure that an implementation does not reveal information about the encapsulated secret value Z. For instance, the observable behavior of the KDF **should not** reveal even partial information about the Z value employed in the key derivation process."		
8 Key Agreement Schemes	In many cases TLS is deployed only with server authentication.	None.	Yes	N/A
8.1 Common Components for Key Agreement	N/A, no shall statements	None.	Yes	N/A
8.2 The KAS1 Family	N/A, no shall statements	None.	Yes	N/A
8.2.1 KAS1 Family Prerequisites	None.	None.	Yes	N/A
8.2.2 KAS1-basic	None.	None.	Yes	N/A
8.2.3 KAS1 Key Confirmation	None.	None.	Yes	N/A
8.2.4 KAS1 Security Properties	N/A, no shall statements	None.	Yes	N/A
8.3 The KAS2 Family	N/A, no shall statements	None.	Yes	N/A
8.3.1 KAS2 Family Prerequisites	None.	None.	Yes	N/A
8.3.2 KAS2-basic	None.	"the observable behavior of the key-agreement process **should not** reveal partial information about the shared secret Z."	Yes	N/A
8.3.3 KAS2 Key Confirmation	None.	None.	Yes	N/A
8.3.4 KAS2 Security Properties	N/A, no shall statements	None.	Yes	N/A
9 IFC based Key Transport Schemes	None.	None.	Yes	N/A
9.1 Additional Input	None.	None.	Yes	N/A
9.2 KTS-OAEP Family: Key Transport Using	N/A, no shall statements	None.	Yes	N/A

Section	Shall/Shall Not Statement(s)	Should (Not) Statements[2]	TOE Compliant?	Rationale
RSA-OAEP				
9.2.1 KTS-OAEP Family Prerequisites	None.	None.	Yes	N/A
9.2.2 Common components	N/A, no shall statements	None.	Yes	N/A
9.2.3 KTS-OAEP-basic	None.	None.	Yes	N/A
9.2.4 KTS-OAEP Key Confirmation	None.	None.	Yes	N/A
9.2.5 KTS-OAEP Security Properties	N/A, no shall statements	None.	Yes	N/A
9.3 KTS-KEM-KWS Family: Key Transport using RSA-KEM-KWS	N/A, no shall statements	None.	Yes	N/A
9.3.1 KTS-KEM-KWS Family Prerequisites	None.	None.	Yes	N/A
9.3.2 Common Components of the KTS-KEM-KWS Schemes	N/A, no shall statements	None.	Yes	N/A
9.3.3 KTS-KEM-KWS-basic	None.	None.	Yes	N/A
9.3.4 KTS-KEM-KWS Key Confirmation	None.	None.	Yes	N/A
9.3.5 KTS-KEM-KWS Security Properties	N/A, no shall statements	None.	Yes	N/A

ANNEX B: REFERENCES

The following documentation was used to prepare this ST:

Table 27: References

[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated September 2006, version 3.1, Revision 4, CCMB-2012-09-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated September 2007, version 3.1, Revision 4, CCMB--2012-09-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated September 2007, version 3.1, Revision 4, CCMB-2012-09-003
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated September 2007, version 3.1, Revision 4, CCMB-2012-09-004

[NDPP]	US Government, Security Requirements for Network Devices (pp_nd_v1.1), version 1.01, dated 8 June 2012
[FIPS 140-2]	FIPS PUB 140-2; Federal Information Processing Standards Publication Security Requirements for Cryptographic Modules
[FIPS 180-3]	FIPS Pub 180-3, Secure Hash Standard
[FIPS 186-2]	FIPS PUB 186-2, Digital Signature Standard
[FIPS 186-3]	FIPS PUB 186-3, Digital Signature Standard
[FIPS 198-1]	FIPS Pub 198-1, The Keyed-Hash Message Authentication Code
[NIST SP 800-56A]	NIST Special Publication 800-56A Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised), March 2007
[NIST SP 800-56B]	NIST Special Publication 800-56B Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography (Revised), August 2009
[ANSI X9.31]	FIPS Pub 140-2 Annex C: X9.31 Appendix 2.4: NIST-Recommended Random Number Generator Based on ANSI X9.31 Appendix A.2.4 Using the 3-Key Triple DES and AES Algorithms
[OPE]	Cisco Identity Services Engine (ISE) Common Criteria Operational User Guidance and Preparative Procedures, Version 1.0, January 2014