Cisco Identity Services Engine (ISE) V3.3 Security Target

Version 1.0

April 21, 2025

Table of Contents

List of Tables

List of Figures

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) v3.3. 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, Security 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 2.

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) V3.3 Security Target
ST Version	1.0
Publication Date	April 21, 2025
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Identity Services Engine (ISE) V3.3
TOE Models	• ISE 3700 series: SNS-3715, SNS-3755 and SNS-3795 • ISE-VM on ESXi 7.0 running on Cisco UCS C220-M6S
TOE Software Version	ISE v3.3, running on Cisco Application Deployment Engine (ADE) Release 3.3 operating system (ADE-OS)
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
AES	Advanced Encryption Standard
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CLI	Command Line Interface
CM	Configuration Management
DH	Diffie-Hellman
FIPS	Federal Information Processing Standard
HMAC	Hashed Message Authentication Code
HTTPS	Hyper-Text Transport Protocol Secure
IP	Internet Protocol
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
NAC	Network Access Control
NAS	Network Access Server
OS	Operating System
OSP	Organizational Security Policies
PP	Protection Profile
cPP	Collaborative Protection Profile for Network Devices (NDcPP)
RNG	Random Number Generator
SGA	Security Group Access
SGT	Security Group tags
SNS	Secure Network Server
SSHv2	Secure Shell (version 2)
ST	Security Target
TCP	Transport Control Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function
VPN	Virtual Private Network
WLC	Wireless LAN Controller

1.3 Terminology

The following terms are used in this Security Target:

Table 3: TOE-related 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, printers, or any other device supported on the ISE network.

Term	Definition
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.
Role	The role identity determines if the TOE is a standalone, primary, or secondary node.
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 the following options: Cisco Identity Services Engine Appliances SNS-3715, SNS-3755, SNS-3795 and Cisco Identity Services Engine Virtual Machine (ISE-VM) on ESXi 7.0 running on UCSC-C220-M6S.

1.4.1 TOE Product Type

The Cisco Identity Services Engine (ISE) 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: • Mozilla Firefox version 70 and later • Google Chrome version 78 and later • Microsoft Edge 115.x and later

Component	Required	Usage/Purpose Description for TOE performance
Network Access Server (NAS)	Yes	Also known as the RADIUS Authenticator, the Network Access Server is used during the 802.1X authentication exchange to relay the supplicant authentication to the Authentication Server. The 802.1X frames carry EAP authentication packets which are passed through to the RADIUS Authentication Server.
Clients	Yes	The network devices that are provided authentication services by ISE are referred to as clients
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.
OCSP Responder	Yes	OCSP is used to validate the revocation status of the certificates for session establishment.
Remote Authentication Store	No	The TOE supports local authentication or authentication via a remote authentication store, including LDAP and Active Directory.
NTP Server	No	The TOE supports communications with an NTP server. Connections to remote NTP servers are protected using SHA1, SHA256, SHA512 as the message digest algorithm(s).

1.5 TOE DESCRIPTION

This section provides an overview of the Cisco Identity Services Engine (ISE) v3.3 Target of Evaluation (TOE) and a brief description of the capabilities of the ISE product. ISE is a consolidated policy-based access control system that combines authentication, authorization, accounting (AAA) and guest management in one appliance. ISE v3.3 software runs on the Cisco Application Deployment Engine (ADE) Release 3.3 operating system (ADE-OS). ADE-OS is a Cisco-proprietary Red Hat Enterprise Linux based Operating system [RHEL v8.4 w/Linux kernel 4.18]. The TOE provides IPsec session capabilities to secure the channel between itself and the NAS.

Network access has evolved beyond just simple username and password verifications. Additional attributes related to users and their devices are used as decision criteria in determining authorized network access. Additionally, network service provisioning can be based on data such as the type of device accessing the network, including whether it is a corporate or personal device. Cisco ISE is a scalable solution that helps network administrators meet complex network access control demands by managing the many different operations that can place heavy loads on applications and servers, including:

- Authorization and authentication requests
- Queries to identity stores such as Active Directory and LDAP databases
- Device profiling and posture checking

- Enforcement actions to remove devices from the network
- Reporting

ISE delivers secure access control across wired, wireless, and VPN connections. ISE can reach deep into the network to deliver visibility into who and what are accessing resources. Through the device profiler feed service, ISE delivers automatic updates of Cisco's validated device profiles for various IP-enabled devices from multiple vendors which simplifies the task of keeping an up-to-date library of the newest IP enabled devices.
The Cisco Secure Network Server (SNS) is based on the Cisco UCS® C220 Rack Server and is configured specifically to support the Cisco Identity Services Engine (ISE) security application. The Secure Network Server supports these applications in three versions. The Cisco Secure Network Server 3715 is designed for small deployments. The Secure Network Servers 3755 and 3795 have several redundant components such as hard disks and power supplies, making it suitable for larger deployments that require highly reliable system configurations.
Apart from the SNS models described above, ISE is also available as a Vitual Machine running on ESXi 7.0 on UCSC-C220-M6S. Cisco ISE supports the following virtual environment platforms, but only the ESXi 7.0 environment is a part of the evaluated configuration:

- ESXi 7.0
- Microsoft Hyper-V on Microsoft Windows Server 2012 R2 and later
- KVM on RHEL 8.4

1.6 TOE Evaluated Configuration

The evaluated configuration of the TOE includes only one instance of ISE in a stand-alone deployment.

The following figure that shows a typical TOE deployment includes the following components:

- **TOE** – An instance of Cisco ISE (SNS appliance or ISE-VM)/node
- **Clients** – The network devices that are provided authentication services by ISE
- **Endpoints** – Devices through which the administrators can log in and manage the TOE.
- **Syslog Server** - The TOE can be configured to send syslog events to the syslog server.
- **Network Access Server** - The Network Access Server is used during the 802.1X authentication exchange as the RADIUS Authenticator to relay the supplicant authentication to the Authentication Server.
- **Remote Authentication Store** - The TOE can be configured to require local authentication and/or remote authentication via a remote authentication store
- **NTP Server** - The TOE supports communications with an NTP server to synchronize time.
- **OCSP Responder** - OCSP is used to validate the revocation status of the certificates for session establishment.

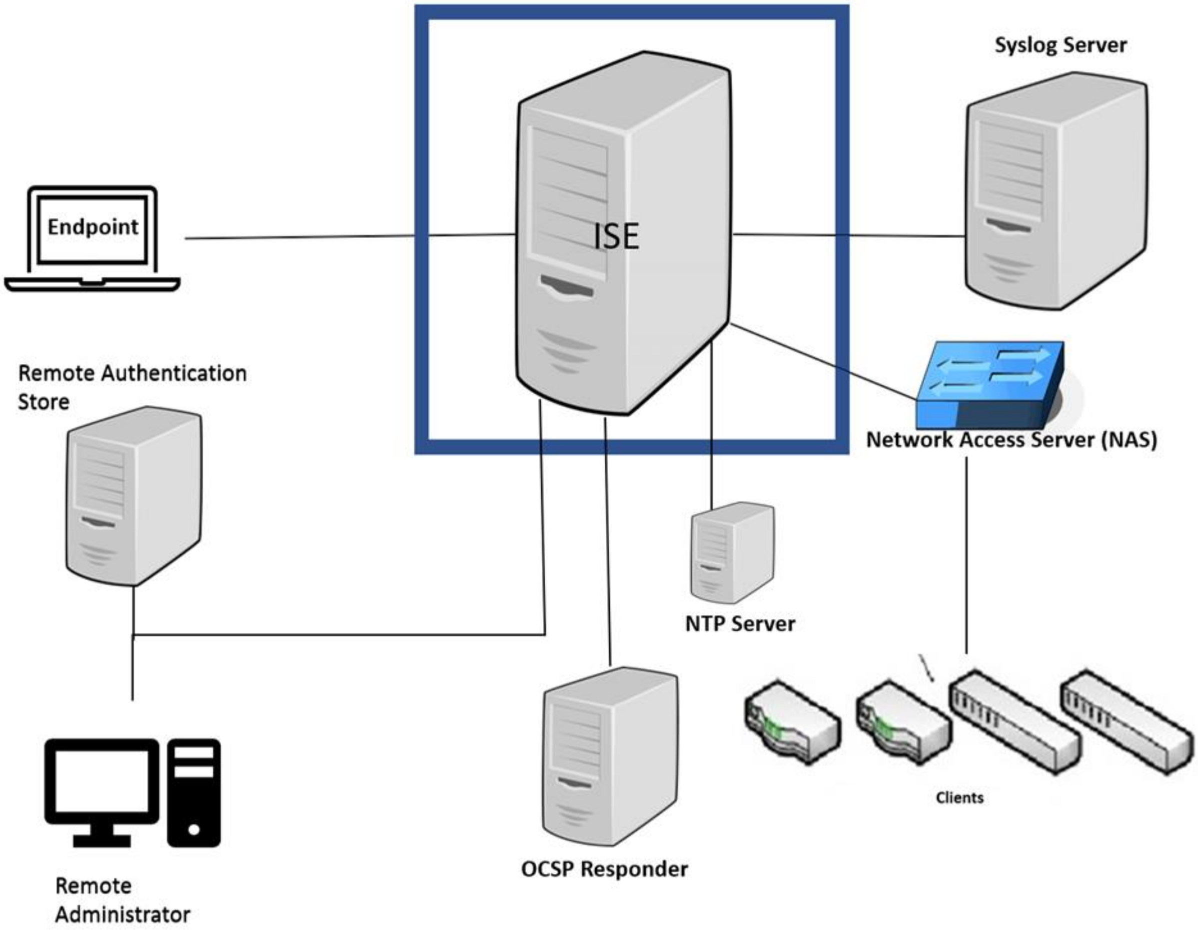

Figure 1: TOE Deployment

▭ - TOE Boundary

The evaluated configuration will include one ISE instance in a network. The evaluated configuration of the TOE includes network devices utilizing the ISE authentication, authorization and accounting (AAA) features, remote administrator, local administrative console and a remote authentication store. Both the remote administrator and local administrator console capabilities must be supported.

1.7 Physical Scope of the TOE

The Cisco ISE software includes the Cisco Application Deployment Engine (ADE) Release 3.3 operating system (ADE-OS). The Cisco ISE software run on a dedicated Cisco ISE 3700 Series appliances and on ESXi 7.0 running on Cisco UCS C220-M6S (UCSC-C220-M6S). All models include the same security functionality.

Table 5: TOE Models

Hardware Models	Cisco Identity Services Engine Appliance 3715 (SNS-3715)	Cisco Identity Services Engine Appliance 3755 (SNS-3755)	Cisco Identity Services Engine Appliance 3795 (SNS-3795)	Cisco Identity Services Engine – VM running on ESXi 7.0/UCSC-C220-M6S (ISE-VM)
Processors	Intel Xeon Silver 4310 (Ice Lake)	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Silver 4310 (Ice Lake)[1]
Memory	32GB	96 GB	256 GB	96 GB
Hard disk	1x600 Gb disk	4x600Gb disk	8x600Gb disk	4x600Gb disk
RAID	No	Yes (RAID 1+0)	Yes (RAID 1+0)	Yes (RAID 1+0)
Network interface	2 x 10GBase-T 4 x 10GE SFP (Intel X710)	2 x 10GBase-T 4 x 10GE SFP (Intel X710)	2 x 10GBase-T 4 x 10GE SFP (Intel X710)	Dual 10GBASE-T Ethernet ports (Intel x550)
Hypervisor	None	None	None	ESXi 7.0

[1] While tested on the Intel Xeon Silver 4310 (Ice Lake), any Intel Xeon processor with the Ice Lake microarchitecture may be used as part of the evaluated configuration with VMware ESXi 7.0

1.8 Logical Scope of the TOE

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

1. Security Audit
2. Cryptographic Support
3. Communications
4. Identification and Authentication
5. Security 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 Security Administrator, and other system events.

The TOE can store the generated audit data on itself, and it 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 Security Administrator, who has no access to edit them, only to copy or delete (clear) them. Audit records are protected from unauthorized modifications and deletions.

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. The previous audit records are overwritten when the allocated space for these records reaches the threshold.

1.8.2 Cryptographic Support

The TOE provides cryptography support for secure communications and protection of information. The cryptographic services provided by the TOE include: symmetric encryption and decryption using AES; asymmetric key generation; cryptographic key establishment using RSA-based and ECDSA key establishment schemes and DH key establishment; digital signature using RSA and ECDSA; cryptographic hashing using SHA1 (and other sizes); random bit generation using

DRBG and keyed-hash message authentication using HMAC-SHA (multiple key sizes). ISE uses the CiscoSSL FIPS Object Module (FOM) Cryptographic Implementation as its cryptographic module. The TOE implements the secure protocols – SSH, TLS/HTTPS on the server side and TLS on the client side and IPsec session capabilities to secure the channel between the TOE and NAS. The algorithm certificate references are listed in Section 7.1.

1.8.3 Communications

The TOE has the ability to validate the NAS and prevent it from being spoofed. It receives the transmitted Access-Request and identifies where it's sent from. The TOE is able to validate the authenticity of the NAS by verifying the Message Authenticator that is computed in part using a shared secret known to both the NAS and the TOE as defined in RFC 3579. It then returns a valid response to the NAS upon receipt of an Access-Request. The response contains the necessary information to the recipient of that message that identifies the TOE as the valid recipient of the original Access-Request and the Access-Request that elicited the response 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 other than the display of the warning banner. Once a user attempts to access the management functionality of the TOE, the TOE prompts the user for a user name and password for remote password-based authentication. 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 X.509 certificate-based authentication to the administration application, a 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 Administrator presents the correct identification and authentication credentials will access to the TOE functionality be granted. The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS/HTTPS connections. The revocation status of the certificates can be validated by the TOE using OCSP.

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:

- Administer the TOE locally and remotely

- Configure the access banner
- Configure the cryptographic services
- Update the TOE and verify the updates using digital signature capability prior to installing those updates
- Specify the time limits of session inactivity

All of these management functions are restricted to the Security Administrator of the TOE, which covers all administrator roles (see table for FMT_SMR.2 in Section 6.1). The Security Administrators of the TOE are individuals who manage specific type of administrative tasks. The Security 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 like system-level configuration in EXEC mode and other configuration tasks in configuration mode and to generate operational logs for troubleshooting. This interface can be used remotely over SSHv2.

1.8.6 Protection of the TSF

The TOE can terminate inactive sessions after a Security Administrator configurable time-period. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session. The TOE provides protection of TSF data (authentication data and cryptographic keys). In addition, the TOE internally maintains the date and time. This date and time is used as the time stamp that is applied to TOE generated audit records. This time can be set manually and via NTP. 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 digital signature mechanism to confirm the integrity of the product.

1.8.7 TOE Access

The TOE can terminate inactive sessions after a Security 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 UI 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. The TOE implements IPsec session capabilities to secure the channel between the TOE and NAS.

1.9 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 6: Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS mode of operation	This mode of operation includes non-FIPS allowed operations.
Guest Management	Not within the scope of the evaluation
The device profiler feed service	Not within the scope of the evaluation
Virtual environment Microsoft Hyper-V on Microsoft Windows Server 2012 R2 for ISE-VM	Only ESXi 7.0 virtual environment will be tested
Virtual environment KVM on RHEL 7.3 for ISE-VM	Only ESXi 7.0 virtual environment will be tested

These services will be disabled by configuration. The exclusion of this functionality does not affect compliance to the collaborative Protection Profile for Network Devices Version 2.2e and mod_authsvr_v1.0.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 5, dated: April 2017. For a listing of Assurance Requirements claimed see section 5.5.

The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profiles as listed in the table below:

Table 7: Protection Profiles

Protection Profile	Version	Date
PP-Configuration for Network Devices and Authentication Servers	1.0	September 6, 2023
The PP-Configuration includes the following components:		
• Base-PP: Collaborative Protection Profile for Network Devices, (CPP_ND_V2.2E)	2.2e	23 March 2020
• PP-Module: PP-Module for Authentication Servers (MOD_AUTHSVR_V1.0)	1.0	25 January 2023

The ST is also compliant to the Technical Decisions listed in the table below –

Table 8: Technical Decisions

Technical Decision#	Technical Decision Name	Applicable?	Exclusion Rationale (if applicable)
Technical Decisions applicable to PP-Module for Authentication Servers v1.0			
TD0833	*Aligning MOD_AUTHSVR 1.0 with NDcPP 3.0E*	*No*	*The evaluation doesn't claim conformance to NDcPP 3.0E*
TD0820	Clarification for Authentication Requests in FCO_NRO.1	Yes	
TD0818	Clarification to FCS_RADIUS_EXT.1 testing	Yes	
Technical Decisions applicable to Network Device Collaborative Protection Profile (NDcPP) v2.2e			

TD0800	Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	Yes	
TD0792	NIT Technical Decision: FIA_PMG_EXT.1 - TSS EA not in line with SFR	Yes	
TD0790	NIT Technical Decision: Clarification Required for testing IPv6	Yes	
TD0738	NIT Technical Decision for Link to Allowed-With List	Yes	
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	Yes	
TD0639	NIT Technical Decision for Clarification for NTP MAC Keys	Yes	
TD0638	NIT Technical Decision for Key Pair Generation for Authentication	Yes	
TD0636	*NIT Technical Decision for Clarification of Public Key User Authentication for SSH*	*No*	*FCS_SSHC_EXT.1 is not selected*
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
TD0631	NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
TD0592	NIT Technical Decision for Local Storage of Audit Records	Yes	
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	Yes	
TD0581	NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
TD0580	NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
TD0569	NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	Yes	
TD0564	NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
TD0563	NiT Technical Decision for Clarification of audit date information	Yes	
TD0556	NIT Technical Decision for RFC 5077 question	Yes	
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	

TD0546	*NIT Technical Decision for DTLS - clarification of Application Note 63*	*No*	*FCS_DTLSC_EXT.1 is not selected*
TD0537	NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	Yes	
TD0536	NIT Technical Decision for Update Verification Inconsistency	Yes	
TD0528	NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	Yes	
TD0527	Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the:

- collaborative Protection Profile for Network Devices (cpp_nd_v2.2e)
- PP-Module: PP-Module for Authentication Servers Version 1.0, (mod_authsvr_v1.0)

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organizational Security Policies included in the Security Target represent the Assumptions, Threats, and Organizational Security Policies specified in the collaborative Protection Profile for Network Devices, Version 2.2e and PP-Module for Authentication Servers Version 1.0, for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in cpp_nd_v2.2e and mod_authsvr_v1.0, for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Objectives are included in the Security Target.

2.3.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in cpp_nd_v2.2e and mod_authsvr_v1.0, for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Requirements are included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in cpp_nd_v2.2e and mod_authsvr_v1.0.

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 9: TOE Assumptions

Assumption	Assumption Definition
Reproduced from cpp_nd_v2.2e	
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.

Assumption	Assumption Definition
Reproduced from cpp_nd_v2.2e	
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ADMINSTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_ADMINISTRATOR	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.

Assumption	Assumption Definition
Reproduced from cpp_nd_v2.2e	
A.VS_REGULAR_UPDATES	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATON	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.
Reproduced from mod_authsvr_v1.0 (Assumptions made on the Operational Environment (OE)	
A.RP_FEDERATION	It is assumed that the TOE is federated with one or more relying parties that transmit authentication requests to it.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment.

Table 10: Threats

Threat	Threat Definition
Reproduced from cpp_nd_v2.2e	
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	Threat agents may attempt to gain administrator access to the network device by nefarious means such as masquerading as an administrator to the device, masquerading as the device to an administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between network devices. Successfully gaining administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.

Threat	Threat Definition
Reproduced from cpp_nd_v2.2e	
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_COMMUNICATION_CHANNELS	Threat agents may attempt to target network devices that do not use standardized secure tunneling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the network device itself.
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the network device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the network device without administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.

Threat	Threat Definition
Reproduced from cpp_nd_v2.2e	
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other network devices.
T.SECURITY_FUNCTIONALITY_FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.
Reproduced from mod_authsvr_v1.0	
T.FALSE_ENDPOINTS	A malicious actor may falsely impersonate the TOE or a federated relying party in order to cause the TOE to operate in an insecure manner or to extract security-relevant, or sensitive user data from the TOE or its Operational Environment.
T.INVALID_USERS	A malicious user may supply incorrect or insufficient credential data or an otherwise invalid authentication request that is approved or ignored by the TSF such that protected resources are subject to unauthenticated access.

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 11: Organizational Security Policies

Policy Name	Policy Definition
Reproduced from cpp_nd_v2.2e	
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.
Reproduced from mod_authsvr_v1.0	
P.AUTH_POLICY	The organization defines, for each protected resource, an authentication policy that specifies the authenticators that must be provided to access a given resource.

4 SECURITY OBJECTIVES

This section 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.

4.1 Security Objectives for the TOE

The following table - Table 12: 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 12: Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
Reproduced from mod_authsvr_v1.0	
O.AUTHORIZED_USE	The TOE shall provide mechanisms that prevent and detect its unauthorized use.
O.SECURITY_ASSOCIATION	The TOE shall provide the information to the relying party to enable it to verify that the claimant has possession of an authentication key.
O.TRUSTED_RP	The TOE shall provide mechanisms to authenticate itself to a federated RP and authenticate a federated RP before providing an identity assertion.
O.USER_AUTH	The TOE shall provide a mechanism to assess authentication requests and respond with an authentication assertion based on data that is supplied in the request.

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 Protection Profile non-TOE 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 13: Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
Reproduced from cpp_nd_v2.2e	
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_ SECURE	The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove

Environment Security Objective	IT Environment Security Objective Definition
Reproduced from cpp_nd_v2.2e	
	unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualization features such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.
Reproduced from mod_authsvr_v1.0	
OE.RP_FEDERATION	The TOE will be deployed in such a manner that it is federated with one or more relying parties that transmit authentication requests to it.
OE.REQUIRE_AUTH	The operational environment will protect assets in a manner that requires authentication commensurate with the sensitivity of the assets.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, dated: April 2017* 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:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated with **bold text** and ~~strikethroughs~~, if necessary;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with *italicized and underlined text*;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/"..
- Where operations were completed in the NDcPPv2.2e and mod_authsvr_v1.0itself, the formatting used there has been retained.
- Formatting used in NDcPPv2.2e and mod_authsvr_v1.0 that is inconsistent with the listed conventions has not being retained in the ST.

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations and iterations matches the formatting specified within the PP and the PP modules themselves.

5.2 TOE Security Functional Requirements

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

Table 14: Security Functional Requirements

Requirement Class Name	Component Name and Identification
Reproduced from cpp_nd_v2.2e	
FAU: Security Audit	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User identity association
	FAU_STG.1 Protected Audit Trail Storage
	FAU_STG_EXT.1 Protected Audit Event Storage
FCS: Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation
	FCS_CKM.2 Cryptographic Key Establishment

Requirement Class Name	Component Name and Identification
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1 HTTPS Protocol
	FCS_IPSEC_EXT.1 IPsec Protocol
	FCS_NTP_EXT.1 NTP Protocol
	FCS_RBG_EXT.1 Random Bit Generation
	FCS_SSHS_EXT.1 SSH Server Protocol
	FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication
	FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication
	FCS_TLSS_EXT.1(1) TLS Server Protocol without Mutual Authentication - WebUI
	FCS_TLSS_EXT.1(2) TLS Server Protocol without Mutual Authentication – EAP-TLS
	FCS_TLSS_EXT.2 TLS Server Support for Mutual Authentication
FIA: Identification and Authentication	FIA_AFL.1 Authentication Failure Management
	FIA_PMG_EXT.1 Password Management
	FIA_UIA_EXT.1 User Identification and Authentication
	FIA_UAU_EXT.2 Password-based Authentication Mechanism
	FIA_UAU.7 Protected Authentication Feedback
	FIA_X509_EXT.1/Rev X.509 Certificate Validation
	FIA_X509_EXT.2 X.509 Certificate Authentication
	FIA_X509_EXT.3 X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate Management of security functions behaviour
	FMT_MOF.1/Functions Management of security functions behaviour
	FMT_MOF.1/Services Management of security functions behaviour
	FMT_MTD.1/CoreData Management of TSF data
	FMT_MTD.1/CryptoKeys Management of TSF data
	FMT_SMF.1 Specification of Management Functions
	FMT_SMR.2 Restrictions on Security Roles

Requirement Class Name	Component Name and Identification
FPT: Protection of the TSF	FPT_SKP_EXT.1 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_APW_EXT.1 Protection of Administrator Passwords
	FPT_TST_EXT.1 TSF Testing (Extended)
	FPT_TUD_EXT.1 Trusted update
	FPT_STM_EXT.1 Reliable Time Stamps
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/Admin Trusted Path
Reproduced from mod_authsvr_v1.0	
FAU: Security Audit	FAU_GEN.1/AuthSvr Audit Data Generation (Authentication Server)
FCO: Communications	FCO_NRO.1 Selective Proof of Origin
	FCO_NRR.1 Selective Proof of Receipt
FCS: Cryptographic Support	FCS_CKM.3 Cryptographic Key Access
	FCS_EAPTLS_EXT.1 EAP-TLS Protocol
	FCS_RADIUS_EXT.1 Authentication Protocol
	FCS_STG_EXT.1 Cryptographic Key Storage
FIA: Identification and Authentication	FIA_AFL.1/AuthSvr Authentication Failure Handling (Claimant)
	FIA_UAU.6 Re-Authenticating
	FIA_X509_EXT.1/AuthSvr X.509 Certificate Validation (Claimant)
FMT: Security management	FMT_SMF.1/AuthSvr Specification of Management Functions (Authentication Server)
FTA: TOE Access	FTA_TSE.1 TOE Session Establishment
FTP: Trusted path/channels	FTP_ITC.1/NAS Inter-TSF Trusted Channel (Relying Party Communications)

5.3 SFRs Drawn from NDcPP

5.3.1 Security Audit (FAU)

5.3.1.1 FAU_GEN.1 Audit Data Generation

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

a) Start-up and shutdown of the audit functions;

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

c) *All administrative actions comprising:*

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

d) Specifically defined auditable events listed in Table 15.

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table **15***.

Table 15: Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA	Reason for failure.
FCS_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity if new/removed time server
FCS_RBG_EXT.1	None.	None.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSC_EXT.2	None	None
FCS_TLSS_EXT.1(1)	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1(2)	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.2	Failure to authenticate the client	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).

Requirement	Auditable Events	Additional Audit Record Contents
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store.	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store.
FIA_X509_EXT.2	None.	None
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Functions	None.	None.
FMT_MOF.1/Services	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_STM_EXT.1	Discontinuous changes to time – either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1 (if "terminate the sessions" is selected)	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failures of the trusted path functions.	None.

5.3.1.2 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.

5.3.1.3 FAU_STG.1 Protected audit trail storage

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

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

5.3.1.4 FAU_STG_EXT.1 Protected Audit Event Storage

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

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. In addition [

- *TOE shall consist of a single standalone component that stores audit data locally,*
  ]

FAU_STG_EXT.1.3 The TSF shall [*overwrite previous audit records according to the following rule: [since the storage period of logs is configurable, the oldest records are overwritten first]*] when the local storage space for audit data is full.

5.3.2 Cryptographic Support (FCS)

5.3.2.1 FCS_CKM.1 Cryptographic Key Generation (Refinement)

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

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4*
- *FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1*
- *FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and* [*RFC 3526*]

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

5.3.2.2 FCS_CKM.2 Cryptographic Key Establishment

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"*
- *Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"*
- *FFC Schemes using "safe-prime" groups that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 3526].*

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

5.3.2.3 FCS_CKM.4 Cryptographic key Destruction

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

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

that meets the following: *No Standard*.

5.3.2.4 FCS_COP.1/DataEncryption Cryptographic operation (AES Data Encryption/Decryption)

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [*CBC, CTR, GCM*] *mode* and cryptographic key sizes [*128 bits, 192 bits, 256 bits*] that meet the following: *AES as specified in ISO 18033-3*, [*CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772*].

Application Note: The claim of key size of 192 bits, applies only to CBC mode.

5.3.2.5 FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)

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

- *RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 3072 bits, 4096 bits]*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits, 521 bits]*

]

that meet the following: [

- *For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,*
- *For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4*

].

5.3.2.6 FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [*SHA-1, SHA-256, SHA-384, SHA-512*] ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ and **message digest sizes** [***160, 256, 384, 512***] **bits** that meet the following: *ISO/IEC 10118-3:2004*.

5.3.2.7 FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*] and cryptographic key sizes [*160, 256, 384, 512 bits*] **and message digest sizes [*160, 256, 384, 512*] bits** that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

5.3.2.8 FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

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

5.3.2.9 FCS_IPSEC_EXT.1 Extended: IPSEC

FCS_IPSEC_EXT.1.1 The TSF shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2 The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched and discards it.

FCS_IPSEC_EXT.1.3 The TSF shall implement [*transport mode, tunnel mode*].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [*AES-CBC-128 (RFC 3602), AES-CBC-192 (RFC 3602), AES-CBC-256 (RFC 3602)*] together with a Secure Hash Algorithm (SHA)-based HMAC [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*]

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- *IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers], and [no other RFCs for hash functions].*
- *IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23)], and [RFC 4868 for hash functions]*

]

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [*IKEv1, IKEv2*] protocol uses the cryptographic algorithms [*AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602)*].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

- *IKEv1 Phase 1 SA lifetimes can be configured by a Security Administrator based on*
  [
  - *length of time, where the time values can be configured within [1-24] hours;*

  ];
- *IKEv2 SA lifetimes can be configured by a Security Administrator based on*
  [
  - *length of time, where the time values can be configured within [1-24] hours;*

  ]

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

- *IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on*
  [
  - *length of time, where the time values can be configured within [1-8] hours;*

  ];
- *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on*
  [
  - *length of time, where the time values can be configured within [1-8] hours;*

  ]

].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*320 (for DH Group 14), 256 (for DH Group 19), 384 (for DH Group 20)*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [*IKEv1, IKEv2*] exchanges of length [

- *according to the security strength associated with the negotiated Diffie-Hellman group;*
  ].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s)
[

- [*14 (2048-bit MODP)] according to RFC 3526;*
- [*19 (256-bit Random ECP), 20 (384-bit Random ECP)] according to RFC 5114.*

]

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [*IKEv1 Phase 1, IKEv2 IKE_SA*] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [*IKEv1 Phase 2, IKEv2 CHILD_SA*] connection.

FCS_IPSEC_EXT.1.13 The TSF shall ensure that all IKE protocols perform peer authentication using [*RSA, ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [*no other method*].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [*SAN: IP address*] and [*no other reference identifier types*].

5.3.2.10 FCS_NTP_EXT.1 NTP Protocol

FCS_NTP_EXT.1.1 The TSF shall use only the following NTP version(s) [*NTP v4 (RFC 5905)*].
FCS_NTP_EXT.1.2 The TSF shall update its system time using [

- Authentication using [*SHA1*] as the message digest algorithm(s);

FCS_NTP_EXT.1.3 The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.
FCS_NTP_EXT.1.4 The TSF shall support configuration of at least three (3) NTP time sources in the Operational Environment.

5.3.2.11 FCS_RBG_EXT.1 Cryptographic operation (random bit generation)

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

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

5.3.2.12 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [*5656, 6668,* 8332].

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

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*262126*] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [*aes128-ctr, aes256-ctr*].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [*rsa-sha2-256, rsa-sha2-512*] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [*hmac-sha1, hmac-sha2-256, hmac-sha2-512*] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [*diffie-hellman-group14-sha1, ecdh-sha2-nistp256*] and [*ecdh-sha2-nistp384, ecdh-sha2-nistp521*] are the only allowed key exchange method used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

5.3.2.13 FCS_TLSC_EXT.1 TLS Client Protocol without Mutual Authentication

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

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*

- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

]

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches [*the reference identifier per RFC 6125 section 6, IPv4 address in SAN, IPv6 address in the SAN*].

FCS_TLSC_EXT.1.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- *Not implement any administrator override mechanism*

]

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

5.3.2.14 FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication

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

5.3.2.15 FCS_TLSS_EXT.1(1) TLS Server Protocol Without Mutual Authentication - WebUI

FCS_TLSS_EXT.1.1(1) The TSF shall implement [*TLS 1.2 (RFC 5246)*] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

].

FCS_TLSS_EXT.1.2(1) The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [*TLS 1.1*].

FCS_TLSS_EXT.1.3(1) The TSF shall perform key establishment for TLS using [*RSA with key size* [*2048 bits, 3072 bits, 4096 bits*]*; Diffie-Hellman parameters with size* [*2048 bits*]*; ECDHE curves* [*secp256r1*] *and no other curves*]].

FCS_TLSS_EXT.1.4(1) The TSF shall support [*session resumption based on session IDs according to RFC 4346 (TLS1.1) or RFC 5246 (TLS1.2), session resumption based on session tickets according to RFC 5077*]

5.3.2.16 FCS_TLSS_EXT.1(2) TLS Server Protocol Without Mutual Authentication – EAP-TLS

FCS_TLSS_EXT.1.1(2) The TSF shall implement [*TLS 1.2 (RFC 5246)*] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites: [

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*

- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*

].

FCS_TLSS_EXT.1.2(2) The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [*TLS 1.1*].

FCS_TLSS_EXT.1.3(2) The TSF shall perform key establishment for TLS using [*RSA with key size* [*2048 bits, 3072 bits, 4096 bits*]*; ECDHE curves* [*secp256r1, secp384r1, secp521r1*] *and no other curves*]].

FCS_TLSS_EXT.1.4(2) The TSF shall support [*session resumption based on session IDs according to RFC 4346 (TLS1.1) or RFC 5246 (TLS1.2)*]

5.3.2.17 FCS_TLSS_EXT.2 TLS Server Protocol with mutual authentication

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

FCS_TLSS_EXT.2.2 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also [

- *Not implement any administrator override mechanism*

]

FCS_TLSS_EXT.2.3 The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifiers according to RFC6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.

5.3.3 Identification and Authentication (FIA)

5.3.3.1 FIA_AFL.1 Authentication Failure Management

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

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

5.3.3.2 FIA_PMG_EXT.1 Password Management

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [_“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”_];

b) Minimum password length shall be configurable to between *[6] and [127] characters.*

5.3.3.3 FIA_UIA_EXT.1 User Identification and Authentication

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

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

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

5.3.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

5.3.3.5 FIA_UAU.7 Protected authentication feedback

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

5.3.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*the Online Certificate Status Protocol (OCSP) as specified in RFC 6960*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.3.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*IPsec, TLS*], and [*no additional uses*].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [*allow the administrator to choose whether to accept the certificate in these cases*].

5.3.3.8 FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.3.4 Security Management (FMT)

5.3.4.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

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

5.3.4.2 FMT_MOF.1/Functions Management of security functions behaviour

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

5.3.4.3 FMT_MOF.1/Services Management of security functions behavior

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

5.3.4.4 FMT_MTD.1/CoreData Management of TSF data

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

5.3.4.5 FMT_MTD.1/CryptoKeys Management of TSF data

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

5.3.4.6 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [digital signature, hash comparison] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*

- [
    - *Ability to start and stop services;*
    - *Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);*
    - *Ability to manage the cryptographic keys;*
    - *Ability to configure the cryptographic functionality;*
    - *Ability to re-enable an Administrator account;*
    - *Ability to configure the lifetime for IPsec SAs;*
    - *Ability to set the time which is used for time-stamps;*
    - *Ability to configure NTP;*
    - *Ability to configure the reference identifier for the peer;*
    - *Ability to manage the trusted public keys database;*
    - *Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;*
    - *Ability to import X.509v3 certificates to the TOE's trust store;]*

5.3.4.7 FMT_SMR.2 Restrictions on Security roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator.*

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.3.5 Protection of the TSF (FPT)

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

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

5.3.5.2 FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

5.3.5.3 FPT_TST_EXT.1 TSF Testing (Extended)

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

- *AES Known Answer Test*
- *AES-GCM Known Answer Test (Separate encrypt and decrypt)*
- *FIPS 186-4 ECDSA Sign/Verify Test*
- *ECC CDH Known Answer Test*
- *RSA Signature Known Answer Test (both signature/verification)*
- *DRBG Known Answer Test*
- *HMAC Known Answer Test*
- *SHA-1/256/384/512 Known Answer Test*
- *Software Integrity Test*

].

5.3.5.4 FPT_TUD_EXT.1 Trusted update

FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [*no other TOE firmware/software version*].

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

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

5.3.5.5 FPT_STM_EXT.1 Reliable time stamps

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

FPT_STM_EXT.1.2 The TSF shall [*allow the Security Administrator to set the time, synchronise time with an NTP server*].

5.3.6 TOE Access (FTA)

5.3.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [

- *terminate the session*]

after a Security Administrator-specified time period of inactivity.

5.3.6.2 FTA_SSL.3 TSF-initiated Termination

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

5.3.6.3 FTA_SSL.4 User-initiated Termination

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

5.3.6.4 FTA_TAB.1 Default TOE Access Banners

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

5.3.7 Trusted Path/Channel (FTP)

5.3.7.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall **be capable of using [*TLS*] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server,** [***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*].

5.3.7.2 FTP_TRP.1/Admin Trusted path

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

FTP_TRP.1.2/Admin The TSF shall permit remote **administrators** to initiate communication via the trusted path.

FTP_TRP.1.3/Admin The TSF shall require the use of the trusted path for *initial administrator authentication and all remote administrative actions*.

5.4 SFRs Drawn from mod_authsvr_v1.0

5.4.1 Security Audit (FAU)

5.4.1.1 FAU_GEN.1/AuthSvr Audit Data Generation (Authentication Server)

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

a) Start-up and shutdown of the audit functions;
b) All auditable events for the [*not specified*] level of audit; and
c) [*Auditable events listed in the Auditable Events table* (
Table 16)

Table 16: Auditable Events - Authentication Server

Requirement	Auditable Events	Additional Audit Record Contents
FCO_NRO.1	Claimant request for which the TOE does not have credential verification data	Identity of the claimant
FCO_NRR.1	None.	None.
FCS_CKM.3	None.	None.
FCS_EAPTLS_EXT.1	Protocol Failures Successful and failed authentication of claimant	If failure occurs, record a descriptive reason for the failure Identifier of claimant
FCS_RADIUS_EXT.1	Protocol Failures Success/failure of authentication	If failure occurs, record a descriptive reason for the failure None

FCS_STG_EXT.1	None.	None.
FIA_AFL.1/AuthSvr	The reaching of the threshold for the unsuccessful authentication attempts Disabling an account due to the threshold being reached	The claimed identity of the entity attempting to authenticate or the IP where the attempts originated
FIA_X509_EXT.1/AuthSvr	Certificate validation failure	Reason for failure
FIA_UAU.6	All use of the authentication mechanism	Origin of the attempt (e.g., IP address)
FMT_SMF.1/AuthSvr	All management actions	Identifier of initiator
FTA_TSE.1	Denial of session establishment due to the session establishment mechanism	Reason for denial, origin of establishment attempt
FTP_ITC.1/NAS	Initiation of the trusted channel Termination of the trusted channel Failure of the trusted channel functions	Identification of the initiator Identification of the initiator Target of failed trusted channels establishment attempt

FAU_GEN.1.2/AuthSvr 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-**Module**/ST, *information specified in column three of Table 16.*

5.4.2 Communications (FCO)

5.4.2.1 FCO_NRO.1 Selective Proof of Origin

FCO_NRO.1.1 The TSF shall be able to generate evidence of origin for transmitted [*identity authentication assertions*, [no other data]] at the request of the [relying party, *[no other entities]*].

FCO_NRO.1.2 The TSF shall be able to relate the [*authenticator*] of the originator of the information, and the [*authentication request*] of the information to which the evidence applies.

FCO_NRO.1.3 The TSF shall provide a capability to verify the evidence of origin of information to [*recipient*] given [*an authenticated channel is established with a trusted relying party*].

5.4.2.2 FCO_NRR.1 Selective Proof of Receipt

FCO_NRR.1.1 The TSF shall be able to generate evidence of receipt for received [*authentication requests, [authentication responses and queries]*] at the request of the [*originator*].

FCO_NRR.1.2 The TSF shall be able to relate the [*claimant identifier and claimant authenticators*] of the recipient of the information, and the [*identity assertion, information requests, and error responses*] of the information to which the evidence applies.

FCO_NRR.1.3 The TSF shall provide a capability to verify the evidence of receipt of information to [*originator*] given [*establishment of a mutually authenticated channel with a trusted relying party*].

5.4.3 Cryptographic Support (FCS)

5.4.3.1 FCS_CKM.3 Cryptographic Key Access

FCS_CKM.3.1 The TSF shall perform [*access control for persistent private and secret keys and critical security parameters required by this PP-Module*] in accordance with a specified cryptographic key access method [*ensuring only authorized security functionality can access plaintext keys or critical security parameters*] that meets the following: [*keys and critical security parameters are not exportable in plaintext and keys and critical security parameters are not viewable in plaintext*].

5.4.3.2 FCS_EAPTLS_EXT.1 EAP-TLS Protocol

FCS_EAPTLS_EXT.1.1 The TSF shall implement [*EAP-TLS as specified in RFC 5216*] as updated by RFC 8996 with [*TLS*] implemented using mutual authentication in accordance with [*FCS_TLSS_EXT.1(2) and FCS_TLSS_EXT.2*].

FCS_EAPTLS_EXT.1.2 The TSF shall generate random values used in the [*EAP-TLS*] exchange using the RBG specified in FCS_RBG_EXT.1.

FCS_EAPTLS_EXT.1.3 The TSF shall support claimant authentication using certificates and [*no other methods*].

FCS_EAPTLS_EXT.1.4 The TSF shall not forward an EAP-Success response to the relying party if the client certificate is not valid according to FIA_X509_EXT.1/AuthSvr, if the [*TLS*] session is not established, or if any of [*no other authenticator*] required by the authentication policy are not provided or if any of the required authenticators presented in the authentication request is not valid.

5.4.3.3 FCS_RADIUS_EXT.1 Authentication Protocol

FCS_ RADIUS_EXT.1.1 The TSF shall implement the [*RADIUS protocol as specified in RFC 2865*] for communication of identity and authentication information with a relying party.

FCS_ RADIUS_EXT.1.2 The TSF shall implement encapsulated EAP in accordance with FCS_EAPTLS_EXT.1.

FCS_ RADIUS_EXT.1.3 The TSF shall provide [*an encrypted value*] for a key held by the successfully authenticated claimant derived from the supported EAP mode and provided to the relying party in accordance with the protocol indicated in FCS_RADIUS_EXT.1.1.

5.4.3.4 FCS_STG_EXT.1 Cryptographic Key Storage

FCS_STG_EXT.1 Persistent private and secret keys shall be stored within the TSF [

- *within an isolated execution environment protected by a hardware key*

].

5.4.4 Identification and Authentication (FIA)

5.4.4.1 FIA_AFL.1/AuthSvr Authentication Failure Handling (Claimant)

FIA_AFL.1.1/AuthSvr The TSF shall detect when [*an administrator configurable positive integer* ***of successive***] unsuccessful authentication attempts occur related to [*claimants attempting to authenticate*].

FIA_AFL.1.2/AuthSvr When the defined number of unsuccessful authentication attempts has been [met], the TSF shall [*prevent the offending remote entity from successfully authenticating until [unlocking the claimant] is taken by a local Administrator*].

5.4.4.2 FIA_UAU.6 Re-Authenticating

FIA_UAU.6.1 The TSF shall re-authenticate the **administrative** user under the conditions [*when the user changes their password*, [*following TSF-initiated session locking*]]

5.4.4.3 FIA_X509_EXT.1/AuthSvr X.509 Certificate Validation (Claimant)

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

- RFC 5280 **version 3** certificate validation and certificate path validation **supporting [*a minimum path length of [three]*]**

- The certification path must terminate with a CA certificate **trusted by the TSF specifically for claimant authentication**.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of **each certificate in the certificate path [**
    - ***containing an OCSP provider in the AIA extension using the Online Certificate Status Protocol (OCSP) as specified in RFC 6960***

  ].
- The TSF shall validate the extendedKeyUsage field **is present and contains key usage values** according to the following rules:
  [
    - ***Client certificates associated with authenticated entities presented for [TLS] shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field***.
    - ***[***
        - ***OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.***

  ]
- **The TSF shall validate that each CA certificate in the certification path indicating a path length constraint in the basicConstraints extension does not have more than the specified number of subordinate CA certificates in the certification path from the end-entity certificate to the CA certificate indicating the constraint, not counting the CA certificate itself or any self-issued certificates in the certification path.**
- **The TSF shall process name constraints of type Directory Name and [*rfc822Name, dnsName, UPN Name (Other Name = id-ms-san-sc-logon-upn)*] by verifying that each name of a supported name type present in the end-entity certificate subject field or subjectAlternateName extension, is allowed in each CA certificate in the certification path, is not disallowed by any of the CA certificates in the certification path, and that each name type included in the end-entity certificate and constrained by a CA certificate in the certification path is processed.**
- **The TSF shall process the following certificate extensions: [**
    - ***Certificate Policy extension in accordance with RFC 5280 and [***
        - ***Policy mapping extension in accordance with RFC 5280***
        - ***Policy constraints extension in accordance with RFC 5280***
        - ***Inhibit anyPolicy extension in accordance with RFC 5280***

  ] in support of claimant authentication and [none]**.**

FIA_X509_EXT.1.2/AuthSvr The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.4.5 Security Management (FMT)

5.4.5.1 FMT_SMF.1/AuthSvr Specification of Management Functions (Authentication Server)

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

- *Ability to configure claimant verification data*
- *Ability to manage trust store data*
- *Ability to configure administrator authentication credential*
- *Ability to configure trusted channel to relying party*
- *[*
    - *Ability to configure IPsec functionality*
    - *Ability to configure TLS functionality*
    - *Ability to manage claimant authentication policy*
    - *Ability to manage supported authentication-verification methods*
    - *Ability to configure RADIUS shared secret*
    - *Ability to define authorized relying parties*
    - *Ability to configure cryptographic key storage*
    - *Ability to configure lockout policy for failed claimant authentication*
    - *Ability to unlock a claimant account*
    - *Ability to configure certificate validation checking mechanisms*
    - *Ability to define conditions in which claimant authentication attempts are rejected*

]

5.4.6 TOE Access (FTA)

5.4.6.1 FTA_TSE.1 TOE Session Establishment

FTA_TSE.1.1 The TSF shall be able to deny **claimant** session establishment based on [*invalid certificate***,** [

[

- *Administrator defined Time and Date Ranges,*
- *Administrator defined Maximum Number of active Concurrent User Sessions, Maximum Number of Concurrent Sessions Per User Identity Group and/or Maximum Number of Concurrent Sessions per User within a User Identity Group.*
- *Administrator defined list of Endpoint IPv4 addresses and/or subnets, IPv6 addresses and/or subnets, and/or MAC Addresses.*

].

5.4.7 Trusted Path/Channel (FTP)

5.4.7.1 FTP_ITC.1/NAS Inter-TSF trusted channel (Relying Party Communications)

FTP_ITC.1.1/NAS The TSF shall provide [***an IPsec***] communication channel between itself and **a relying party** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2/NAS The TSF shall permit [*the TSF, **or the relying party***] to initiate communication via the trusted channel.

FTP_ITC.1.3/NAS The TSF shall initiate the communication via the trusted channel for [*responses to authentication request messages received from the relying party*].

5.5 TOE SFR Dependencies Rationale

The Security Functional Requirements (SFRs) in this Security Target represent the SFRs identified in the NDcPPv2.2e and mod_authsvr_v1.0. Guidance from this PP and the PP-module were followed to include selection-based SFRs based on the behavior of the TSF.

5.6 Security Assurance Requirements

5.6.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPPv1.0 which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Table 17: Assurance Measures

Assurance Class	Components	Components Description
Security Target (ASE)	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational enironment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE summary specification
Development (ADV)	ADV_FSP.1	Basic Functional Specification
Guidance documents (AGD)	AGD_OPE.1	Operational user guidance

Assurance Class	Components	Components Description
	AGD_PRE.1	Preparative procedures
Life cycle support (ALC)	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability assessment (AVA)	AVA_VAN.1	Vulnerability survey

5.6.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPPv2.2e. As such, the NDcPPv2.2e SAR rationale is deemed acceptable.

5.7 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 18: 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 development evidence also contains a tracing of the interfaces to the SFRs described in this ST. The ST and the CC Guidance document contain all of this information.
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 19: How TOE SFRs are Met

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td>FAU_GEN.1
FAU_GEN.1/AuthSvr</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, Table 15. 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>Nov 20 18:15:14 sec-sns-3715 CISE_Administrative_and_Operational_Audit 0000000004 1 0 2018-11-20 18:15:14.613 +00:00 0000017199 51001 NOTICE Administrator-Login: Administrator authentication succeeded, ConfigVersionId=89, AdminInterface=GUI, AdminIPAddress=10.155.84.67, AdminSession=AdminGUI_Session, AdminName=martinf8, OperationMessageText=Administrator authentication successful

Each record contains the following fields:
• Category Name—The logging category to which a message belongs (sec-sns-3715 in the above record)
• Message Class—The group to which a message belongs (CISE_Administrative_and_Operational_Audit in the above record)
• Message Code—A unique message code identification number associated with a message (0000000004 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 (2018-11-20 18:15:14.613 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 (SSH, TLS/HTTPS) and successful SSH rekey</td><td>Attempts of secure encrypted communications/connections (SSH, TLS/HTTPS). The communications include the remote administrator establishing</td></tr>
</table>
</td></tr>
</table>

TOE SFRs	How the SFR is Met	
		a session and the TOE sending syslog data. The identity of the non-TOE entity is included in the log record.
	All use of the user identification and authentication mechanism.	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.
	Unsuccessful attempt to validate a certificate	The reason for failure of certificate validation attempts is logged.
	Changes to the time.	Changes to the time are logged, including old and new values for time, as well as origin of attempt
	Initiation of an update to the TOE.	TOE updates and the result of the update attempts are logged as configuration changes.
	Termination of a remote session.	Termination of a remote session (due to inactivity) is logged (as a terminated cryptographic path).
	Termination of an interactive session.	Termination of an Interactive session (due to logging off) is logged (as the session ending).
	Initiation, termination and failures in trusted channels.	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.
	Initiation, termination and failures in trusted paths.	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.

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td><table><tr><td>All management activities of TSF data (e.g. Modification of the behaviour of the transmission of audit data to an external IT entity; Any attempt to initiate a manual update; Modification, deletion, generation/import of cryptographic keys</td><td>The use of the security management functions are logged, along with the origin or source of the attempt. For the generating/import of, changing, or deleting of cryptographic keys, The subject and the purpose of the cryptographic key is logged.</td></tr></table>
<p>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”.</p>
<p>One can obtain reports on the log collection status for all Cisco ISE nodes. Log collection errors are noted by alarms via the dashboard.</p></td></tr>
<tr><td>FAU_GEN.2</td><td>The TOE ensures 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 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.</td></tr>
<tr><td>FAU_STG.1
FAU_STG_EXT.1
FMT_MOF.1/Functions
FMT_MOF.1/Services</td><td><p>The TOE is stand-alone and stores its own syslog events locally on the platform. All the events generated per day are stored on the TOE and secured from unauthorized deletion. The TOE can offload events to other entities (including other ISE nodes) over TLS protected syslog. The Security Administrators can configure securing the syslog data using TLS. By default upon adding the remote logging target through the GUI, the remote logging target is enabled. The audit events are not sent to the remote logging target until the administrator has configured which type of logging audit records need to be sent. ISE will transmit audit information in realtime when the ISE has an established connection to the Non-TOE External Secure Syslog Server</p>
<p>The TOE uses a log rotation mechanism for local log files. The local log files are rotated out and overwritten by newer ones after a certain size threshold (configurable through the WebUI) is reached. The configuration can be from 10MB to 100MB. 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. Only the Security Administrator may delete all of the rolled over log files by the "Delete Local Logs Now" selection in the administration application. The ISE RBAC (Role-Based Access Control) policy does not allow for any user that is not a Security Administrator to delete log files. No user can modify log files because there is no mechanism that allows this.
After the configured storage period of time has passed for logs the events exceeding the age are deleted.</p></td></tr>
</table>

TOE SFRs	How the SFR is Met
	The administrators that are able to view the logs (at Operations > Reports) 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. ISE allows the administrators to start and stop the following services – • Ability to stop and start services via the CLI command "application stop ise" and "application start ise" • Ability to re-load or shutdown the ISE appliance • Local and remote logging • Clock settings • Cryptographic services • Authentication services
FCO_NRO.1 FCO_NRR.1	The TOE has the ability to validate the authenticity of the NAS (as defined by RFC 3579) and prevent this component from being spoofed. The TOE receives the transmitted Access-Request and has the ability to identify that it was transmitted from the Authenticator. The TOE ensures secure communication between ISE and the Network Access Server (NAS) by utilizing IPsec. IPsec provides secure communication by authenticating the sender, detecting any changes in data during transmission, and encrypting the data that is sent. This ensures that no data passes between ISE and NAS via an unauthenticated protocol. ISE uses a session cache to store active sessions, which helps in managing session continuity and handling interruptions. The TOE has the ability to return a valid response to the NAS upon receipt of an Access-Request as defined RFC 2865.
FCS_CKM.1 FCS_CKM.2	Asymmetric cryptographic keys are generated in accordance with the FFC schemes using cryptographic key sizes of 2048 bits or greater that meet the FIPS 186-4, Digital Signature Standard and using Diffie-Hellman group 14 that meets RFC 3256, Section 3. The TOE also uses ECC schemes using "NIST curves" P-256, P-384, P-521 that meets the FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 The TOE implements RSA key establishment schemes (key sizes – 2048, 3072 and 4096 bits) that is conformant to RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1" The cryptographic key establishment is implemented in the TOE according to the RSA-based schemes that meet the NIST SP 800-56B for TLS and digital signatures, Elliptic curve-based schemes for digital signatures in IKE authentication, TLS and SSH, Finite-field based schemes that meet FIPS PUB 186-4 (CAVP Cert # A4446, 4595) for TLS for TLS, and FFC Schemes using 'safe-prime' groups that meet NIST SP 800-56A for IPsec and SSH.

TOE SFRs	How the SFR is Met
	The TOE implements Diffie-Hellman (group 14) based key establishment schemes that meets RFC 3526, Section 3 and Elliptic curve-based schemes that meet the NIST Special Publication 800-56A Revision 2., "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography". The TOE also implements and uses the prime and generator specified in RFC 3526 Section 3 when generating parameters for the key exchange. The TOE also uses Finite field -based key establishment schemes that meets the NIST SP800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography". The TOE acts as both a sender and receiver for RSA based key establishment and Elliptic curve-based key establishment schemes.
FCS_CKM.3	By default, the TOE implements access control mechanisms so the permissions in the WebUI to the administrators are restricted to prevent disclosure after the cryptographic keys are provisioned. There is no mechanism that would allow any non-authorized administrators to access plaintext keys or critical security parameters. These persistent secret and private keys are used by the TSF in authentication services to ensure secure communication and verification of claimants – • **EAP-TLS** – Private keys used in the server-side certificates (installed on ISE) to establish secure TLS tunnels during authentication. • **RADIUS Authentication** - Private keys used for secure authentication. • **LDAP Authentication** – Secret keys when authenticating users via LDAP. • **Secure Communication** – Private keys used in HTTPS certificates to secure the WebUI, SSH administrative access and TLS channel to the syslog server.
FCS_CKM.4	The TOE meets all requirements 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. The secret keys used for symmetric encryption, private keys, and CSPs used to generate keys, are zeroized immediately after use, or on system shutdown. See Table 22, below for more information. This is followed by a read-verify, which if fails, leads to zeroization process repeating. The AES key that is used to encrypt these other keys is stored in the DRAM. The keys stored on the hard disk drive can be destroyed completely by overwriting the hard disk drive with zeroes and this is accomplished by the *Perform System Erase* utility.
FCS_COP.1/DataEncryption	The TOE provides symmetric encryption and decryption capabilities using AES (as specified in ISO 18033-3), in CBC mode (as specified in ISO 10116), CTR mode (as specified in IOS 10116) and GCM mode (as specified in ISO 19772) with key sizes of 128 bits, 192 bits (CBC only) and 256 bits. These key sizes are used for both TLS, IPsec and SSH. CTR mode is used only in SSH. The AES CAVP certificate number is listed in Table 21: CAVP Certificate References
FCS_COP.1/SigGen	The TOE will provide cryptographic signature services using RSA Digital Signature Algorithm with key sizes of 2048, 3072 and 4096 bits that meets the FIPS 186-4 Digital Signature Standard. The ISE product can be configured to

TOE SFRs	How the SFR is Met
	generate key sizes of 1024 bit, but administrative guidance for the evaluated configuration instructs administrators to only use keys with size 2048, 3072 and 4096 bits. In addition, the TOE will provide cryptographic signature services using ECDSA with key size of 256 or greater as specified in FIPS PUB 186-4, "Digital Signature Standard". The TOE provides cryptographic signature services using ECDSA that meets ISO/IEC 14888-3, Section 6.4 with NIST curves P-256, P-384 and P-521.
FCS_COP.1/Hash	The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384 and SHA-512. SHA-256 and SHA-512 are used for generating certificate signing requests or generating self-signed certificates on the TOE. SHA-1, SHA-256, SHA-384 and SHA-512 are used for TLS, IPsec and SSH.
FCS_COP.1/KeyedHash	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1(key size – 160 bits, block size 512 bits), HMAC-SHA-256 (key size – 256 bits, block size 512 bits) HMAC-SHA-384 (key size – 384 bits, block size 1024 bits) and HMAC-SHA-512 (key size -512 bits, block size 1024 bits) and meets the ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2" standard. Note that HMAC-SHA-1, HMAC-SHA-256 and HMAC-SHA-512 are used for SSH connections, while HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 are used for TLS and IPsec connections. The MAC lengths for HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 are 160, 256, 384 and 512 bits respectively.
FCS_EAPTLS_EXT.1	The user access policies rely on EAP-TLS authentication method to validate users and devices. They are designed to provide secure authentication and authorization for users and devices. Certificate based authentication ensures that only trusted endpoints can access network resources. Here are the key components of the user access policy enforced by the TOE – **Authentication using Certificates** – The TOE supports EAP-TLS, where users or devices present X.509 digital certificates for identity verification. The TOE validates the certificates against a trusted CA list, ensuring that the certificate is issued by an authorized entity. The claimant certificate revocation checking via OCSP is supported. The TOE can use certificates to identify users from Active Directory (AD) or LDAP. **Authorization using Certificates –** The TOE can use certificate attributes for authorization decisions and the policies can be based on the issuer (CA), SAN, Key Usage and Extended Key Usage (EKU). By leveraging certificates for authentication and authorization, the TOE ensures a secure access control mechanism. EAP is an authentication framework and key distribution protocol. TLS is the only permitted authentication type for the Supplicant and the Authentication server. If authentication of the Supplicant is successful, the Authentication server returns Access-Accept packet and generates the Pairwise Master Key (PMK). The PMK is received by the Authenticator via the MS-MPPE-Recv-Key EAP attribute but is not forwarded to the Supplicant as the Supplicant has derived its own copy of the PMK it received from the Authentication Server.

TOE SFRs	How the SFR is Met
	The TOE implements the EAP-TLS protocol with only the TLS v1.2. The cipher suites supported are listed in Section 5.4.3.2. The client requesting authentication is verified by the TOE after the client certificate (X509v3) is verified. The following ciphersuites are supported: Mandatory Ciphersuite: • TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288 • TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
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 to securely establish the encrypted remote session. The sessions are not established with invalid certificates. Note that port 80 is exposed on the product, but only as a redirect to port 443. HTTP connections are not allowed.
FCS_IPSEC_EXT.1	The TOE implements IPsec to provide authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network. The IPsec implementation provides VPN client to TOE capabilities. The VPN client to TOE configuration would be where a remote VPN client connects into the TOE in order to gain access to an authorized

TOE SFRs	How the SFR is Met
	private network. Authenticating with the TOE would give the VPN client a secure IPsec tunnel to connect over the internet into their private network. In addition to tunnel mode, which is the default IPsec mode, the TOE also supports transport mode, allowing for only the payload of the packet to be encrypted. If tunnel mode is explicitly specified, the router will request tunnel mode and will accept only tunnel mode. The TOE implements IPsec to provide certificates-based authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. The TOE uses the following encryption algorithms - AES-CBC-128, AES-CBC-192 and AES-CBC-256, for encrypting the IKEv1 Phase 1, IKEv1 Phase 2 and IKEv2 payloads. The administrator is instructed in the AGD to ensure that the size of key used for ESP must be greater than or equal to the key size used to protect the IKE payload. The IPsec protocol ESP is implemented using the cryptographic algorithms AES-CBC-128, AES-CBC-192 and AES-CBC-256 together with HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512. The IKE protocols implement Peer Authentication using RSA and ECDSA with X.509v3 certificates. The TOE supports reference identifiers as configured by the Administrator to be the Subject Alternative Name (SAN) IPv4 address field in the certificate of the peer. If the TOE successfully matches the reference identifier to the presented identifier, IKE authentication will succeed. In the WebUI, the identifier is configured as "NAD IP address" field. IKE separates negotiation into two phases: phase 1 and phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2 IKE establishes the IPsec SA. IKE maintains a trusted channel, referred to as a Security Association (SA), between IPsec peers that is also used to manage IPsec connections, including: • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based), • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation. The TOE supports both IKEv1 and IKEv2 session establishment. As part of this support, the TOE can only use main mode as the Aggressive mode is disabled by default. The TOE supports configuration lifetimes of both Phase 1 SAs and

TOE SFRs	How the SFR is Met
	Phase 2 SAs for IKEv1 and IKEv2 and this is configured through the WebUI. The time values for Phase 1 SAs can be limited up to 24 hours and for Phase 2 SAs up to 720 hours, but it is configurable to 8 hours. The TOE supports Diffie-Hellman Group 14 (2048-bit keys), 19 (256-bit Random ECP) and 20 (384-bit Random ECP) in support of IKE Key Establishment. These groups are configurable in the IPsec configuration page of the WebUI. These keys are generated using the HMAC Deterministic Random Bit Generator (DRBG), as specified in SP 800-90, and the following corresponding key sizes (in bits) are used: 320 (for DH Group 14), 256 (for DH Group 19), and 384 (for DH Group 20). The nonces used in IKE exchanges are generated in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^[128]. The secret value 'x' used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) is generated using a NIST-approved HMAC Deterministic Random Bit Generator (DRBG). IPsec provides secure tunnels between two peers, such as two routers and remote VPN clients. An authorized administrator defines which packets are considered sensitive and should be sent through these secure tunnels. When the IPsec peer recognizes a sensitive packet, the peer sets up the appropriate secure tunnel and sends the packet through the tunnel to the remote peer. More accurately, these tunnels are sets of security associations (SAs) that are established between two IPsec peers or between the TOE and remote VPN client. The SAs define the protocols and algorithms to be applied to sensitive packets and specify the keying material to be used. SAs are unidirectional and are established per security protocol (AH or ESP). In the evaluated configuration only ESP will be configured for use. The TOE's SPD defines the rules for handling inbound and outbound IPsec traffic. It plays the crucial role in securing communication between the TOE and network devices by specifying which traffic is protected by the IPsec policies. IPsec policy enforcement specifies whether a packet should be: • **Protected**: Encrypted and authenticated using IPsec. • **Bypassed**: Allowed without encryption • **Dropped**: Blocked if it doesn't match the policy The TOE's SPD is based on IP addresses, so the type of traffic being tunneled (for example, syslog) is irrelevant to the tunneling decisions. The key features are describe below - - The local-address is the TOE's IP address. - The remote-address is the IP of the IPsec peer (in tunnel mode or transport mode). - A remote-subnet is applicable only in tunnel mode and defines the subnet that would be reachable beyond the remote-addr. - Outbound traffic will be encrypted when the source address is local-address, and the destination address is the remote-address (in tunnel or transport mode) - Outbound traffic will bypass the tunnel if the destination address is not the remote-address. - Inbound traffic will be dropped if:

TOE SFRs	How the SFR is Met
	- the source address (prior to decryption) is on the remote-subnet (in tunnel mode); or - the source address is the remote-address, and the packets are not IKE or ESP.
FCS_NTP_EXT.1	The TOE can be configured to use NTP to synchronize the TOE's clock with at least three external time sources. NTPv4 is supported by the TOE and the NTP timestamp is not updated from broadcast or multicast addresses. Authentication using hashing algorithms SHA-1 is used to secure the connection between the TOE and the NTP time source to prevent spoofing attacks, man-in-the-middle (MITM)attacks and unauthorized NTP sources.
FCS_RADIUS_EXT.1	The RADIUS protocol is implemented by the TOE for communication with the NAS (Authenticator) per RFC 2865. RADIUS encapsulated EAP and use of EAP-TLS for authentication is implemented according to RFC 3579 and 5216 respectively. Other authentication frameworks are disallowed. The TOE uses the following types of claimant-held keys for authenticating the claimants – - **Claimant held private key** that corresponds to a public key in a certificate issued by a trusted CA. - **Software keys** – Private keys that can be stored in the claimant's operating system key store. - **Client device held private keys** where certificate-based keys are issued to mobile devices. - **Hardware-based private keys** that can be stored on a smart card, TPM or HS, ensuring that the key is protected from extraction
FCS_RBG_EXT.1	The TOE uses a platform-based random bit generator that complies with ISO/IEC 18031:2011 using HMAC_DRBG w/SHA-256 Deterministic Random Bit Generation (DRBG) operating in FIPS mode. In addition, the DRBG is seeded by an entropy source that is at least 256-bit value derived from a highly sensitive and proprietary noise source described in the proprietary Entropy Design document. This implementation is not configurable.
FCS_SSHS_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 262126 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 supports RSA client public-keys authentication (rsa-sha2-256, rsa-sha2-512) and password-based authentication for administrators accessing the TOE through SSHv2. It supports rsa-sha2-256 and rsa-sha2-512 for server host key authentication. It supports RSA key sizes of 2048, 3072 and 4096 for both client and host public key authentication. The TOE implementation of SSHv2 supports the following encryption algorithms - AES-128-CTR, AES-256-CTR to ensure confidentiality of the session. SSH

TOE SFRs	How the SFR is Met
	connection are rekeyed before 1 hour or 1GB has been transmitted using that key. Note that the TOE complies with RFCs 4251, 4252, 4253, 4254, 5656 and 6668. The following integrity algorithms are supported: hmac-sha1, hmac-sha2-256, hmac-sha2-512 The diffie-hellman-group14-sha1 or ecdh-sha2-nistp521, ecdh-sha2-nistp384, and/or ecdh-sha2-nistp256 are the only allowed key exchange method used. Optional characteristics are not supported. The TOE ensures and verifies that the SSH client's presented public key matches one that is stored within the TOE's SSH server's authorized keys file.
FCS_STG_EXT.1	The TOE employs the Trusted Platform Module (TPM) to protect persistent private and secret keys. The TPM uses hardware keys to protect critical security parameters by encrypting them, so they can only be decrypted by the TPM. This process that protects the keys from disclosure is called wrapping or binding a key.
FCS_TLSC_EXT.1 FCS_TLSC_EXT.2	The TOE implements TLS 1.2, conformant to RFC 5246, and rejects all other versions of TLS and SSL. The TOE supports the following ciphersuites as a TLS client – • TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288 • TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 The TOE only supports standard extensions, methods, and characteristics. TLS is used for establishing encrypted sessions with other instances of the TOE and IT entities to send/receive audit data. The trusted channel is established only when the peer certificate is valid. LDAPS has support for additional extensions to support communication with external authentication stores.

TOE SFRs	How the SFR is Met
	The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125. When the TOE acts as a TLS client to LDAPS servers, it obtains the RFC 6125 reference identifiers from the administrator configured value in the LDAP Identity Source Hostname/IP field. (Administration application. Menu: Administration > Identity Management > External Identity Sources. Left-Navigation: LDAP. "Connection" tab. Hostname/IP field) When the TOE acts as a TLS client to TLS Secure Syslog servers, it obtains the reference identifiers from the administrator configured value in the Remote Logging Targets **IP/Host Address** field. (Administration application. Menu: Administration > System > Logging. Left-Navigation: Remote Logging Targets. IP/Host Address field). The TOE supports the following presented identifier types: • subjectAltName entry of type dNSName (DNS-ID in RFC 6125); • CN-ID as defined in RFC 6125, • subjectAltName entry of type IPAddress; and • Wildcards in DNS domain names. The TOE, as a client, can present a certificate to a TLS server for TLS mutual authentication. The TOE supports mutual authentication using X.509 certificates conforming to RFC 5280. When presented with X509 certificates, the TOE verifies the certificate path, and certification validation process by verifying the following rules: • RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates. • The certificate path must terminate with a trusted CA certificate designated as a trust anchor. • The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE. • The TSF shall validate the revocation status of the certificate using OCSP. • The TSF shall validate the extendedKeyUsage field according to the following rules: • *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.* • *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.* • *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.* • *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

TOE SFRs	How the SFR is Met
	The keys establishment parameters are generated using RSA with key sizes 2048 bits, 3072 bits and 4096 bits and ECDHE curves - P-256, P-384, and P-521. The TOE supports Diffie-Hellman parameters with size 2048, 3072 and 4096 bits. The TOE presents the elliptic curve extension in the Client Hello message. They are configured by default and no additional configuration is needed for the Supported Elliptic Curves/Supported Groups Extension. Keyed-hashing message authentication services HMAC-SHA-1, HMAC-SHA-256 and HMAC-SHA-384 are supported for TLS.

FCS_TLSS_EXT.1(1)
FCS_TLSS_EXT.1(2)
FCS_TLSS_EXT.2
	The TOE implements TLS 1.2, conformant to RFC 5246 and supports the following ciphersuites as a TLS server –
	In WebUI:
	• TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
	• TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
	• TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
	• TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
	• TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
	• TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
	• TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246
	• TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
	• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422
	• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
	• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422
	• TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
	• TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
	In EAP-TLS:
	• TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
	• TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
	• TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
	• TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
	• TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
	• TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288
	• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422
	• TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
	• TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
	• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422
	• TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
	• TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

TOE SFRs	How the SFR is Met
	• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 All connections from clients requesting SSL2.0, SSL3.0, TLS1.0 and TLS1.1 are denied. TLS is used for HTTPS/TLS for management purposes. For the WebUI connections, the TOE checks the identifier by the SAN UPN field in the client's certificate and matching it against the list in the external authentication server. For the EAP-TLS connections, the TOE checks the TLS client's certificate identifier by setting an Authorization Policy to check for the DNS name field in the certificate's SAN. If the claimant's certificate does not contain a valid identifier, the TOE sends a RADIUS Access-Reject packet back to show that the claimant's certificate does not have authorization. If the SAN within the client certificate does not match the expected identifier on the TOE, the connection will be rejected. Certificate pinning is unsupported by the TOE. For the EAP-TLS connections, the TOE supports all three "NIST curves" P-256, P-384 and P-521 for key exchange whereas the WebUI connection only supports P-256. The key establishment for TLS uses RSA with key size 2048, 3072 and 4096 bits, Diffie-Hellman parameters with size 2048 bits (WebUI only) and ECDHE curve-secp256r1. When presented with X509 certificates, the TOE verifies the certificate path, and certification validation process by verifying the following rules: • RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates. • The certificate path must terminate with a trusted CA certificate designated as a trust anchor. • The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE. • The TSF shall validate the revocation status of the certificate using OCSP. • The TSF shall validate the extendedKeyUsage field according to the following rules: • *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.* • *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.* • *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.* • *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

TOE SFRs	How the SFR is Met
	TLS session resumption is supported by the TOE on WebUI and EAP-TLS connections based on session IDs according to RFC 5246 (TLS1.2). The TOE keeps track of the negotiated sessions using sessions IDs that allows the TOE to resume a TLS session. When a client attempts to reconnect to a TLS server with a session ID, the TLS server can resume the encrypted communication by looking up the session keys. The WebUI also supports session resumption based on session tickets and adhere to the structural format provided in section 4 of RFC 5077.

TOE SFRs	How the SFR is Met
FIA_AFL.1	The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts before privileged administrator or non-privileged administrator is locked out through the administrative CLI using a privileged CLI command or the GUI. When a privileged administrator or non-privileged administrator attempting to log into the administrative CLI or GUI exceeds the administratively set maximum number of failed authentication attempts, the user will not be granted access to the administrative functionality of the TOE until a privileged administrator resets the user's number of failed login attempts through the administrative CLI or GUI. To ensure the Administrator account does not get locked out by the number of failed attempts, the Emergency account (also known as local CLI admin account) must be enabled. This requires the use of an enabled local administrator account and is intended as a last-resort administrative access. This is useful when the TOE is inaccessible via the network or when remote authentication is not available. It is not accessible via SSH or HTTPS. Access to this account should be limited since it has full administrative privileges and only used when no other option is available to gain access to the TOE. Limited access to emergency account helps maintain security. This emergency account is a CLI-only account that is created during the initial set-up of the TOE and this account is not associated with any web-based Admin roles or enabled for SSH sessions. This helps reduce the attack surface by preventing this account to be accessed by SSH or HTTPS.
FIA_AFL.1/AuthSvr	The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts by a claimant using the TOE's authentication services. When the claimant exceeds the maximum number of unsuccessful authentication attempts, it will be locked out until a privileged administrator resets the authentication attempts counter associated with the claimant.
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 Security 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 UI, 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 (other than the display of the warning banner) 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

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>user database or a remote authentication store. The SSH interface supports authentication using SSH keys which are provided during the SSH connection request.</td></tr>
<tr><td>FIA_UAU_EXT.2</td><td>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, the administrator 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 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 (administrator credentials stored locally)
• remote password-based (administrator credentials stored remotely)
• Certificate authentication with the credentials being stored in the LDAPS server.</td></tr>
<tr><td>Remote SSH (CLI)</td><td>• SSH public key (local and remote authentication)
• 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.6</td><td>The TSF requires the administrative users to re-authenticate to the TOE once they change their passwords. They also need to re-authenticate once a privileged administrator resets the number of consecutive failed login attempts that follows a lockout of an administrator account that exceeds the maximum number of consecutive failed login attempts.</td></tr>
<tr><td>FIA_UAU.7</td><td>When a user enters their password at the local console nothing is displayed so that the user password is obscured. Also, the error displayed for the user does</td></tr>
</table>

TOE SFRs	How the SFR is Met
	not give clues about which part of the credentials entered for authentication failed.
FIA_X509_EXT.1/AuthSvr FIA_X509_EXT.1/Rev FIA_X509_EXT.2 FIA_X509_EXT.3	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections and to support authentication for TLS connections to the audit server, the authentication server and EAP-TLS connections. When a certificate is imported/added in to the TOE, the purpose for which the certificate is to be used needs to be specified - • Admin: Authenticating the Admin portal • EAP: For TLS-based EAP authentication • Portal: For communicating with all Cisco ISE end-user portals Different certificates from each node for communicating with the Admin portal (Admin) and for TLS-based EAP authentication (EAP) can be associated. However, only one certificate from each node for each of these purposes can be associated. The certificate path is validated by ensuring that all the CA certificates has the basicConstraints extension and the CA flag is set to TRUE and the certificate path must terminate with a trusted CA certificate. The extendedKeyUsage field is validated according to the rules listed in Section 5.3.3.6. The certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. The digital signature verifications process would show that the certificate had been tampered with when the hash value would be invalid. The physical security of the TOE (A.PHYSICAL_PROTECTION) protects the router and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. OCSP checking is performed when authenticating a certificate provided by the remote server during TLS establishment and IPsec peer authentication. OCSP is used to validate the revocation status of the certificates. Checking is also done for the basicConstraints extension and the CA flag to determine whether they are present and set to TRUE. If they are not, the certificate is not accepted. The TOE performs validation when communication is received from a peer during establishment of a session. During TLS session establishment, when the connection to determine the validity of the certificate cannot be established, the TOE allows the administrator to either accept/not accept the certificate based on the following conditions - a. accept the certificate when: OCSP revocation checks on client connections fail and the ISE configuration contains the two checkboxes unchecked: Reject the request if OCSP returns UNKNOWN status; and Reject the request if OCSP Responder is unreachable

TOE SFRs	How the SFR is Met
	b. not accept the certificate when: OCSP revocation checks on LDAPS client connections fail and the ISE configuration contains the two checkboxes checked: Reject the request if OCSP returns UNKNOWN status; and Reject the request if OCSP Responder is unreachable. If the connection to determine the certificate validity cannot be established, the administrator is able to choose whether or not to accept the certificate. A Certificate Request Message can be generated as specified by RFC 2986 and provide the following information in the request – public key, device-specific information (Node, city and state), Common Name, Organization, Organizational Unit and Country. The TOE can validate the chain of certificates from the Root CA when the CA Certificate Response is received
FMT_MOF.1/ManualUpdate FMT_MTD.1/CoreData	The TOE restricts the ability to enable the functions to perform manual update to the Security Administrator. The TOE restricts access to the management functions to the Security Administrator, which includes managing the TOE's certificate trust store. The TOE supports two levels of administrators, the CLI-admin (local console or SSHv2 accessible) and the web-based admin user. 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.
FMT_SMF.1 FMT_SMF.1/AuthSvr FMT_MTD.1/ CryptoKeys	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 the SFRs - FMT_SMF.1 and FMT_SMF.1/AuthSvr. The Security administrator have the ability to generate, delete and import/export cryptographic keys. The different types of keys that the Security Administrators can manage are – • **Encryption Keys**: Used to encrypt sensitive data in transit and at rest, ensuring data confidentiality. • **Signing Keys**: Utilized for digital signatures to verify the integrity and authenticity of data. • **SSH Keys**: Secure Shell (SSH) keys are used for secure remote access to network devices. • **TLS/SSL Certificates**: Used to secure HTTPS traffic and other communications, ensuring data is encrypted and secure. • **EAP (Extensible Authentication Protocol) Keys**: EAP keys are used in various EAP methods for secure wireless authentication.
FMT_SMR.2	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

TOE SFRs	How the SFR is Met
	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 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 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 deployment, 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 administrators, 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

Read Only Admin	This role may only view the configuration settings in least privilege mode. Modifications are disallowed by design. Any Administrator user can be intentionally moved to Read Only Admin role to operate in least privilege role and only elevated to make modifications on an as-needed basis.

TOE SFRs	How the SFR is Met	
	Helpdesk Admin	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 • View the Cisco ISE dashboard and livelogs • View alarms This role cannot create, update, or delete reports, troubleshooting flows, live authentications, or alarms.
	Identity Admin	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).
	Network Device Admin	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
	Policy Admin	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 provisioning. This role has the following permissions: • Read and write permissions on identities, endpoints, and identity groups (user identity groups and endpoint identity groups)
	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 • Read permissions on administrator account settings and admin group settings
	Super Admin	This role provides access to every Cisco ISE administrative function. This role is assigned to the default administrator account, and has

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><td></td><td>create, read, update, delete, and eXecute (CRUDX) permissions on all Cisco ISE resources.</td></tr>
<tr><td>System Admin</td><td>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
• generate and view reports</td></tr>
<tr><td>LDAP Admin</td><td>Responsible for integrating Cisco ISE with external directory services, such as Microsoft Active Directory or other LDAP-compliant directories</td></tr>
<tr><td>TACACS+ Admin</td><td>Access permission to TACACS+ Device Administration functionality</td></tr>
<tr><td>Customization Admin</td><td>Access permission to Guest Menu and Device Portal Management</td></tr>
<tr><td>ERS Admin</td><td>When External RESTful Services (ERS) is enabled, this role is assigned to administrator(s) capable of using the ERS REST APIs.</td></tr>
<tr><td>ERS Operator</td><td>When External RESTful Services (ERS) is enabled this role allows read only access to ERS REST APIs.</td></tr>
<tr><td>MnT Admin</td><td>This role is limited to perform all activities under the Operations tab to monitor RADIUS, TACACS+, Threat Centric NAC, and to download logs.</td></tr>
<tr><td>Elevated System Admin</td><td>Highest level of access within Cisco ISE, allowing them to perform all possible administrative tasks, including configuration, monitoring, and management of the system.</td></tr>
<tr><td>SPOG Admin</td><td>A Single Point of Governance (SPOG) Admin that manages the centralized control and configuration of the ISE deployment.</td></tr>
</table>

Only the CLI-admin user can perform the following Cisco ISE system-related tasks:
</td></tr>
</table>

TOE SFRs	How the SFR is Met
	- 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. The ability to administer the TOE locally is provided through a console connection to the appliance or hardware hosting the software-only 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 'Security Administrator' specified in the SFRs is synonymous/equivalent to the entire set of TOE administrative levels/administrators.
FPT_SKP_EXT.1 FPT_APW_EXT.1	The TOE by default secures all locally defined user passwords using SHA256 hashing for CLI passwords, and AES encryption for GUI credentials. In addition, the TOE ensures that plaintext user passwords will not be disclosed even to administrators as there exists no interface to access/view passwords. The TOE stores all private keys in a secure directory that is not accessible to administrators. There is no way an administrator can access/view the private keys in the secure directory where they are stored. All pre-shared and symmetric keys are stored in encrypted (AES) form to prevent access. The 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. The AES key used for this encryption is stored on the filesystem and in DRAM.
FPT_STM_EXT.1	The TOE provides a source of date and time information, used in audit timestamps. 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. 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, administrative session timeout and checking for expiry of certificates. The TOE can also synchronize time with a NTP server.
FPT_TST_EXT.1	ISE runs a suite of self-tests during the TOE 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, the administrative web-based UI will not be accessible, and the security administrator will for a limited time window be able to login to the CLI on the KVM (keyboard, video, mouse) console to run the CLI command – "*show application status ise*" to determine that services have been disabled because "FIPS INTEGRITY CHECK HAS FAILED". Eventually the administrator will be unable to login to the CLI even on the KVM as all services are shutdown including the ability to login to the CLI. After authenticating, a fatal error is displayed and the user is only allowed to press <Enter> to logout and no other actions can be performed. The error message

TOE SFRs	How the SFR is Met
	is: "ERROR: ISE SERVICES HAVE BEEN DISABLED BECAUSE FIPS INTEGRITY CHECK HAS FAILED! EITHER REIMAGE FROM ISE INSTALLATION MEDIA, OR CONTACT CISCO TECHNICAL SUPPORT CENTER FOR INSTRUCTIONS ON DIAGNOSING THE FAILURE. Press <Enter> to logout". If the tests pass successfully the FIPS badge is displayed on the web-based screen and the web-based UI will be accessible for login by the security administrator. The self-tests include: **AES Known Answer Test** - With a known input and output, the AES algorithm implementation is tested by comparing the result with the expected result. This is done separately for both encryption and decryption. **AES-GCM Known Answer Test** - With a known input and output, the AES algorithm implementation in GCM mode is tested by comparing the result with the expected result. This is done separately for both encryption and decryption. **FIPS 186-4 ECDSA Sign/Verify Test –** The ECDSA signature and verification implementation is tested to ensure the correct implementation of ECDSA for generating and verifying digital signatures. The test involves creating a pair of cryptographic keys (public and private) using specified elliptic curves, generating a signature using the private key for a test message and validating the output against expected results by checking that a given signature correctly validates against a message using the public key. **ECC CDH Know Answer Test** – This tests the SP800-56A Section 5.7.1.2 Elliptic Curve Cryptography Cofactor Diffie-Hellman (ECC CDH) Primitive to ensure the correct implementation of the ECC-based Diffie-Hellman key exchange protocol by verifying that the computed shared secrets match expected values. The test utilizes known inputs (private and public keys) to compute the shared secret and checks it against the expected output. **RSA Known Answer Test** – With a known input and output, the RSA signature service algorithm is tested by comparing the result with the expected result. This is done separately for both signing and verification. **DRBG Known Answer Test**– With known input and output, the DRBG computation is tested by comparing an expected pre-computed and stored result against the result computed at runtime. **HMAC Known Answer Test** - This includes the HMAC-SHA1 KAT, HMAC-SHA256KAT, HMAC-SHA384KAT and HMAC-SHA512 KAT. With a known input and output, the keyed-hash message authentication using each of the HMAC-SHA1, HMAC-SHA256 and HMACSHA512 algorithms is tested by comparing the result with the expected result. **SHA-1/256/384/512 Known Answer Test** - With a known input and output, the cryptographic hashing service implementation using each of the SHA1, SHA256 and SHA512 algorithms is tested by comparing the result with the expected result.

TOE SFRs	How the SFR is Met
	Software Integrity Test (HMAC-SHA1) - The HMAC-SHA1 value of the module is computed and compared to the correct already-computed HMAC-SHA1 value for verification. 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.
FPT_TUD_EXT.1	The TOE has specific ISE 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 manually obtain the updates from the Cisco website and install them. Digital Signatures and published hash mechanisms 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 updates can be downloaded from the software.Cisco.com. The TOE image files are digitally signed so their integrity can be verified during the boot process, and an image that fails an integrity check will not be loaded. If the integrity check fails the administrator should reach out to Cisco support. If the integrity check succeeds the update installation with continue as expected and no further action is required for the update installation to complete. Detailed instructions for how to do this verification are provided in the administrator guidance for this evaluation. The TOE does not support delayed activation. The Security administrator can also use a published hash to verify the integrity of the downloaded image. This is not an automated process, and the Security Administrator needs to compare the hash value of the downloaded image with the published hash to confirm integrity. If the hash value does not match the administrator should not proceed with the update and reach out to Cisco support. If the hash value does match the administrator can proceed with the update. Logs for update actions are located in Operations > Reports > Catalog > Server Instance Report.
FTA_SSL_EXT.1 FTA_SSL.3 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 session inactivity time can be configured between 0 to 999998 minutes. For the CLI, this timeout is configured using the command, by the CLI admin: **idle-timeout *seconds*** [valid range is from 0 to 999998**]**

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>On the WebUI, these settings are configurable by setting the Administration > System > Admin Access > Settings-> Session Timeout setting, which defines a session idle timeout period in minutes. After this period elapses, the session times out and access is no longer possible during this session. The WebUI timeout can be configured between 6 and 100 minutes. The ability to configure these settings is limited to the Super Admin or System Admin.
Each administrator logged onto the TOE can manually terminate their session using the “LogOut” link in the web-based or the “exit” command at the CLI.</td></tr>
<tr><td>FTA_TAB.1</td><td>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. The banner available at the local console and remote CLI via SSH are the same. The banners for the CLI and the GUI are separately configurable.</td></tr>
<tr><td>FTA_TSE.1</td><td>The TOE rejects authentication requests based on invalid credentials but can also impose authorization policies to deny requests based on the following criteria –
• Administrator defined Time and Date Ranges
• Administrator defined Maximum Number of Concurrent User Sessions, Maximum Number of Concurrent Sessions Per User Identity Group and/or Maximum Number of Concurrent Sessions per User within a User Identity Group.
• Administrator defined list of Endpoint IPv4 addresses and/or subnets, IPv6 addresses and/or subnets, and/or MAC Addresses.</td></tr>
<tr><td>FTP_ITC.1
FTP_ITC.1/NAS</td><td>The TOE protects communications with devices to which it sends syslogs, including other iterations of ISE, using TLS. For this communication the TOE acts as a TLS client. The communication channel between the TOE and the NAS is secured via IPsec and the communication via the trusted channel can be initiated by either of the two communicating parties.
The TOE acting as a client also protects communications with external authentication stores in the following manner:
<table>
<tr><td>External Authentication Store</td><td>Protection Mechanism</td></tr>
<tr><td>LDAP Server(s)</td><td>TLS</td></tr>
<tr><td>Active Directory Services (acting as the Secure LDAP server)</td><td>TLS</td></tr>
</table>
The use of TLS or IPSec for the various channels protects the data from disclosure by encryption and by checksums that verify that data has not been modified.</td></tr>
</table>

TOE SFRs	How the SFR is Met
FTP_TRP.1/Admin	All remote administrative communications take place over a secure encrypted SSHv2 (CLI) session or HTTPS/TLS (web-based GUI) session. Both SSHv2 and HTTPS sessions are protected using AES encryption. The remote users can initiate both TLS and SSHv2 communications with the TOE.

7 Annex A: Additional Information

7.1 CAVP Certificate Equivalence

The TOE models, processors, and cryptographic modules included in the evaluation are shown in the following table. The cryptographic module used in all TOE platforms is the CiscoSSL FOM 7.3a

Table 20: Processors and CAVP Certificate References

CPU Family	Operating Environment	Physical Appliance / Platforms	Implementation	CAVP#
Intel Xeon Silver	Intel Xeon Silver 4310 (Ice Lake)	Cisco ISE SNS-3715	CiscoSSL FOM Cryptographic Implementation 7.3a	A4446
	Intel Xeon Silver 4316 (Ice Lake)	Cisco ISE SNS-3755, Cisco ISE SNS-3795		
	Intel Xeon Silver 4310 (Ice Lake) w/ Linux 4 on ESXi 7.0	ISE-VM running on ESXi 7.0/UCSC-C220-M6S (ISE-VM)	CiscoSSL FOM-Virtual 7.3a	A4595

Table 21: CAVP Certificate References

Algorithm	Description	Mode Supported	CAVP Cert. #
AES	Used for symmetric encryption/decryption	CBC (128, 192 and 256 bits) CTR (128 and 256 bits) GCM (128, and 256 bits)	A4446 A4595
Hashing SHS (SHA-1, SHA-256, SHA-384, and SHA-512)	Cryptographic hashing services	Byte Oriented	
Keyed Hash HMAC (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA384, and HMAC-SHA-512)	Keyed hashing services and software integrity test	Byte Oriented	

Algorithm	Description	Mode Supported	CAVP Cert. #
DRBG (Key Size – 256)	Deterministic random bit generation services in accordance with ISO/IEC 18031:2011	HMAC_DRBG	
RSA 2048/3072/4096 bits Signature Gen & Verify Key Gen	Key Generation Signature Generation and Signature Verification	FIPS PUB 186-4 Key Generation (2048-bit key, 3072-bit key, 4096-bit key)	
RSA key establishment	RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1"	Tested with known good implementation	
ECDSA curves P-256, P-384 and P-521 Key Sizes – 256, 384 and 521 bits Signature Gen & Verify Key Gen and Verify	Key Generation and Key verification Signature generation and Signature verification	FIPS PUB 186-4, "Digital Signature Standard (DSS)" (256 bits, 384 bits and 521 bits) NIST curves- P-256, P-384 and P-521	
FFC Scheme using key sizes of 2048-bit DSA KeyPairGen	Key generation	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1	
CVL – KAS-FFC	Key Agreement	NIST Special Publication 800-56A	
CVL-KAS-ECC	Key Agreement	NIST Special Publication 800-56A	
FFC Schemes using 'safe-prime' groups	NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and	Tested with known good implementation	

Algorithm	Description	Mode Supported	CAVP Cert. #
	groups listed in RFC 3526		

7.2 Key Protection and Zeroization

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

Table 22: 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. This key is stored in DRAM.	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. This key is stored in DRAM.	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. The ISE server certificate private key is stored on the local filesystem and in DRAM.	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. The SSH server host private key is stored on the local filesystem and in DRAM.	Generation of a new key Overwritten with: 0x00

Name	Description	Zeroization
SSH Session Key	The results zeroized by overwriting the values with 0x00. This is done when a session is ended. This key is stored in DRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00
RNG Seed	This seed is for the RNG. The seed is stored in DRAM.	Zeroized upon power cycle the device
RNG Seed Key	This is the seed key for the RNG. The seed key is stored in DRAM.	Zeroized upon power cycle the device
RADIUS Shared Secrets	RADIUS Shared Secrets are stored within a local database in non-volatile storage.	When shared secrets are changed. Overwritten by a new value of the key
IKE session encrypt key	This the key IPsec key used for encrypting the traffic in an IPsec connection. This key is stored in DRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session authentication key	This the key IPsec key used for authenticating the traffic in an IPsec connection. This key is stored in DRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IPsec encryption key	This is the key used to encrypt IPsec sessions. This key is stored in DRAM.	Automatically when IPsec session terminated. Overwritten with: 0x00
IPsec authentication key	This is the key used to authenticate IPsec sessions. This key is stored in DRAM.	Automatically when IPsec session terminated. Overwritten with: 0x00
TLS Session Keys	The results zeroized by overwriting the values with 0x00. This is done when a session is ended. This key is stored in DRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00

Name	Description	Zeroization
CLI Passwords	command line interface (CLI) passwords are stored on the local filesystem in a SHA-256 hashed crypted format	When passwords are changed. Overwritten by a new value of the key
Admin UI Passwords	Administrators to administration web application are stored in AES-128 CBC mode encrypted format within a local database in non-volatile storage, when ISE has been configured to use identities in the local storage.	When passwords are changed. Overwritten by a new value of the key
Pairwise Master Key (PMK)	Key generated by the authentication Server after the successful authentication of the Supplicant	Automatically when the TLS session is terminated. Overwritten by zeroes
Key Encryption Key for Encrypting critical security parameters stored in local database	The KEK used to encrypt critical security parametrers in the local database is stored on the filesystem and inaccessible from any software interface.	When modified by running the CLI command ‘application reset-config’, the KEK is modified. Overwritten by zeroes
Local Database passwords	The local database administrator and user passwords are automatically generated using random unique values for each ISE deployment. Security administrators may modify the passwords using the CLI commands: application reset-passwd ise internal-database-admin application reset-passwd ise internal-database-user	When passwords are changed. Overwritten by zeroes

8 ANNEX B: REFERENCES

The following documentation was used to prepare this ST:

Table 23: References

[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated April 2017, version 3.1, Revision 5, CCMB-2017-04-003
[CEM]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, CCMB-2017-04-004, Version 3.1, Revision 5, April 2017
[NDcPP]	collaborative Protection Profile for Network Devices, Version 2.2e, 23-March- 2020
[mod_authsvr_v1.0]	PP-Module for Authentication Servers Version 1.0, 25 January 2023