Assurance Activity Report for
Red Hat Enterprise Linux 8.6

Red Hat Enterprise Linux 8.6 Security Target
Version 1.1

Protection Profile for General Purpose Operating Systems, Version 4.2.1

Functional Package for SSH, Version 1.0 [PKG_SSH_V1.0]

AAR Version 1.1, January 2024

Evaluated by:

2400 Research Blvd, Suite 395
Rockville, MD 20850

Prepared for:

National Information Assurance Partnership
Common Criteria Evaluation and Validation Scheme

The Developer of the TOE:
Red Hat, Inc.

The Author of the Security Target:
Acumen Security, Inc.

The TOE Evaluation was Sponsored by:
Red Hat, Inc.

Evaluation Personnel:
Elliot Keen
Chaitanya Muzumdar

Common Criteria Version
Common Criteria Version 3.1 Revision 5

Common Evaluation Methodology Version
CEM Version 3.1 Revision 5

Revision History

VERSION	DATE	CHANGES
0.1	13/07/2023	Initial Release
0.2	27/11/2023	Updated version number of reference documents
0.3	01/12/2023	Updated version number of reference documents
0.4	01/12/2023	Minor updates
1.0	05/12/2023	Final version for submission.
1.1	01/15/2024	Minor update

Contents

1 TOE Overview

Red Hat® Enterprise Linux® is an open-source operating system (OS) that supports multiple users, user permissions, access controls, and cryptographic functionality.

1.1 Architectural Description of the TOE

1.1.1 TOE Environment

The following components must be present in the operational environment to operate the TOE in the evaluated configuration:

Table 1 - Operational Environment Components

Component	Required	Usage/Purpose/Description for TOE Performance
Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with an SSH client installed that is used by the TOE users (including administrators) to remotely connect to the TOE through SSH protected channels. Any SSH client that supports SSHv2 may be used.
Update Server	Yes	Provides the ability to check for updates to the TOE as well as providing signed updates.

1.1.2 Physical Boundaries

The TOE itself is an operating system which can be installed on any compatible hardware; as such, the TOE does not have physical boundaries. However, the TOE was evaluated on the following hardware:

Table 1 – Hardware Platforms

Vendor	Model	CPU
Dell Inc.	PowerEdge R440	Xeon Silver 42xx
Dell Inc.	PowerEdge R540	Xeon Silver 42xx
Dell Inc.	PowerEdge R640	Xeon Silver 42xx
Dell Inc.	PowerEdge R740	Xeon Silver 42xx
Dell Inc.	PowerEdge R740XD	Xeon Silver 42xx
Dell Inc.	PowerEdge R840	Xeon Silver 42xx
Dell Inc.	PowerEdge R940	Xeon Silver 42xx
Dell Inc.	PowerEdge R940xa	Xeon Silver 42xx
IBM	z15 8561-T01	IBM z15
IBM	z15 8562-T02	IBM z15
IBM	z15 8561-LT1	IBM z15
IBM	z15 8562-LT2	IBM z15

Dell Platforms:
The Xeon Silver 4200 series processors are 2nd Generation Intel® Xeon® Scalable Processors and implement the Cascade Lake microarchitecture.
The TOE was tested on a PowerEdge R740 with a Xeon Silver 4216 CPU.

IBM Platforms:
The TOE is one instance of RHEL 8 running on an abstract machine and has full control over the abstract machine inside an IBM z15 T01, T02, LT1, or LT2 mainframe (machine type 8561 or 8652). The abstract machine is provided by a logical partition of the z15 processor. The partition includes 5 IFL (Integrated Facility for Linux) processors. An IFL is a processor dedicated to and optimized for Linux workloads. Because of SMT, the IFL's appear as 10 logical processors allocated to the partition.
The TOE was tested on a IBM z15 T01 mainframe machine type 8561.

2 Assurance Activities Identification

The Assurance Activities contained within this document include all those defined within the PP_OS_V4.2.1 and PKG_SSH_V1.0 based upon the core SFRs and those implemented based on selections within the PPs/Functional Package.

3 Test Equivalency Justification

3.1 TOE description

The TOE is Red Hat Enterprise Linux 8.6.The TOE itself is an operating system which can be installed on any compatible hardware; as such, the TOE does not have physical boundaries. However, the TOE was evaluated on the following hardware:

Hardware Platforms

Microarchitecture	Processor	Device Family	Hardware Reference	Model
Cascade Lake	Intel Xeon Silver 42XX	Dell	PowerEdge	R440, R540, R640, R740, R740XD, R840, R940, R940xa
z/Architecture	z15	IBM	z15	z15 8561-T01, z15 8562-T02, z15 8561-LT1, z15 8562-LT2

The TOE is Red Hat Enterprise Linux 8.6.

Intel Xeon Silver 42XX:

The test subset was determined by the following factors:

1. All mentioned Dell models use Intel Xeon Silver 4200 series processors.
    a. For Intel Xeon Scalable processor model numbers the 100s place is the generation (i.e. 2nd generation).
    b. All 2nd Generation Scalable Processors use the Cascade Lake microarchitecture.
    c. All of the Xeon Silver 4200 series processors provide the same Instruction Set Extensions.
2. The OS is identical on each of the platforms, and there are no differences in the crypto libraries on the platform themselves.
3. The differences between models are enclosure size, memory, storage, and network ports.
4. The supported Instructions Set Extensions are Intel® SSE4.2, Intel® AVX, Intel® AVX2, Intel® AVX-512

Based on the above factors, Acumen Security tested one CPU model for Dell.

z15:

The test subset was determined by the following factors:

1. All mentioned IBM models use z15 series processors.
2. The TOE is one instance of RHEL 8 running on an abstract machine and has full control over the abstract machine inside an IBM z15 T01, T02, LT1, or LT2 mainframe (machine type 8561 or 8652).
3. The abstract machine is provided by a logical partition of the z15 processor. The partition includes 5 IFL (Integrated Facility for Linux) processors. An IFL is a processor dedicated to and optimized for Linux workloads. Because of SMT, the IFL's appear as 10 logical processors allocated to the partition.
4. The differences between models are enclosure size, memory, storage, and network ports.
5. The processor supports the following instructions sets to support functions:

- Hexadecimal floating point instructions for various unnormalized multiply and multiply add instructions.
- Divide engine scheduler.
- Second generation of BCD-RR architecture.
- Modulo arithmetic.
- Immediate instructions, including various add, compare, OR, exclusive-OR, subtract, load, and insert formats. The use of these instructions improves performance.
- Load instructions for handling unsigned halfwords, such as those used for Unicode.
- Cryptographic instructions, which are known as the MSA, offer the full complement of the AES, SHA-1, SHA-2, and DES algorithms. They also include functions for random number

generation.

- Extended Translate Facility-3 instructions, which are enhanced to conform with the current Unicode 4.0 standard.
- Assist instructions that help eliminate hypervisor processor usage.
- SIMD instructions, which allow the parallel processing of multiple elements in a single instruction.

3.2 Platform/Hardware Differences

In the case of the TOE running on Dell devices, it relies on RDRAND for entropy provided by Intel CPU. All Dell platforms mentioned share the same series of CPU. The hardware only differs by configuration.
With IBM z15 each processor has a true-random-number unit, which can be accessed by TOE, and which is used to feed the kernel entropy pool and to seed pseudo-random-number generators.

3.3 TOE Functional Differences

The TOE boundary on each hardware model provides identical functionality. There is no difference in the way the user interacts with each of the devices or the services that are available for each of these devices. Devices with common device family and common processors will run the same version of the software.

3.4 TOE Management Interface Differences

The TOE is managed via either remote CLI session (SSH) or directly connected CLI. These management options are available on all hardware platforms regardless of the configuration. There is no difference in the management interface for any platform.

3.5 Differences in Libraries Used to Provide TOE Functionality

All software binaries compiled in the TOE are identical and have the same version numbers when the processor is the same except for Initial program loader. IBM Z devices use “zipl” while Dell devices use “grub” as their respective boot loaders.

3.6 Recommendations/Conclusions

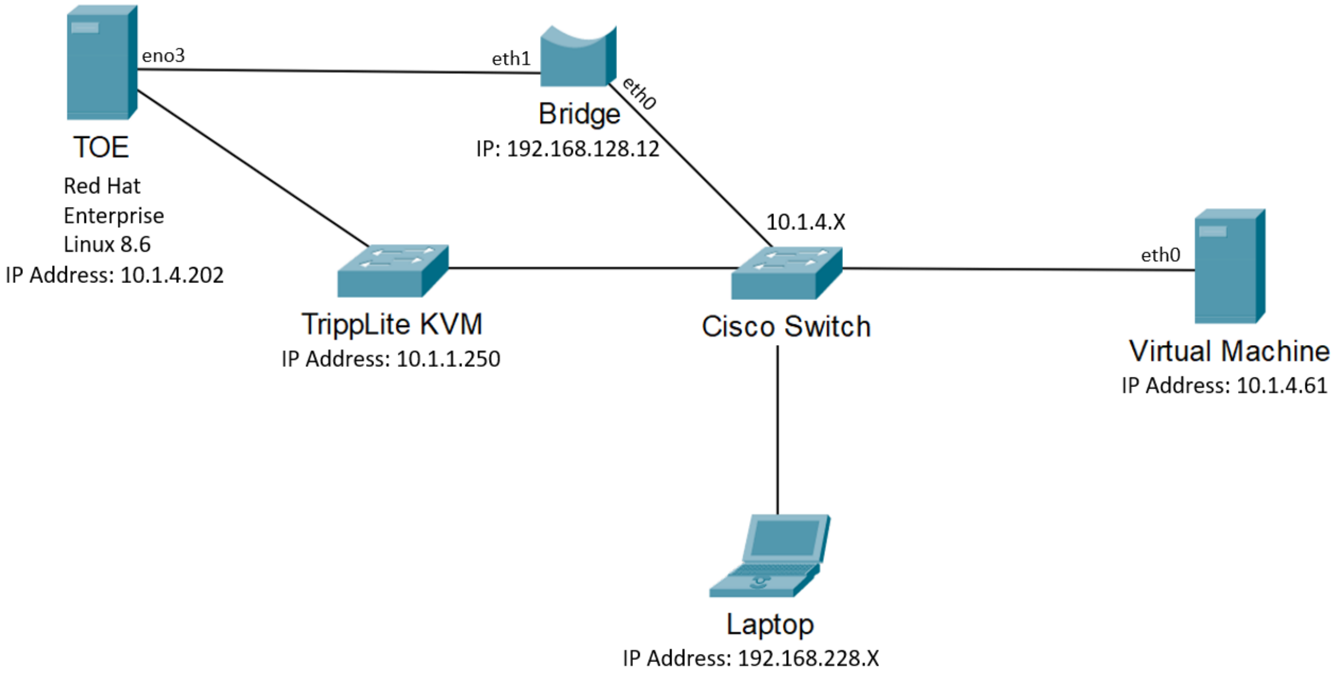
Based on the equivalency rationale listed above, testing will be performed on the following subset:

- PowerEdge R740 with a Xeon Silver 4216 CPU.
- One instance of RHEL 8 running on an abstract machine which has full control over the abstract machine inside an IBM z15 T01, T02, LT1, or LT2 mainframe.

4 Test Bed Descriptions

4.1 Intel

4.1.1 Test Bed Diagram



4.1.2 Configuration Information

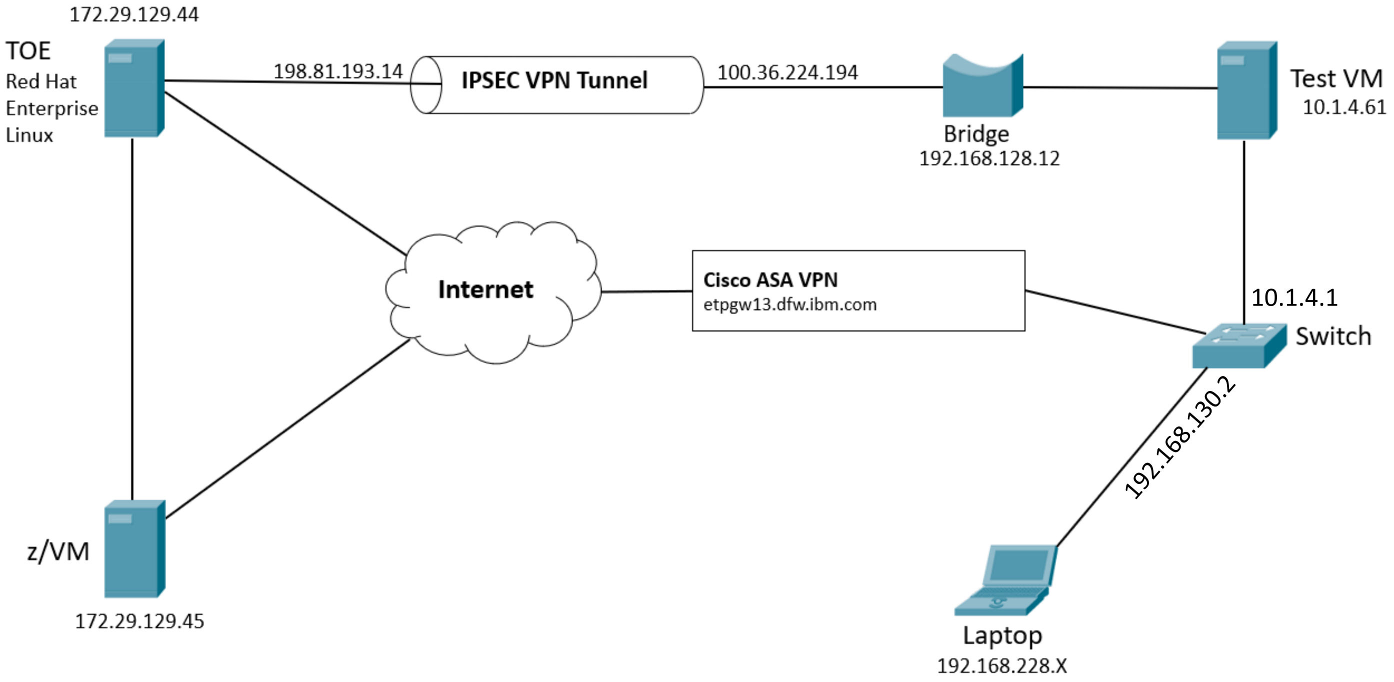
Name	Hardware	OS	Version	Function	Protocols	IP Address	MAC Address	Time	Tools (version)
TOE - OS: Red Hat Enterprise Linux 8.6	PowerEdge R740 with a Xeon Silver 4216 CPU	RhelOS-Linux Kernel - 4.18.0-372.32.1.el8_6.x86_64	8.6	TOE	SSH TLS	eno3: 10.1.4.202	b0:26:28:bd:1a:06	Adjusted manually and verified	tcpdump - version 4.9.3 libpcap - version 1.9.1 OpenSSL-1.1.1k FIPS 25 Mar 2021 OpenSSH_8.0p1
Laptop	HP Pavilion	Windows	Windows 10 pro	Testing Laptop	SSH	192.168.228.X	90-E8-68-B8-8B-E7	Adjusted manually	MobaXtreme V21.3

								and verified	Wireshark Version 4.0.2 WinSCP V5.21.6 tcpdump v4.99.0 HxD Hex Editor Version 2.5.0.0 (x86-64)
Bridge	Raspberry pi 4 Serial: 0000000005a65356	Raspberry Pi OS	4.14.71-v7+	Bridge	SSH	192.168.128.12	00:50:b6:e1:51:01	Adjusted manually and verified	ettercap v0.8.2 tcpdump version 4.9.3 libpcap version 1.8.1 OpenSSL 1.0.2l 25 May 2017 OpenSSH_7.4p1 Raspbian-10+deb9u4
Cisco Switch	Cisco Catalyst 2960-L	ios 15.2	ios 15.2	Gateway (Also acts like a router)	NA	NA	NA	Adjusted manually and verified	NA
TrippLite KVM	NetCommander 16-Port Cat5 KVM	Firmware version 2.2.1263.1.0	NA	Console Access to TOE	RS-232 connection to the TOE on Port 1	10.1.1.250	NA	Adjusted manually and verified	NA

	over IP Switch				KVM with remote IP access				
Testing VM	Dell PowerEdge R830	Kali Linux Kernel - 6.1.0-kali7-amd64	6.1.0	Testing VM	SSH TLS	eth0: 10.1.4.61/ 10.1.4.54	00:0c:29:42:ba:ab	Adjusted manually and verified	tcpdump version 4.99.3 libpcap version 1.10.3 OpenSSL 3.0.8 7 Feb 2023 OpenSSH_9.2p1 acumen-sshc version 1.1.4 acumen-sshs version 1.1.2 acumen-tlsc, Updated to 10/12/2021

4.2 z15

4.2.1 Test Bed Diagram



4.2.2 Configuration Information

Name	Hardware	OS	Version	Function	Protocols	IP Address	MAC Address	Time	Tools (version)
TOE - OS: Red Hat Enterprise Linux 8.6	z15 mainframe	Red Hat Enterprise Linux	8.6	TOE	SSH TLS	enc1b00 : 172.29.129.44	be:02:7a:46:0c:76	Adjusted manually and verified	tcpdump - version 4.9.3 libpcap version 1.9.1 OpenSSH_8.0p1 OpenSSL 1.1.1k FIPS 25 Mar 2021
z/VM	z15 mainframe	Red Hat Enterprise Linux	8.3	Test VM	SSH TLS	enc1c00 : 172.29.129.45	02:df:02:00:00:43	Adjusted manually and verified	tcpdump version 4.9.3 libpcap version 1.9.1 OpenSSL 1.1.1g FIPS 21 Apr 2020

									OpenSSH_8.0p1
Laptop	HP Pavilion	Windows	Windows 10 pro	Testing Laptop	SSH	192.168.228.X	90-E8-68-B8-8B-E7	Adjusted manually and verified	MobaXterm V21.3 Wireshark Version 4.0.2 WinSCP V5.21.6 tcpdump v4.99.0 HxD Hex Editor Version 2.5.0.0 (x86-64)
Bridge	Raspberry pi 4 Serial: 0000000005a65356	Raspberry Pi OS	4.14.71-v7+	Bridge	SSH	192.168.128.12	00:50:b6:e1:51:01	Adjusted manually and verified	ettercap v0.8.2 tcpdump version 4.9.3 libpcap version 1.8.1 OpenSSL 1.0.2l 25 May 2017 OpenSSH_7.4p1 Raspbian-10+deb9u4
Cisco Switch	Cisco Catalyst 2960-L	ios 15.2	ios 15.2	Gateway (Also acts like a router)	NA	NA	NA	Adjusted manually and verified	NA
Testing VM	Dell PowerEdge R830	Kali Linux Kernel-	6.1.0	Testing VM	SSH TLS	eth0: 10.1.4.61/ 10.1.4.54	00:0c:29:42:ba:ab	Adjusted manually and	tcpdump version 4.99.3

		6.1.0-kali7-amd64						verified	libpcap version 1.10.3 OpenSSL 3.0.8 7 Feb 2023 OpenSSH_9.2p1 acumen-sshc version 1.1.4 acumen-sshs version 1.1.2 acumen-tlsc, Updated to 10/12/2021

4.3 Test Time and Location

Testing was carried out at the Acumen Security offices located in 2400 Research Blvd Suite #395, Rockville, MD 20850. Testing occurred from June 2022 through November 2023. Onsite testing occurred from Monday August 28th 2023 to Wednesday August 30th 2023. Onsite testing was performed in Poughkeepsie where the physical server resides. All SFRs that were tested onsite are marked with “onsite testing” in the section headers in the test reports.

The TOE was in a physically protected, access controlled, designated test lab with no unattended entry/exit ways. At the start of each day, the test bed was verified to ensure that it was not compromised. All evaluation documentation was always kept in a secure repository.

5 Detailed Test Cases (TSS and AGD Activities)

5.1 Mandatory Requirements

5.1.1 Audit Data Generation (FAU) - TSS and Guidance Activity

5.1.1.1 FAU_GEN.1 Audit Data Generation (Refined)

5.1.1.1.1 FAU_GEN.1.1 Guidance 1

Objective	The evaluator will check the administrative guide and ensure that it lists all of the auditable events. The evaluator will check to make sure that every audit event type selected in the ST is included.
Evaluator Findings	The evaluator examined the section titled **"Audit"** and **"Audit Event Reference"** in the AGD to verify that it lists all of the auditable events, including every audit event type selected in the ST. Upon investigation, the evaluator found that the AGD lists all audit events found in the GPOSPP and SSH Functional Package. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.1.1.2 FAU_GEN.1.2 Guidance 1

Objective	The evaluator will check the administrative guide and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator will ensure that the fields contains the information required.
Evaluator Findings	The evaluator examined the section titled **"Audit Event Reference"** in the AGD to verify that it provides a format for audit records and that the fields contains the information required. Upon investigation, the evaluator found that the AGD provides lists and example logs that conform to the requirements of the GPOSPP and SSH Functional Package, and all audit logs have provided examples. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2 Cryptographic Support (FCS) - TSS and Guidance Activity

5.1.2.1 FCS_CKM.1 Cryptographic Key Generation (Refined)

5.1.2.1.1 FCS_CKM.1 Cryptographic Key Generation (Refined) TSS 1

Objective	The evaluator shall ensure that the TSS identifies the key sizes supported by the OS.
Evaluator Findings	The evaluator examined the section titled **6 TOE Summary Specification** in the Security Target to verify that the TSS identifies the key sizes supported by the OS. Upon investigation, the evaluator found that the TSS states that **the TOE implements RSA and ECC key generation as specified in FIPS 186-4. The TOE implements FFC key generation as specified in FIPS 186-4 and RFC 3526. RSA key**

	sizes of 2048, 3072, and 4096 are supported. ECC curves P-256, P-384, and P-521 are supported. The FFC key size of L=2048, N=2047 (Group 14) is supported. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.1.2 FCS_CKM.1 Cryptographic Key Generation (Refined) TSS 2

Objective	If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.
Evaluator Findings	The evaluator examined the section titled **6 TOE Summary Specification** in the Security Target to verify that the TSS identifies the usage for each scheme. Upon investigation, the evaluator found that the TSS states that **table 2 in ST section 6.2 identifies which specific cryptographic algorithms are supported for which functional area.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.1.3 FCS_CKM.1 Cryptographic Key Generation (Refined) Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the OS to use the selected key generation scheme(s) and key size(s) for all uses defined in this PP.
Evaluator Findings	The evaluator examined the section titled "**SSH Public key based authentication**" in the AGD to verify that it instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all uses defined in this PP. **Upon investigation, the evaluator found that no configuration is required for generating keys for SSH and TLS. The evaluator also found key generation for SSH addressed in section 3.3.3 "SSH Public key based authentication". In particular, the evaluator found that the configuration for generating keys is done through the CLI.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.1.4 FCS_CKM.1 Cryptographic Key Generation (Refined) CAVP 1

Objective	Key generation test activities are satisfied via CAVP certificates.
Evaluator Findings	For **Key Generation for FIPS PUB 186-4 RSA Schemes** testing is satisfied by CAVP certificate **A1823**. For **Key Generation for Elliptic Curve Cryptography (ECC)** testing is satisfied by CAVP certificate A1823. For **Diffie-Hellman Group 14 and FFC Schemes using "safe-prime" groups** testing is satisfied by *FCS_CKM.2 test 1.* CAVP Certs: #**A1823** (For more information on this CAVP certificate, see Appendix A). Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.1.2.2 FCS_CKM.2 Cryptographic Key Establishment (Refined)

5.1.2.2.1 FCS_CKM.2.1 TSS 1

Objective	The evaluator will ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.
Evaluator Findings	The evaluator examined the section titled **6 TOE Summary Specification** in the Security Target to verify that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1, and if the ST specifies more than one scheme, it identifies the usage for each scheme. Upon investigation, the evaluator found that the TSS states **that ST table 15 describes the usage for each key generation scheme; further TSS identifies the key establishment schemes as EC and FFC.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2.2 FCS_CKM.2.1 Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the OS to use the selected key establishment scheme(s).
Evaluator Findings	The evaluator examined the section titled "Changing SSH Policies" in the AGD to verify that it instructs the administrator how to configure the OS to use the selected key establishment scheme(s). Upon investigation, the evaluator found that the AGD **describes using crypto policies to configure SSH key establishment, and that the key establishment schemes conform to the list in [ST] section 5.2.2.2** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2.3 FCS_CKM.2.1 CAVP

Objective	Key establishment test activities are satisfied via CAVP certificates.
Evaluator Findings	CAVP Certs**: #A1823** (For more information on this CAVP certificate, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3 FCS_CKM_EXT.4 Cryptographic Key Destruction

5.1.2.3.1 FCS_CKM_EXT.4.2 TSS 1

Objective	The evaluator examines the TSS to ensure it describes how the keys are managed in volatile memory. This description includes details of how each identified key is

	introduced into volatile memory (e.g. by derivation from user input, or by unwrapping a wrapped key stored in non-volatile memory) and how they are overwritten.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the keys are managed in volatile memory, including the details of how each identified key is introduced into volatile memory and how they are overwritten. Upon investigation, the evaluator found that the **TSS Description** for **FCS_CKM_EXT.4** in table 14 states that **For volatile memory, the TOE destroys keys and key material by performing a single overwrite consisting of zeroes. For non-volatile memory, the TOE destroys keys and key material by performing an administrator configurable number (default 3) overwrites of the logical storage location with a pseudo random pattern. The pseudo random pattern is generated by an ISAAC PRNG which is initialized from /dev/urandom.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.2 FCS_CKM_EXT.4.2 TSS 2

Objective	The evaluator will check to ensure the TSS lists each type of key that is stored in in non-volatile memory, and identifies how the TOE interacts with the underlying platform to manage keys (e.g., store, retrieve, destroy). The description includes details on the method of how the TOE interacts with the platform, including an identification and description of the interfaces it uses to manage keys (e.g., file system APIs, platform key store APIs).
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS lists each type of key that is stored in in non-volatile memory, and identifies how the TOE interacts with the underlying platform to manage keys, including details on the method of how the TOE interacts with the platform, including an identification and description of the interfaces it uses to manage keys. Upon investigation, the evaluator found that the **TSS description** for **FCS_CKM_EXT.4** mentions that the **TOE destroys keys and key material by performing a single overwrite consisting of zeroes for volatile memory. For non-volatile memory, the TOE destroys keys and key material by performing an administrator configurable number (default 3) overwrites of the logical storage location with a pseudo random pattern. The pseudo random pattern is generated by an ISAAC PRNG which is initialized from /dev/urandom**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.3 FCS_CKM_EXT.4 TSS 3

Objective	If the ST makes use of the open assignment and fills in the type of pattern that is used, the evaluator examines the TSS to ensure it describes how that pattern is

	obtained and used. The evaluator will verify that the pattern does not contain any CSPs.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how that pattern is obtained and used. The evaluator examined the SFR in the Security Target and determined that the open assignment is not used to fill in the type of pattern that is used. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.4 FCS_CKM_EXT.4 TSS 4

Objective	The evaluator will check that the TSS identifies any configurations or circumstances that may not strictly conform to the key destruction requirement.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies any configurations or circumstances that may not strictly conform to the key destruction requirement. Upon investigation, the evaluator found that the **TSS Description** for **FCS_CKM_EXT.4** does not describe any circumstances that would not conform to the requirement Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.5 FCS_CKM_EXT.4 TSS 5 [TD0365 Applied]

Objective	If the selection "destruction of all key encrypting keys protecting target key according to FCS_CKM_EXT.4.1, where none of the KEKs protecting the target key are derived" is included the evaluator shall examine the TOE's keychain in the TSS and identify each instance when a key is destroyed by this method. In each instance the evaluator shall verify all keys capable of decrypting the target key are destroyed in accordance with a specified key destruction method in FCS_CKM_EXT.4.1 The evaluator shall verify that all of the keys capable of decrypting the target key are not able to be derived to reestabish the keychain after their destruction.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies each instance in the keychain when a key is destroyed by this method. The evaluator examined the SFR in section **5.2.2.3** titled **FCS_CKM_EXT.4 Cryptographic Key Destruction** the Security Target and determined that "destruction of all key encrypting keys protecting target key according to FCS_CKM_EXT.4.1, where none of the KEKs protecting the target key are derived" is not included. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.6 FCS_CKM_EXT.4.2 AGD

5.1.2.1 FCS_CKM_EXT.4 Guidance 1

Objective	There are a variety of concerns that may prevent or delay key destruction in some cases. The evaluator will check that the guidance documentation identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS and any other relevant Required Supplementary Information. The evaluator will check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer and how such situations can be avoided or mitigated if possible.
Evaluator Findings	The evaluator examined the section **4.7** titled **Non-volatile drives and keys** in the AGD to verify that it identifies configurations or circumstances that may not strictly conform to the key destruction requirement, that this description is consistent with the relevant parts of the TSS and any other relevant Required Supplementary Information, and that it provides guidance on situations where key destruction may be delayed at the physical layer and how such situations can be avoided or mitigated if possible. Upon investigation, the evaluator found that the AGD states that **"All instances of keys in non-volatile storage might not be deleted if the physical drive has replaced a sector containing a key with a spare sector. To minimize this risk, the physical drive should be end-of-life before a significant amount of damage to the drive's health can occur."** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2 FCS_COP.1(1) Cryptographic Operation - Encryption/Decryption (Refined)

5.1.2.2.1 FCS_COP.1(1) Guidance 1

Objective	The evaluator will verify that the AGD documents contains instructions required to configure the OS to use the required modes and key sizes.
Evaluator Findings	The evaluator examined the sections titled **Installation Steps and TLS Usage** in the AGD to verify that it contains instructions required to configure the OS to use the required modes and key sizes. Upon investigation, the evaluator found that the [AGD] Section 2.2 "Installation steps" and Section 3.4 "TLS Usage" indicates that no configuration is necessary for SSH and TLS. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2.2 FCS_COP.1(1) CAVP

Objective	Encryption/decryption test activities are satisfied via CAVP certificates.
Evaluator Findings	CAVP Certs: **#A1794, #A2781, #A1816** (For more information on these CAVP certificates, see Appendix A). Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.1.2.2.3 FCS_COP.1(1)/SSH Activity 1

Objective	The evaluator shall review the TSF of the base PP to verify consistency with the functionality that was claimed by the base PP to ensure that applicable dependencies are met.
Evaluator Findings	The evaluator examined the section **5.2.2.4** titled **"FCS_COP.1(1) Cryptographic Operation – Encryption/Decryption (Refined)"** in the Security Target to verify consistency with the functionality that was claimed by the base PP to ensure that applicable dependencies are met. Upon investigation, the evaluator found that the ST describes the TOE and includes all dependencies. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2.4 FCS_COP.1(1)/SSH TSS 1

Objective	If perform encryption/decryption services is chosen, the evaluator shall verify that the TSS describes the counter mechanism including rationale that the counter values provided are unique.
Evaluator Findings	The evaluator examined the section **6** titled **"TOE Summary Specification**" in the Security Target to verify that the TSS describes the counter mechanism including rationale that the counter values provided are unique. The evaluator examined the **"TSS Description"** for "**FCS_COP.1(1")** in table 14 of the Security Target which states, "**The CTR mode counter is a 128-bit value output from the SSH key exchange, so it is guaranteed to be unique. The counter is incremented by 1 for each block that is encrypted. The SSH client rekeys at least every 1 GB of data transmitted using a key, so only a maximum of 2^26 counter values could be used, ensuring the counter does not wrap. The SSH server rekeys at least every 512 MB of data transmitted using a key, so only a maximum of 2^25 counter values could be used, ensuring the counter does not wrap."** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2.5 FCS_COP.1(1)/SSH/CAVP

Objective	Test requirements are satisfied via CAVP certificate.
Evaluator Findings	CAVP Certs**: #A1794, #A2781, #A1816** (For more information on these CAVP certificates, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3 FCS_COP.1(2) Cryptographic Operation - Hashing (Refined)

5.1.2.3.1 FCS_COP.1(2) TSS 1

Objective	The evaluator will check that the association of the hash function with other application cryptographic functions (for example, the digital signature verification function) is documented in the TSS.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS documents the association of the hash function with other application cryptographic functions. Upon investigation, the evaluator found that the TSS refers to table 15, in [ST] section 6.1. Table 5 states that SHA-256, SHA-384, and SHA-512 are used for Key Derivation, while SHA-256, SHA-384, and SHA-512 are used for Digital Signatures and HMACs. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.3.2 FCS_COP.1(2) CAVP 1

Objective	Hashing test activities are satisfied via CAVP certificates.
Evaluator Findings	CAVP Certs**: #A1823, A4710** (For more information on these CAVP certificates, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.4 FCS_COP.1(3) Cryptographic Operation - Signing (Refined)

5.1.2.4.1 FCS_COP.1(3) CAVP 1

Objective	Signing test activities are satisfied via CAVP certificates.
Evaluator Findings	CAVP Certs: **#A1823** (For more information on this CAVP certificate, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.5 FCS_COP.1(4) Cryptographic Operation - Keyed-Hash Message Authentication (Refined)

5.1.2.5.1 FCS_COP.1(4) CAVP 1

Objective	Keyed-Hash Message Authentication test activities are satisfied via CAVP certificates.
Evaluator Findings	CAVP Certs: **#A1823** (For more information on this CAVP certificate, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.6 FCS_RBG_EXT.1 Random Bit Generation

5.1.2.6.1 FCS_RBG_EXT.1.1

According to the PP, there are no AA requirements for this SFR TSS and AGD requirements for this SFR.

5.1.2.6.2 FCS_RBG_EXT.1.2

According to the PP, there are no AA requirements for this SFR TSS and AGD requirements for this SFR.

5.1.2.6.3 FCS_RBG_EXT.1 - CAVP

Objective	The evaluator shall verify the implementation of SP 800-90A DRBG supported by the TOE.
Evaluator Findings	CAVP Certs: #**A1794 #A4710** (For more information on this CAVP certificate, see Appendix A). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3 FCS_SSH_EXT.1 SSH Protocol

5.1.3.1.1 FCS_SSH_EXT.1.1 TSS 1

Objective	The evaluator shall ensure that the selections indicated in the ST are consistent with selections in this and subsequent components. Otherwise, this SFR is evaluated by activities for other SFRs.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification i**n the Security Target to verify that the selections indicated in the ST are consistent with selections in the dependent components. Upon investigation, the evaluator found that this **SFR is evaluated by activities for other SFRs.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.2 FCS_SSH_EXT.1.2 TSS 1

Objective	The evaluator shall check to ensure that the authentication methods listed in the TSS are identical to those listed in this SFR component; and, ensure if password-based authentication methods have been selected in the ST then these are also described; and, ensure that if keyboard-interactive is selected, it describes the multifactor authentication mechanisms provided by the TOE.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the authentication methods listed in the TSS are identical to those listed in this SFR component. Upon investigation, the evaluator found "Public Key" and "Password" are the Authentication method listed in the TSS Description for **FCS_SSH_EXT.1** in Table 14. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.3 FCS_SSH_EXT.1.2 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure the configuration options, if any, for authentication mechanisms provided by the TOE are described.
Evaluator Findings	The evaluator examined the **section 3.3.3 and 3.3.4** of guidance documentation titled **"SSH Public key based authentication"** and "**SSH Password based authentication**" respectively to verify that the configuration options for authentication mechanisms provided by the TOE are described. Upon investigation, the evaluator found that **"A user can generate an ssh public/private keypair by running "ssh-keygen -t [rsa|ecdsa] -b [2048|3072|256|384|521]".** and **"The administrator can enable or disable SSH password-based authentication to the SSH server by configuring PasswordAuthentication in /etc/ssh/sshd_config"** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.4 FCS_SSH_EXT.1.3 TSS 1

Objective	The evaluator shall check that the TSS describes how "large packets" are detected and handled.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how "large packets" are detected and handled. Upon investigation, the evaluator found that the TSS states that **The TOE drops any SSH packet with a packet_length field greater than 262,144 bytes.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.5 FCS_SSH_EXT.1.4 TSS 1

Objective	The evaluator will check the description of the implementation of SSH in the TSS to ensure the encryption algorithms supported are specified. The evaluator will check the TSS to ensure that the encryption algorithms specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the optional characteristics and the encryption algorithms supported, and verify that the encryption algorithms are identical to those listed for this component. Upon investigation, the evaluator found that the TSS states that **the TOE supports encryption algorithms aes128-ctr, aes256-ctr, aes128-cbc, aes256-cbc, aes128-gcm@openssh.com, and aes256-gcm@openssh.com**. These are consistent with the selections in [ST] section 5.2.2.10. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.6 FCS_SSH_EXT.1.4 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.7 FCS_SSH_EXT.1.5 TSS 1

Objective	The evaluator will check the description of the implementation of SSH in the TSS to ensure the hashing algorithms supported are specified. The evaluator will check the TSS to ensure that the hashing algorithms specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** section in the Security Target to verify that the TSS lists the supported data integrity algorithms, and that that list corresponds to the list in this component. Upon investigation, the evaluator found that the TSS states that **the TOE supports MAC algorithms hmac-sha2-256, hmac-sha2-512, and implicit**. This selection conforms to [ST] section 5.2.2.10 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.8 FCS_SSH_EXT.1.5 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange**

	algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.9 FCS_SSH_EXT.1.6 TSS 1

Objective	The evaluator will check the description of the implementation of SSH in the TSS to ensure the shared secret establishment algorithms supported are specified. The evaluator will check the TSS to ensure that the shared secret establishment algorithms specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** section in the Security Target to verify that the TSS lists the shared secret establishment algorithms and are identical to those listed for this component. Upon investigation, the evaluator found that the TSS states that **the TOE supports key exchange algorithms ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521**. This is consistent with the selections in [ST] section 5.2.2.10 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.10 FCS_SSH_EXT.1.6 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.11 FCS_SSH_EXT.1.7 TSS 1

Objective	The evaluator will check the description of the implementation of SSH in the TSS to ensure the KDFs supported are specified. The evaluator will check the TSS to ensure that the KDFs specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** of the Security Target to verify that the KDFs specified are identical to those listed for this component.

	Upon investigation, the evaluator found out that Table 15 in section 6.1 details KDF support. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.12 FCS_SSH_EXT.1.8 TSS 1

Objective	The evaluator shall check the TSS to ensure that if the TOE enforces connection rekey or termination limits lower than the maximum values that these lower limits are identified. In cases where hardware limitation will prevent reaching data transfer threshold in less than one hour, the evaluator shall check the TSS to ensure it contains: a. An argument describing this hardware-based limitation and b. Identification of the hardware components that form the basis of such argument. For example, if specific Ethernet Controller or Wi-Fi radio chip is the root cause of such limitation, these subsystems shall be identified.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification i**n ST to ensure that if the TOE enforces connection rekey or termination limits lower than the maximum values that these lower limits are identified. Upon investigation, the evaluator found out that “**The TOE can rekey SSH client connections before a key has been used for over an hour or used to protect more than 1 GB of data. The TOE can also rekey SSH server connections before a key has been used for over an hour or used to protect more than 512 MB of data.”** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.1.13 FCS_SSH_EXT.1.8 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that if the connection rekey or termination limits are configurable, it contains instructions to the administrator on how to configure the relevant connection rekey or termination limits for the TOE
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.2 FCS_SSHC_EXT.1 SSH Protocol – Client

5.1.3.2.1 FCS_SSHC_EXT.1 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.3 FCS_SSHS_EXT.1 SSH Protocol – Server

5.1.3.3.1 FCS_SSHS_EXT.1 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it contains instructions to the administrator on how to ensure that only the allowed mechanisms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section **3.3.1** titled **Changing SSH policies** in the AGD to verify that it contains instructions to the administrator on configuring the TOE so only the allowed mechanisms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD describes **the process to update-crypto-policies, that is used to set the policies for the various cryptographic back-ends such as OpenSSH. The administrator can make any changes to the files located in /etc/crypto-policies/. Both server and client application inherits the cipher preferences, the key exchange algorithms as well as the GSSAPI key exchange algorithms. To opt-out from the policy for the server, uncomment the line containing CRYPTO_POLICY= in /etc/sysconfig/sshd.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.4 FCS_STO_EXT.1 Storage of Sensitive Data

5.1.3.4.1 FCS_STO_EXT.1.1 TSS

Objective	The evaluator will check the TSS to ensure that it lists all persistent sensitive data for which the OS provides a storage capability. For each of these items, the evaluator will confirm that the TSS lists for what purpose it can be used, and how it is stored. The

	evaluator will confirm that cryptographic operations used to protect the data occur as specified in FCS_COP.1(1).
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS lists all persistent sensitive data for which the OS provides a storage capability. Upon investigation, the evaluator found that the **TSS Description** for **FCS_STO_EXT.1** states that "**The TOE includes the OpenSSL library to securely store sensitive data. OpenSSL provides file encryption services using AES-128 or AES-256 in CBC mode. Sensitive data include passwords and keys and can be found in /etc directory. /etc contains system-wide configuration files and system databases. Access to the files in /etc is limited with strict file permissions and/or encryption**." Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.5 FCS_TLSC_EXT.1 TLS Client Protocol

5.1.3.5.1 FCS_TLSC_EXT.1.1 TSS 1

Objective	The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the cipher suites supported are specified. The evaluator shall check the TSS to ensure that the cipher suites specified include those listed for this component.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the cipher suites supported and that the cipher suites specified include those listed for this component. Upon investigation, the evaluator found that the **TSS Description** for **FCS_TLSC_EXT.1**/**Inte**l and **z15** states the lists of supported ciphersuites. The list includes ciphers " 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_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 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_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_GCM_SHA384 as defined in RFC 5289" for Intel based TOEs and "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_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288, " for z15 based TOEs. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.5.2 FCS_TLSC_EXT.1.1 Guidance 1

Objective	The evaluator shall also check the operational guidance to ensure that it contains instructions on configuring the OS so that TLS conforms to the description in the TSS.
Evaluator Findings	The evaluator examined the section titled **“TLS Usage”** in the AGD to verify that it contains instructions on configuring the product so that TLS conforms to the description in the TSS. Upon investigation, the evaluator found that the AGD states that the ciphersuites and supported curves are do not need to be configured. The AGD provides the list of additional parameters that must be used to ensure TLS conforms to the description in the TSS. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.5.3 FCS_TLSC_EXT.1.2 TSS 1

Objective	The evaluator shall ensure that the TSS describes the client’s method of establishing all reference identifiers from the application-configured reference identifier, including which types of reference identifiers are supported (e.g. Common Name, DNS Name, URI Name, Service Name, or other application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the client’s method of establishing all reference identifiers from the application-configured reference identifier, including which types of reference identifiers are supported and whether IP addresses and wildcards are supported. Upon investigation, the evaluator found that the **TSS Description** for **FCS_TLSC_EXT.1 /Intel** , **FCS_TLSC_EXT.1/z15** describes the reference identifier, how incoming certificates are parsed, and how correct or incorrect reference identifiers are determined. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.5.4 FCS_TLSC_EXT.1.2 TSS 2

Objective	The evaluator shall ensure that this description identifies whether and the manner in which certificate pinning is supported or used by the OS.
Evaluator Findings	The evaluator examined the TSS section in the Security Target to verify that the TSS identifies whether and the manner in which certificate pinning is supported or used by the product. Upon investigation, the evaluator found that the TSS states that **the TOE does not support certificate pinning**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.5.5 FCS_TLSC_EXT.1.2 Guidance 1

Objective	The evaluator will verify that the AGD guidance includes instructions for setting the reference identifier to be used for the purposes of certificate validation in TLS.
Evaluator Findings	The evaluator examined the section titled **“TLS Usage”** in the AGD to verify that it includes instructions for setting the reference identifier to be used for the purposes of certificate

	validation in TLS. Upon investigation, the evaluator found that the AGD describes the "-verify_hostname" and "-verify_ip" parameters necessary to set the reference identifier. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.6 FCS_TLSC_EXT.2

5.1.3.6.1 FCS_TLSC_EXT.2 TSS 1

Objective	The evaluator will verify that TSS describes support for the Supported Groups Extension and whether the required behavior is performed by default or may be configured.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes support for the Supported Groups Extension and whether the required behavior is performed by default or may be configured. Upon investigation, the evaluator found out that **"On Intel-based platforms, the TOE presents the supported Elliptic Curves Extension in the Client Hello message with the P-256, P-384, and P-521 curves."** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.6.2 FCS_TLSC_EXT.2 Guidance 1

Objective	If the TSS indicates that support for the Supported Groups Extension must be configured to meet the requirement, the evaluator will verify that AGD guidance includes configuration instructions for the Supported Groups Extension.
Evaluator Findings	The evaluator examined the section titled **"TLS Usage"** in the AGD to verify, if support for the Supported Groups Extension must be configured, that the AGD includes configuration instructions for the Supported Groups Extension. Upon investigation, the evaluator found that the AGD states that [ST] conformant behavior is configured by default, with no administrator action necessary. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.4 User Data Protection (FDP) - TSS and Guidance Activity

5.1.4.1 FDP_ACF_EXT.1 Access Controls for Protecting User Data

5.1.4.1.1 FDP_ACF_EXT.1.1 TSS 1

Objective	The evaluator will confirm that the TSS comprehensively describes the access control policy enforced by the OS. The description must include the rules by which accesses to particular files and directories are determined for particular users. The evaluator will inspect the TSS to ensure that it describes the access control rules in such detail that given any possible scenario between a user and a file governed by the OS the access control decision is unambiguous.

Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS comprehensively describes the access control policy enforced by the OS, including the rules by which accesses to files and directories are determined for particular users. Upon investigation, the evaluator found that the **TSS Description** for **FDP_ACF_EXT.1** thoroughly describes the access policy as standard UNIX permission bits, defining access for read, write, and execute permissions, with automatic blocking of write access to filesystems mounted as read-only. The evaluator verified that the descriptions of the “**umask” attribute, POSIX-type Access Control Lists, and the additional access control bits of “SUID”, “SGID”, and “SAVETXT”** are fully described. The TSS also describes the files and filesystems to be protected. The evaluator verified that the description of the access control rules is sufficiently detailed that all scenarios of access control from users to files are unambiguously identified. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.5 Identification and Authentication (FIA) - TSS and Guidance Activity

5.1.5.1 FIA_UAU.5 Multiple Authentication Mechanisms (Refined)

5.1.5.1.1 FIA_UAU.5.1 TSS 1

Objective	If user name and PIN that releases an asymmetric key is selected, the evaluator will examine the TSS for guidance on supported protected storage.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target and determined that “authentication based on user name and a PIN is **not selected**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.5.1.2 FIA_UAU.5.2 TSS 1

Objective	The evaluator will ensure that the TSS describes each mechanism provided to support user authentication and the rules describing how the authentication mechanism(s) provide authentication.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes each mechanism provided to support user authentication and the rules describing how the authentication mechanism(s) provide authentication. Upon investigation, the evaluator found that the **TSS Description** for **FIA_UAU.5 mentions the supported user authentication mechanism as username and password, it further describes how password-based authentication is performed on the TOE using PAM (Pluggable Authentication Module).For key-based authentication OpenSSH server verifies the signature of the client with the public key stored in the user’s authorized_keys file.** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.1.5.1.3 FIA_UAU.5.2 Guidance 1

Objective	The evaluator will verify that configuration guidance for each authentication mechanism is addressed in the AGD guidance.
Evaluator Findings	The evaluator examined the section titled “Creating User Accounts” in the AGD to verify that it addresses configuration guidance for each authentication mechanism. Upon investigation, the evaluator found that the AGD describes the process for creating new user accounts, configuring passwords. AGD section “SSH Public key-based authentication” describes the process for adding SSH public keys to an existing user account. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.5.2 FIA_X509_EXT.1 X.509 Certificate Validation

5.1.5.2.1 FIA_X509_EXT.1.1 TSS 1 [TD715 Applied]

Objective	The evaluator will ensure the TSS describes where the check of validity of the certificates takes place. The evaluator ensures the TSS also provides a description of the certificate path validation algorithm. If there are exceptional use cases where the OS cannot perform revocation checking in accordance with at least one of the revocation methods, the evaluator will ensure the TSS describes each revocation checking exception use case, and for each exception, the alternate functionality the TOE implements to determine the status of the certificate and disable functionality dependent on the validity of the certificate.
Evaluator Findings	The evaluator reviewed the section **6** titled **TOE Summary Specification** to ensure that it describes where the check of validity of the certificates takes place. The evaluator ensures the TSS Description for FIA_X509_EXT.1 also provides a description of the certificate path validation algorithm. The evaluator reviewed the TSS Description for FIA_X509_EXT.1 to ensure that, if the OS cannot perform revocation in accordance with one of the revocation methods, the evaluator ensured that the TSS describes each revocation checking exception use case, and for each exception, the alternate functionality the TOE implements to determine the status of the certificate and disable functionality dependent on the validity of the certificate. Upon investigation, the evaluator found that the TSS states that “**The X.509 certificates are validated using the certificate path validation algorithm defined in RFC 5280 ie: the public key algorithm and parameters are checked, the current date/time is checked against the validity period, revocation status is checked using CRL, issuer name of X matches the subject name of X+1, extensions are processed.”** It also states **that the certificate validity is checked when the TOE receives the certificate during a TLS handshake and that the TOE certificate validation algorithm ensures that the certificate path terminates in a trusted root CA.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.6 Security Management (FMT) - TSS and Guidance Activity

5.1.6.1 FMT_MOF_EXT.1 Management of security functions behavior

5.1.6.1.1 FMT_MOF_EXT.1.1 TSS 1

Objective	The evaluator will verify that the TSS describes those management functions that are restricted to Administrators, including how the user is prevented from performing those functions, or not able to use any interfaces that allow access to that function.
Evaluator Findings	The evaluator examined the section titled section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS describes those management functions that are restricted to Administrators, including how the user is prevented from performing those functions, or not able to use any interfaces that allow access to that function. Upon investigation, the evaluator found that the **TSS Description** for **FMT_MOF_EXT.1** states that **"The TOE restricts all "Administrator" management activities listed in FMT_SMF_EXT.1 to users who are members of the "wheel" group. Members of this group are considered the administrators, because group membership allows users to elevate their privileges, allowing management of the TOE, using the sudo command."** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.6.2 FMT_SMF_EXT.1 Specification of Management Functions

5.1.6.2.1 FMT_SMF_EXT.1.1 AGD 1

Objective	The evaluator will verify that every management function captured in the ST is described in the operational guidance and that the description contains the information required to perform the management duties associated with the management function.
Evaluator Findings	The evaluator examined the sections titled **"User/Administrator Accounts**", **"Audit , "NTP", "System Updates", "Firewall", "Warning Banner", "Changing SSH policies**", **"Software Restriction Policies"**, **"Configure Password Policy"** and **"Changing the ECDSA curve"** in the AGD to verify that every management function captured in the ST is described in the operational guidance and that the description contains the information required to perform the management duties associated with the management function. Upon investigation, the evaluator found that the AGD describes all management functions of the TOE in sufficient detail for the administrator to carry out those duties. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.7 Protection of Security Functions (FPT) - TSS and Guidance Activity

5.1.7.1 FPT_ACF_EXT.1 Access controls

5.1.7.1.1 FPT_ACF_EXT.1.1 TSS 1

Objective	The evaluator will confirm that the TSS specifies the locations of kernel drivers/modules, security audit logs, shared libraries, system executables, and system configuration files. Every file does not need to be individually identified, but the system's conventions for storing and protecting such files must be specified.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the locations of kernel drivers/modules, security audit logs, shared libraries, system executables, and system configuration files. Upon investigation, the evaluator found that the **TSS description** for **FPT_ACF_EXT.1** states that "**The TOE uses the file/directory permissions described in FDP_ACF_EXT.1 to prevent unprivileged users from modifying kernel drivers/modules, security audit logs, shared libraries, system executables, and system configuration files**" Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.2 FPT_SBOP_EXT.1 Stack Buffer Overflow Protection

5.1.7.2.1 FPT_SBOP_EXT.1.1 TSS 1

Objective	For stack-based OSes, the evaluator will determine that the TSS contains a description of stack-based buffer overflow protections used by the OS. These are referred to by a variety of terms, such as stack cookie, stack guard, and stack canaries. The TSS must include a rationale for any binaries that are not protected in this manner.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS contains a description of stack-based buffer overflow protections used by the TOE, and includes a rationale for any binaries that are not protected in this manner. Upon investigation, the evaluator found that the **TSS description** for **FPT_SBOP_EXT.1.** states that "**The TOE is compiled with the option "stack-protector-strong" to add a stack canary and associated verification code during the entry and exit of function frames to prevent stack-based buffer overflows."** The ST **section 6.3** also lists all binaries not protected by stack mashing protections in use by the TOE, and their rationales for exclusion. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.2.2 FPT_SBOP_EXT.1.1 TSS 2

Objective	For OSes that store parameters/variables separately from control flow values, the evaluator will verify that the TSS describes what data structures control values, parameters, and variables are stored. The evaluator will also ensure that the TSS includes a description of the safeguards that ensure parameters and variables do not intermix with control flow values.

Evaluator Findings	The evaluator examined the description of the OS in the Security Target and determined that parameters/variables are not stored separately from control flow values. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.3 FPT_TST_EXT.1 Boot Integrity

5.1.7.3.1 FPT_TST_EXT.1.1 TSS 1

Objective	The evaluator will verify that the TSS section of the ST includes a comprehensive description of the boot procedures, including a description of the entire bootchain, for the TSF.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS includes a comprehensive description of the boot procedures, including a description of the entire bootchain, for the TSF. Upon investigation, the evaluator found that the **TSS description** for **FPT_TST_EXT.1** describes the entire boot chain for both Dell and IBM hardware in explicit detail. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.3.2 FPT_TST_EXT.1.1 TSS 2

Objective	The evaluator will ensure that the OS cryptographically verifies each piece of software it loads in the bootchain to include bootloaders and the kernel. Software loaded for execution directly by the platform (e.g. first-stage bootloaders) is out of scope. For each additional category of executable code verified before execution, the evaluator will verify that the description in the TSS describes how that software is cryptographically verified.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS states that the OS cryptographically verifies each piece of software it loads in the bootchain to include bootloaders and the kernel. For each additional category of executable code verified before execution, the evaluator verifies that the description in the TSS describes how that software is cryptographically verified. Upon investigation, the evaluator found that the TSS states that “**For Dell, The signature on the first-stage boot loader (shim.efi) is verified to be signed by a certificate authority (CA) stored in the firmware database. shim.efi then uses an embedded RSA 2048 public key to verify the signature on the RSA 2048 code signing public key. This code signing key is used to verify the signature of the second-stage boot loader, GRUB 2 (grubx64.efi). Finally, GRUB 2 uses the code signing key to verify the signature on the OS kernel before passing control to the kernel. The kernel has 2 more embedded keys that are used to authenticate drivers and kernel modules.”** And for IBM “**The machine loader locates the signature data stored on disk. It then works through a list of certificates used to verify the signature, stopping on the first matching certificate. These certificates can only be changed through firmware updates of the machine loader. The loader locates the address of the program (in this case, the Stage 3 Bootloader) to be booted and verifies that its signature matches the certificate. If it does, then it executes the program. Other parts of the boot chain and the operating system can find the certificate**

	by looking at absolute address 14 in the IPL Parameter Block. This can be used to locate the IPL Information Report Block where the certificate is located. **The machine loader then turns control over to the TOE. The TOE controlled boot process begins with the Stage 3 Bootloader (a component of the larger zipl bootloader). The Stage 3 Bootloader reads the signature and RSA 4096-bit public key for the Kernel from the IPL Parameter Block. The Stage 3 Bootloader attempts to verify the signature of the Kernel. If the signature verification succeeds, the Stage 3 Bootloader passes control to the Kernel. If there is no signature, or signature verification failed, the Stage 3 Bootloader terminates the boot.”.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.3.3 FPT_TST_EXT.1.1 TSS 3

Objective	The evaluator will verify that the TSS contains a description of the protection afforded to the mechanism performing the cryptographic verification.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS contains a description of the protection afforded to the mechanism performing the cryptographic verification. Upon investigation, the evaluator found that the TSS describes that the mechanisms for performing cryptographic verification are embedded in firmware, which are only modifiable through system updates. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.7.4 FPT_TUD_EXT.1 Trusted Update

5.1.7.4.1 FPT_TUD_EXT.1.1 Guidance 1

Objective	The evaluator will check for an update using procedures described in the documentation and verify that the OS provides a list of available updates.
Evaluator Findings	The evaluator examined the section titled **“Systems Updates**” and **“Configure automatic software updates”** in the AGD to verify that it describes procedures to check for an update. Upon investigation, the evaluator found that the AGD describes the use of the “dnf” program to check for updates, and the process for configuring automatic updates. AGD describes the possible values of the “dnf” command, and states that the dnf program will provide a list of updates. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.4.2 FPT_TUD_EXT.1.2 TSS 1

Objective	All supported origins for the update must be indicated in the TSS and evaluated.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS indicates all supported origins for the update.

	Upon investigation, the evaluator found that the **TSS Description** for **FPT_TUD_EXT.2** mentions that the update to itself and application software are verified by RSA 4096 with SHA-256 prior to installation. Updates to the TOE and application software are downloaded by the TOE from the Red Hat CDN. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.5 FPT_TUD_EXT.2 Trusted Update for Application Software

5.1.7.5.1 FPT_TUD_EXT.2.1 Guidance 1

Objective	The evaluator will check for updates to application software using procedures described in the documentation and verify that the OS provides a list of available updates.
Evaluator Findings	The evaluator examined the section titled **"Systems Updates" and "Configure automatic software updates"** in the AGD to verify that it describes procedures to check for an update to application software. Upon investigation, the evaluator found that the AGD describes the use of the "dnf" program to check for updates, and the process for configuring automatic updates. AGD describes the possible values of the "dnf" command, and states that the dnf program will provide a list of updates. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.7.5.2 FPT_TUD_EXT.2.2 TSS 1

Objective	All origins supported by the OS must be indicated in the TSS and evaluated. However, this only includes those mechanisms for which the OS is providing a trusted installation and update functionality. It does not include user or administrator-driven download and installation of arbitrary files.
Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS indicates all supported origins for updates. Upon investigation, the evaluator found that the **TSS Description** for **FPT_TUD_EXT.2** states "**The TOE has the ability to check for updates to itself and application software. Both types of updates are verified by RSA 4096 with SHA-256 prior to installation. Updates to the TOE and application software are downloaded by the TOE from the Red Hat CDN**". Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.8 **Trusted Path (FTP)**

5.1.8.1 FTP_TRP.1.3 TSS 1

Objective	The evaluator will examine the TSS to determine that the methods of remote OS administration are indicated, along with how those communications are protected.

Evaluator Findings	The evaluator examined section **6** titled **TOE Summary Specification** in the Security Target to verify that the TSS indicates the methods of remote OS administration, along with how those communications are protected. Upon investigation, the evaluator found that the **TSS Description** for **FTP_TRP.1** states "**The TOE uses the SSH Server protocol to protect the communications with remote users."** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.8.1.1 FTP_TRP.1.3 TSS 2

Objective	The evaluator will also confirm that all protocols listed in the TSS in support of OS administration are consistent with those specified in the requirement, and are included in the requirements in the ST.
Evaluator Findings	The evaluator examined the section **6** titled **TOE Summary Specification** in the Security Target to verify that all protocols listed in the TSS are consistent with those specified in the requirement, and are included in the requirements in the ST. Upon investigation, the evaluator found that the **TSS Description** for **FTP_TRP.1** states " **TOE uses the SSH Server protocol to protect the communications with remote users.**" This is consistent with **Section 5.2.8.1 and 5.2.8.2** of the Security Target which specify SSH to provide a trusted communication channel between itself and authorized IT entities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.8.1.2 FTP_TRP.1.3 Guidance 1

Objective	The evaluator will confirm that the operational guidance contains instructions for establishing the remote administrative sessions for each supported method.
Evaluator Findings	The evaluator examined the section titled **"Creating user accounts"** in the AGD to verify that it contains instructions for establishing the remote administrative sessions for each supported method. Upon investigation, the evaluator found that the AGD states that **administrative sessions can only be established over SSH.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2 Optional Requirements

5.2.1 User Data Protection (FDP)

5.2.1.1 FDP_IFC_EXT.1 Information flow control

5.2.1.1.1 FDP_AFC_EXT.1.1 TSS 1

Objective	The evaluator will confirm that the TSS comprehensively describes the access control policy enforced by the OS. The description must include the rules by which accesses to particular files and directories are determined for particular users. The evaluator will inspect the TSS to ensure that it describes the access control rules in such detail that given any possible scenario between a user and a file governed by the OS the access control decision is unambiguous.
Evaluator Findings	The evaluator examined section **6** of the **TOE Summary Specification** in the Security Target to verify that the TSS comprehensively describes the access control policy enforced by the OS, including the rules by which accesses to particular files and directories are determined for particular users. Upon investigation, the evaluator found that the TSS thoroughly describes the access policy as standard UNIX permission bits, defining access for read, write, and execute permissions, with automatic blocking of write access to filesystems mounted as read-only. The evaluator verified that the descriptions of the “umask” attribute, POSIX-type Access Control Lists, and the additional access control bits of “SUID”, “SGID”, and “SAVETXT” are fully described. The TSS also describes the files and filesystems to be protected. The evaluator verified that the description of the access control rules is sufficiently detailed that all scenarios of access control from users to files are unambiguously identified. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3 Selection-Based Requirements

5.3.1 FCS_TLSC_EXT.2 TLS Client Protocol

5.3.1.1 FCS_TLSC_EXT.2 TLS Client Protocol TSS 1

Objective	The evaluator will verify that TSS describes support for the Supported Groups Extension and whether the required behavior is performed by default or may be configured.
Evaluator Findings	The evaluator examined section **6** of **TOE Summary Specification** in the Security Target to verify that the TSS describes support for the Supported Groups Extension and whether the required behavior is performed by default or may be configured. Upon investigation, the evaluator found that the TSS states that **The TOE presents the supported Elliptic Curves Extension in the Client Hello message with the P-256, P-384, and P-521 curves.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.2 FCS_TLSC_EXT.2 TLS Client Protocol Guidance 1

Objective	If the TSS indicates that support for the Supported Groups Extension must be configured to meet the requirement, the evaluator will verify that AGD guidance includes configuration instructions for the Supported Groups Extension.
Evaluator Findings	The evaluator examined the section titled **"TLS Usage"** in the AGD to verify, if support for the Supported Groups Extension must be configured, that the AGD includes configuration instructions for the Supported Groups Extension. Upon investigation, the evaluator found that the AGD states that [ST] conformant behavior is configured by default, with no administrator action necessary. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6 Security Assurance Requirements

6.1 AGD_OPE.1 Operational User Guidance

6.1.1 AGD_OPE.1

6.1.1.1 AGD_OPE.1 Guidance 1

Objective	If cryptographic functions are provided by the OS, the operational guidance shall contain instructions for configuring the cryptographic engine associated with the evaluated configuration of the OS. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the OS.
Evaluator Findings	The evaluator examined the section titled "**Installation"** in the AGD to verify that it contains instructions for configuring the cryptographic engine associated with the evaluated configuration of the OS. Upon investigation, the evaluator found that the AGD provides instructions for configuring the TOE into its CC configuration. As part of this configuration, all cryptographic algorithms are limited to only the allowed algorithms. The section "Administration", "Installation", "System Updates", "Configure Automatic Software Updates" of AGD provides instructions to the Administrator for performing an update. Step by step instructions are provided for the administrator to follow including downloading the image, copying it to the TOE and installing it. This includes integrity verification. The entirety of the guidance documentation identifies the evaluated capabilities of the TOE by describing how to configure each for Common Criteria. The evaluator also examined the section titled **"Disclaimers"** in the AGD to verify that it provides a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the OS. Upon investigation, the evaluator found that the AGD states that **only OpenSSL** was tested during the evaluation and that no other engines should be used**.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.1.1.2 AGD_OPE.1 Guidance 2

Objective	The documentation must describe the process for verifying updates to the OS by verifying a digital signature – this may be done by the OS or the underlying platform. The evaluator shall verify that this process includes the following steps: • Instructions for obtaining the update itself. This should include instructions for making the update accessible to the OS (e.g., placement in a specific directory). • Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes generation of the hash/digital signature. The OS will likely contain security functionality that does not fall in the scope of evaluation under this PP. The operational guidance shall make it clear to

	an administrator which security functionality is covered by the evaluation activities.
Evaluator Findings	The evaluator examined the section titled **"Updates"** in the AGD to verify that it describes the process for verifying updates to the TOE by verifying a digital signature. Upon investigation, the evaluator found that the AGD states that **all updates are signed using the vendor-controlled RSA 4096 key.** The evaluator examined the section titled **"Updates"** in the AGD and found the instructions to check for updates and install updates. The commands used are "dnf check-update" and "dnf update" respectively. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.2 AGD_PRE.1 Preparative Procedures

6.2.1 AGD_PRE.1

6.2.1.1 AGD_PRE.1 Guidance 1

Objective	As indicated in the introduction above, there are significant expectations with respect to the documentation—especially when configuring the operational environment to support OS functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for the OS in the ST.
Evaluator Findings	The evaluator examined the AGD to verify that it adequately addresses all platforms claimed for the OS in the ST. Upon investigation, the evaluator found that the AGD describes all supported platforms in section "**Introduction**" and describes the composition of the operational environment in section "**Getting Started**". Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3 ALC Assurance Activities

6.3.1 ALC_CMC.1

6.3.1.1 ALC_CMC.1 TSS 1

Objective	The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST.
Evaluator Findings	The evaluator examined the Security Target to verify that the ST contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. Upon investigation, the evaluator found that the ST provides a product name and version number in section "**Security Target and TOE Reference.**" Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.1.2 ALC_CMC.1 TSS 2

Objective	If the vendor maintains a web site advertising the OS, the evaluator shall examine the information on the web site to ensure that the information in the ST is sufficient to distinguish the product.
Evaluator Findings	The evaluator examined the vendor web site to ensure that the information in the ST is sufficient to distinguish the product. Upon investigation, the evaluator found that the vendor website at https://www.redhat.com/en/technologies/linux-platforms/enterprise-linux describes the RHEL operating system in a manner sufficient to distinguish it from all other products. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.1.3 ALC_CMC.1 Guidance 1

Objective	Further, the evaluator will check the AGD guidance and OS samples received for testing to ensure that the version number is consistent with that in the ST.
Evaluator Findings	The evaluator examined the AGD to verify that the version number is consistent with that in the ST. Upon investigation, the evaluator found that the AGD describes TOE versions that are consistent with the [ST]. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.2 ALC_CMS.1

6.3.2.1 ALC_CMS.1 Guidance 1

Objective	The evaluator will ensure that the developer has identified (in guidance documentation for application developers concerning the targeted platform) one or more development environments appropriate for use in developing applications for the developer's platform. For each of these development environments, the developer shall provide information on how to configure the environment to ensure that buffer overflow protection mechanisms in the environment(s) are invoked (e.g., compiler and linker flags). The evaluator will ensure that this documentation also includes an indication of whether such protections are on by default, or have to be specifically enabled.
Evaluator Findings	The evaluator examined the platform developer guidance documentation to verify that it identifies one or more development environments appropriate for use in developing applications for the developer's platform. For each of these development environments, the evaluator verified that the developer provides information on how to configure the environment to ensure that buffer overflow protection mechanisms in the environment(s) are invoked (e.g., compiler flags) and whether such protections are on by default. Upon investigation, the evaluator found that [AGD] section "**Application Developers**" provides best practices for use with the TOE, including the compiler flags which must be specifically invoked. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.2.2 ALC_CMS.1 Guidance 1

Objective	The evaluator shall ensure that the TSF is uniquely identified (with respect to other products from the TSF vendor), and that documentation provided by the developer in association with the requirements in the ST is associated with the TSF using this unique identification.
Evaluator Findings	The evaluator examined the section titled **Introduction** in the AGD to verify that it is associated with the TSF using unique identification. Upon investigation, the evaluator found that the guidance documentation states that **the TOE is version 8.6, which is consistent with the [ST]**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.3 ALC_TSU.1

6.3.3.1 ALC_TSU.1 TSS 1

Objective	The evaluator will verify that the TSS contains a description of the timely security update process used by the developer to create and deploy security updates. The evaluator will verify that this description addresses the entire application. The evaluator will also verify that, in addition to the TOE developer's process, any third-party processes are also addressed in the description. The evaluator will also verify that each mechanism for deployment of security updates is described.
Evaluator Findings	The evaluator examined section **6** titled "**TOE Summary Specifications** of the Security Target to verify that the TSS contains a description of the timely security update process that addresses the entire application (including third-party processes). Upon investigation, the evaluator found that the TSS describes the process by which security issues may be submitted to the vendor for evaluation and remediation through the CVE system. The evaluator also examined the Security Target to verify that each mechanism for deployment of security updates is described. Upon investigation, the evaluator found that the TSS describes how updates are obtained. [ST] section 6 describes how security update information is collected by the vendor and how updates are distributed. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.3.2 ALC_TSU.1 TSS 2

Objective	The evaluator will verify that, for each deployment mechanism described for the update process, the TSS lists a time between public disclosure of a vulnerability and public availability of the security update to the OS patching this vulnerability, to include any third-party or carrier delays in deployment. The evaluator will verify that this time is expressed in a number or range of days.
Evaluator Findings	The evaluator examined "ALC_TSU_EXT.1" under the section **6** titled "**TOE Summary Specifications"** in the Security Target to verify that, for each deployment mechanism described for the update process, the TSS lists a time between public disclosure of a vulnerability and public availability of the security update to the OS patching this

	vulnerability. Upon investigation, the evaluator found that the TSS describes how security updates are generated and distributed via the normal update mechanism. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.3.3.3 ALC_TSU.1 TSS 3

Objective	The evaluator will verify that this description includes the publicly available mechanisms (including either an email address or website) for reporting security issues related to the OS. The evaluator shall verify that the description of this mechanism includes a method for protecting the report either using a public key for encrypting email or a trusted channel for a website.
Evaluator Findings	The evaluator examined "ALC_TSU_EXT.1" under the section titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS includes the publicly available mechanisms for reporting security issues related to the TOE, including a method for protecting the report. Upon investigation, the evaluator found that the ST describes the use of the secalert@redhat.com email address and GnuPG encryption mechanism by which security reports may be sent to the vendor. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.4 AVA_VAN.1 Vulnerability Survey

6.4.1 AVA_VAN.1

6.4.1.1 AVA_VAN.1 Activity 1 **[Labgram #116]**

Objective	The evaluator will generate a report to document their findings with respect to this requirement.
Evaluator Findings	The evaluator documented their analysis and testing of potential vulnerabilities with respect to this requirement. Public searches were performed against all keywords found within the Security Target and AGD that may be applicable to specific TOE components. This included protocols, TOE software version, and TOE hardware to ensure sufficient coverage under AVA. The evaluator searched the Internet for potential vulnerabilities in the TOE using the NVD website; The following was performed on 15th January 2024. The National Vulnerability Database (NVD) was searched for publicly reported CVEs. The following components of the TOE were searched:

Component	CPE
aide 0.16-14.el8_5.1	cpe:/:advanced_intrusion_detection_environment
Audit 3.0.7-2.el8.2	cpe:/:linux_audit_project:linux_audit:3.0.7
chrony 4.1-1.el8	cpe:/:chrony_project:chrony:4.1
cryptsetup 2.3.7-2.el8	cpe:/:cryptsetup_project:cryptsetup:2.3.7
curl 7.61.1-22.el8	cpe:/:haxx:curl:7.61.0
dnf 4.7.0-8.el8	cpe:2.3:a:rpm:dnf:*
fapolicyd 1.1-6.el8	cpe:/:fapolicyd_project:fapolicyd:1.1
firewalld 0.9.3-13.el8	cpe:2.3:a:firewalld:firewalld:*

gpgme 1.13.1-11.el8	cpe:/:gnupg:gpgme:1.13.1
grub2-common 2.02-123.el8	cpe:/:gnu:grub2:2.02
gnutls 3.6.16-4.el8	cpe:2.3:a:gnu:gnutls:3.6.16:*:*:*:*:*:*:*
gzip 1.9-12.el8	cpe:2.3:a:gnu:gzip:1.9:*:*:*:*:*:*:*
iptables 1.8.4-22.el8	cpe:2.3:a:netfilter:iptables:1.8.4:*:*:*:*:*:*:*
iputils 20180629-9.el8	cpe:2.3:a:iputils_project:iputils:s20180629:*:*:*:*:*:*:*
kernel 4.18.0-372.9.1.el8_4	cpe:2.3:o:linux:linux_kernel:4.18.0:*:*:*:*:*:*:*
libcap 2.48-2.el8	cpe:2.3:a:libcap_project:libcap:2.48:*:*:*:*:*:*:*
libcap-ng 0.7.11-1.el8	cpe:2.3:a:libcap-ng_project:libcap-ng:0.7.11:*:*:*:*:*:*:*
libpcap 1.9.1-5.el8	cpe:2.3:a:tcpdump:libpcap:1.9.1:*:*:*:*:*:*:*
lzo 2.08-14.el8	cpe:2.3:a:lzo_project:lzo:2.08:*:*:*:*:*:*:*
openldap 2.4.46-18.el8	cpe:2.3:a:openldap:openldap:2.4.46:*:*:*:*:*:*:*
openssh 8.0p1-13.el8	cpe:2.3:a:openbsd:openssh:8.0:p1:*:*:*:*:*:*
openssl 1.1.1k-6.el8_5	cpe:/:openssl:openssl:1.1.1k
pam 1.3.1-16.el8	cpe:2.3:a:linux-pam:linux-pam:1.3.1:*:*:*:*:*:*:*
polkit 0.115-13.el8_5.2	cpe:2.3:a:polkit_project:polkit:0.115:*:*:*:*:*:*:*
rpm 4.14.3-23.el8	cpe:2.3:a:rpm:rpm:4.14.3:*:*:*:*:*:*:*
rsyslog 8.2102.0-7.el8	cpe:2.3:a:rsyslog:rsyslog:8.2102.0:*:*:*:*:*:*:*
sudo 1.8.29-8.el8	cpe:2.3:a:sudo_project:sudo:1.8.29:-:*:*:*:*:*:*
tar 1.30-5.el8	cpe:2.3:a:gnu:tar:1.30:*:*:*:*:*:*:*
xz 5.2.4-3.el8	cpe:2.3:a:tukaani:xz:5.2.4:*:*:*:*:*:*:*
zlib 1.2.11-18.el8_5	cpe:2.3:a:zlib:zlib:1.2.11:*:*:*:*:*:*:*

The searched components were identified based on processing network traffic and parsing file formats. Most components were not vulnerable; a full discussion of the status of each identified vulnerability is in the AVA_VAN.

Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

7 Detailed Test Cases (Test Activities)

7.1 FAU – Audit Data Generation

7.1.1 GEN

7.1.1.1 FAU_GEN.1.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will test the OS's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the ST. This should include all instance types of an event specified. When verifying the test results, the evaluator will ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries.
Test Steps	1. Start-up and shutdown of the audit functions a. Shut down i) Success b. Startup i) Success 2. Authentication events i) Success ii) Failure 3. Use of privileged/special rights events (Successful and unsuccessful security, audit, and configuration changes a. Security changes i) Success ii) Failure b. Audit changes i) Success ii) Failure c. Configuration changes i) Success ii) Failure 4. Privilege or role escalation events i) Success ii) Failure 5. File and object events a. Create i) Success ii) Failure b. Access i) Success ii) Failure c. Delete i) Success ii) Failure d. Modification

	i) Success ii) Failure e. Permission change i) Success ii) Failure 6. User and Group management events a. Add User i) Success ii) Failure b. Delete user. i) Success ii) Failure: c. Modify i) Success ii) Failure d. Disable i) Success ii) Failure e. Enable i) Success ii) Failure f. Credential change i) Success ii) Failure 7. Audit Log Access i) Success ii) Failure 8. Cryptographic verification of software i) Success ii) Failure 9. System reboot, restart, and shutdown events a. System Boot i) Success b. System Shutdown i) Success 10. Kernel module loading and unloading Events. a. Module Load i) Success ii) Failure b. Module Unload i) Success ii) Failure 11. Administrator Access: i) Success ii) Failure 12. Root Access i) Success

	ii) Failure. 13. Software restriction Policies: i) Success ii) Failure 14. Establishment of SSH connection i) Success 15. Termination of SSH connection i) Success
Expected Test Results	The TOE should properly log all events listed
Pass/Fail with Explanation	Pass. The TOE generates the appropriate audit logs. This satisfies the testing requirement.

7.1.1.2 FAU_GEN.1.2 Test#1

Item	**Data**
Test Assurance Activity	The evaluator shall test the OS's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the ST. The evaluator will ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record provide the required information.
Pass/Fail with Explanation	Pass. *FAU_GEN.1.1 Test#1 satisfies all the testing requirements.*

7.2 FCS – Cryptographic Support

7.2.1 CKM

7.2.1.1 FCS_CKM.1 Test #1

Item	**Data**
Test Assurance Activity	**Key Generation for FIPS PUB 186-4 RSA Schemes** The evaluator will verify the implementation of RSA Key Generation by the OS using the Key Generation test. This test verifies the ability of the TSF to correctly produce values for the key components including the public verification exponent e, the private prime factors p and q, the public modulus n and the calculation of the private signature exponent d. Key Pair generation specifies 5 ways (or methods) to generate the primes p and q. These include: 1. Random Primes: • Provable primes • Probable primes 2. Primes with Conditions: • Primes p1, p2, q1,q2, p and q shall all be provable primes • Primes p1, p2, q1, and q2 shall be provable primes and p and q shall be probable primes • Primes p1, p2, q1,q2, p and q shall all be probable primes To test the key generation method for the Random Provable primes method and for all the Primes with Conditions methods, the evaluator must seed the TSF key generation routine with sufficient data to deterministically generate the RSA key

pair. This includes the random seed(s), the public exponent of the RSA key, and the desired key length. For each key length supported, the evaluator shall have the TSF generate 25 key pairs. The evaluator will verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation.

If possible, the Random Probable primes method should also be verified against a known good implementation as described above. Otherwise, the evaluator will have the TSF generate 10 keys pairs for each supported key length nlen and verify:

- n = pâ‹…q,
- p and q are probably prime according to Miller-Rabin tests,
- GCD(p-1,e) = 1,
- GCD(q-1,e) = 1,
- 216 â‰¤ e â‰¤ 2^{256} and e is an odd integer,
- |p-q| > 2nlen/2 - 100,
- p â‰¥ 2nlen/2 -1/2,
- q â‰¥ 2nlen/2 -1/2,
- 2(nlen/2) < d < LCM(p-1,q-1),
- eâ‹…d = 1 mod LCM(p-1,q-1).

Key Generation for Elliptic Curve Cryptography (ECC)
FIPS 186-4 ECC Key Generation Test

For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator will require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator will submit the generated key pairs to the public key verification (PKV) function of a known good implementation.

FIPS 186-4 Public Key Verification (PKV) Test

For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator will generate 10 private/public key pairs using the key generation function of a known good implementation and modify five of the public key values so that they are incorrect, leaving five values unchanged (i.e., correct). The evaluator will obtain in response a set of 10 PASS/FAIL values.

Key Generation for Finite-Field Cryptography (FFC)
The evaluator will verify the implementation of the Parameters Generation and the Key Generation for FFC by the TOE using the Parameter Generation and Key Generation test. This test verifies the ability of the TSF to correctly produce values for the field prime p, the cryptographic prime q (dividing p-1), the cryptographic group generator g, and the calculation of the private key x and public key y.

The Parameter generation specifies 2 ways (or methods) to generate the cryptographic prime q and the field prime p:

- Cryptographic and Field Primes:
    - Primes q and p shall both be provable primes
    - Primes q and field prime p shall both be probable primes

	and two ways to generate the cryptographic group generator g: • Cryptographic Group Generator: ○ Generator g constructed through a verifiable process ○ Generator g constructed through an unverifiable process The Key generation specifies 2 ways to generate the private key x: • Private Key: ○ len(q) bit output of RBG where 1 â‰¤ x â‰¤ q-1 ○ len(q) + 64 bit output of RBG, followed by a mod q-1 operation where 1 â‰¤ x â‰¤ q-1 The security strength of the RBG must be at least that of the security offered by the FFC parameter set. To test the cryptographic and field prime generation method for the provable primes method and/or the group generator g for a verifiable process, the evaluator must seed the TSF parameter generation routine with sufficient data to deterministically generate the parameter set. For each key length supported, the evaluator will have the TSF generate 25 parameter sets and key pairs. The evaluator will verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation. Verification must also confirm: • g != 0,1 • q divides p-1 • gq mod p = 1 • gx mod p = y for each FFC parameter set and key pair. **Diffie-Hellman Group 14 and FFC Schemes using "safe-prime" groups** Testing for FFC Schemes using Diffie-Hellman group 14 and/or "safe-prime" groups is done as part of testing in FCS_CKM.2.1 ***TD0501 Applied***
Pass/Fail with Explanation	Pass. For **Key Generation for FIPS PUB 186-4 RSA Schemes** testing is satisfied by CAVP certificate A1823. For **Key Generation for Elliptic Curve Cryptography (ECC)** testing is satisfied by CAVP certificate A1823. For **Diffie-Hellman Group 14 and FFC Schemes using "safe-prime" groups** testing is satisfied by *FCS_CKM.2 test 1*.

7.2.1.2 FCS_CKM.2 Test #1

Item	Data
Test Assurance Activity	**Key Establishment Schemes** The evaluator will verify the implementation of the key establishment schemes supported by the OS using the applicable tests below. **SP800-56A Key Establishment Schemes** The evaluator will verify the OS's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for

each key agreement scheme verify that the OS has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the discrete logarithm cryptography (DLC) primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator will also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MAC data and the calculation of MAC tag.

Function Test

The Function test verifies the ability of the OS to implement the key agreement schemes correctly. To conduct this test the evaluator will generate or obtain test vectors from a known good implementation of the OS's supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested.

The evaluator will obtain the DKM, the corresponding OS's public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and OS id fields.

If the OS does not use a KDF defined in SP 800-56A, the evaluator will obtain only the public keys and the hashed value of the shared secret.

The evaluator will verify the correctness of the TSF's implementation of a given scheme by using a known good implementation to calculate the shared secret value, derive the keying material DKM, and compare hashes or MAC tags generated from these values.

If key confirmation is supported, the OS shall perform the above for each implemented approved MAC algorithm.

Validity Test

The Validity test verifies the ability of the OS to recognize another party's valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator will obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the OS should be able to recognize. The evaluator generates a set of 30 test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator's public keys, the OS's public/private key pairs, MAC tag, and any inputs used in the KDF, such as the other info and OS id fields.

	The evaluator will inject an error in some of the test vectors to test that the OS recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MAC'd, or the generated MAC tag. If the OS contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties' static public keys, both parties' ephemeral public keys and the OS's static private key to assure the OS detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass). The OS shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator will compare the OS's results with the results using a known good implementation verifying that the OS detects these errors. **RSAES-PKCS1-v1_5 Key Establishment Schemes** The evaluator shall verify the correctness of the TSF's implementation of RSAES-PKCS1-v1_5 by using a known good implementation for each protocol selected in FTP_ITC_EXT.1 that uses RSAES-PKCS1-v1_5. **Diffie-Hellman Group 14** The evaluator shall verify the correctness of the TSF's implementation of Diffie-Hellman group 14 by using a known good implementation for each protocol selected in FTP_ITC_EXT.1 that uses Diffie-Hellman Group 14. **FFC Schemes using "safe-prime" groups (identified in Appendix D of SP 800-56A Revision 3)** The evaluator shall verify the correctness of the TSF's implementation of "safe-prime" groups by using a known good implementation for each protocol selected in FTP_ITC_EXT.1 that uses "safe-prime" groups. This test must be performed for each "safe-prime" group that each protocol uses.
Pass/Fail with Explanation	Pass. For **Diffie-Hellman Group 14** testing is satisfied by *FTP_ITC_EXT.1 test.* For **FFC Schemes using "safe-prime" groups (identified in Appendix D of SP 800-56A Revision 3)** testing is satisfied by CAVP certificate A1834.

7.2.1.3 FCS_CKM.4 Test #1

Item	Data
Test Assurance Activity	Test 1: Applied to each key held as in volatile memory and subject to destruction by overwrite by the TOE (whether or not the value is subsequently encrypted for storage in volatile or non-volatile memory). In the case where the only selection made for the destruction method key was removal of power, then this test is unnecessary. The evaluator will: 1. Record the value of the key in the TOE subject to clearing.

	2. Cause the TOE to perform a normal cryptographic processing with the key from Step #1. 3. Cause the TOE to clear the key. 4. Cause the TOE to stop the execution but not exit. 5. Cause the TOE to dump the entire memory of the TOE into a binary file. 6. Search the content of the binary file created in Step #5 for instances of the known key value from Step #1 Steps 1-6 ensure that the complete key does not exist anywhere in volatile memory. If a copy is found, then the test fails. ***TD0365 Applied***
Test Steps	SSHS: • Start debugging the sshd file stored on the TOE. • Set breakpoints and start the SSH service on 2220 port. • Continue to breakpoint 1 to print key values and memory address of keys. • Print keys and memory address of keys in next breakpoint • Print the next key values. • Print memory address of above keys and continue to next breakpoint. • Print the next key values. • Print memory address of above keys. • Print the next keys. • Print memory address of the above keys and move to next breakpoint. Server • Convert all the gcore-dump files to hex files and store it in a directory • Search for the following keys in all files and verify that the keys are not found in zeroized files. SSHC: • Start debugging the ssh file on the TOE. • Set breakpoints and start a SSHC session from the TOE to a server. • Reach the end of Breakpoint and print key values. • Print memory location where these keys are stored. • Continue to next breakpoint. • Print the next key values. • Print the memory location where the keys are stored. • Print the next keys values. • Print memory address where keys are stored. • Print the next key values. • Print memory address of keys and continue to next breakpoint. • Print key values and memory address of keys in last breakpoint. • Continue to next breakpoint and exit SSHC session. • Verify the keys were deleted by printing values present on memory address used by keys earlier. • Show gcore files saved on the TOE and convert them to hexadecimal format. • Search for the following keys in all files and verify that the keys are not found in zeroized files. TLSC: • Start debugging the openssl file on the TOE. • Set Breakpoints and initiate a connection to TLS server:

	- Reach Breakpoint 1 and print keys. - Print next keys and see the keys stored on relevent memory location. - Continue to next breakpoint. - Print key values and show them saved in relevant memory address. - Continue to last breakpoint. - Server. - Reach the end of breakpoint. - Verify that the keys were zeroized by comparing memory addresses and quit GDB. - Convert gcore-dump files to hex files and store it in conv_hex directory. - Search for the following key values in all .hex files and verify that the keys are not found in zeroized file.
Expected Test Results	The TOE should properly destroy keys.
Pass/Fail with Explanation	Pass. The TOE behaves as expected when the cryptographic keys are removed from the non-volatile memory. This satisfies the testing requirements.

7.2.1.4 FCS_CKM.4 Test #2

Item	Data
Test Assurance Activity	Test 2: Applied to each key help in non-volatile memory and subject to destruction by the TOE. The evaluator will use special tools (as needed), provided by the TOE developer if necessary, to ensure the tests function as intended. 1. Identify the purpose of the key and what access should fail when it is deleted. (e.g. the data encryption key being deleted would cause data decryption to fail.) 2. Cause the TOE to clear the key. 3. Have the TOE attempt the functionality that the cleared key would be necessary for. The test succeeds if step 3 fails. ***TD0365 Applied***
Test Steps	SSHC - Delete the keys used for that process. - Attempt a connection to the TOE and verify the connection fails. SSHS - Create a key pair on TOE. - Copy the public key to VM. - Delete the private key from TOE. - Try connecting TOE to the VM and verify the connection fails. TLSC - Delete the Root certificate from TOE. - Verify the connection fails when TOE tries to connect the openssl server (VM).
Expected Test Results	The TOE should not be allowed to perform a function that relies on keys that have been removed.
Pass/Fail with Explanation	Pass. The TOE behaves as expected when the cryptographic keys are removed from the non-volatile memory. This satisfies the test requirements

7.2.1.5 FCS_CKM.4 Test #3

Item	Data

Test Assurance Activity	**Tests 3 and 4** do not apply for the selection **instructing the underlying platform to destroy the representation of the key**, as the TOE has no visibility into the inner workings and completely relies on the underlying platform. **Test 3:** The following tests are used to determine the TOE is able to request the platform to overwrite the key with a TOE supplied pattern. Applied to each key held in non-volatile memory and subject to destruction by overwrite by the TOE. The evaluator will use a tool that provides a logical view of the media (e.g., MBR file system): 1. Record the value of the key in the TOE subject to clearing. 2. Cause the TOE to perform a normal cryptographic processing with the key from Step #1. 3. Cause the TOE to clear the key. 4. Search the logical view that the key was stored in for instances of the known key value from Step #1. If a copy is found, then the test fails. ***TD0365 Applied***
Test Steps	SSHC • Create a keypair on the TOE • Verify the keys are stored on the TOE • Copy the public key to VM • Display value of private key before shredding • Perform a SSH connection and verify it successfully connected • Shred the private key thrice: • Verify that the key value has been changed Verify that the ssh connection failed when shredded key was used SSHS • Create a keypair on the VM • Verify the keys are stored on the VM • Copy the public key to TOE • Display value of public key before shredding • Perform a SSH connection and verify it successfully connected • Shred the private key thrice • Verify that the public key value has been changed • Verify that the ssh connection failed when shredded key was used TLSC: • Show ICA1 chain present on TOE • Initiate an TLSC session using the same certificate chain Server • Shred the ICA1.pem certificate. • Show the value of certificate after shredding • Search for key value in ICA1.pem and verify it is not found • Try a TLSC connection and verify it fails
Expected Test Results	The TOE should be able to request the platform to overwrite the key.
Pass/Fail with Explanation	Pass. TOE was able to request the platform to overwrite the key. This satisfies the testing requirements

7.2.1.6 FCS_CKM.4 Test #4

Item	**Data**

Test Assurance Activity	**Tests 3 and 4** do not apply for the selection **instructing the underlying platform to destroy the representation of the key**, as the TOE has no visibility into the inner workings and completely relies on the underlying platform. **Test 4:** Applied to each key held as non-volatile memory and subject to destruction by overwrite by the TOE. The evaluator will use a tool that provides a logical view of the media: 1. Record the logical storage location of the key in the TOE subject to clearing. 2. Cause the TOE to perform a normal cryptographic processing with the key from Step #1. 3. Cause the TOE to clear the key. 4. Read the logical storage location in Step #1 of non-volatile memory to ensure the appropriate pattern is utilized. The test succeeds if correct pattern is used to overwrite the key in the memory location. If the pattern is not found the test fails. ***TD0365 Applied***
Pass/Fail with Explanation	Pass. *This test is satisfied by FCS_CKM.4 Test#3*

7.2.2 SSH

7.2.2.1 FCS_SSH_EXT.1.2Test#1

Item	**Data**
Test Assurance Activity	Test 1: [conditional] If the TOE is acting as SSH Server: a. The evaluator shall use a suitable SSH Client to connect to the TOE, enable debug messages in the SSH Client, and examine the debug messages to determine that only the configured authentication methods for the TOE were offered by the server.
Test Steps	<u>Password based Authentication:</u> • Attempt a connection to the TOE. • Verify that only the configured authentication methods were offered. • Verify that the connection succeeds. • Verify the connection succeeds using TOE logs <u>Public key-based Authentication:</u> ecdsa-sha2-nistp256 • Create a key pair on the TOE • Copy the public key to the TOE • Attempt to connect to the TOE from the VM • Verify the connection succeeds • Verify the connection succeeds using TOE logs rsa-sha2-256: • Create a key pair on the TOE • Copy the public key to the TOE • Attempt to connect to the TOE from the VM

	• Verify the connection succeeds • Verify the connection succeeds using TOE logs rsa-sha2-512: • Create a key pair on the TOE • Copy the public key to the TOE • Attempt to connect to the TOE from the VM • Verify the connection succeeds • Verify the connection succeeds using TOE logs ecdsa-sha2-nistp384: • Create a key pair on the TOE • Copy the public key to the TOE • Attempt to connect to the TOE from the VM • Verify the connection succeeds • Verify the connection succeeds using TOE logs ecdsa-sha2-nistp521: • Create a key pair on the TOE • Copy the public key to the TOE • Attempt to connect to the TOE from the VM • Verify the connection succeeds • Verify the connection succeeds using TOE logs
Expected Test Results	The TOE should only offer the configured authentication methods to a SSH client.
Pass/Fail with Explanation	Pass. The TOE only offers configured authentication methods to a SSH client. This satisfies the test requirements.

7.2.2.2 FCS_SSH_EXT.1.2Test#2

Item	**Data**
Test Assurance Activity	Test 2: [conditional] If the TOE is acting as SSH Client, the evaluator shall test for a successful configuration setting of each authentication method as follows: a. The evaluator shall initiate a SSH session using the authentication method configured and verify that the session is successfully established. b. Next, the evaluator shall use bad authentication data (e.g. incorrectly generated certificate or incorrect password) and ensure that the connection is rejected. Steps a-b shall be repeated for each independently configurable authentication method supported by the server.
Test Steps	Password based Authentication: Good credentials – SSH connection successful • Attempt SSH connection to the VM from the TOE • Verify that the password was accepted • Verify the connection was successful • Verify the connection succeeds using server logs Bad Credentials – SSH connection unsuccessful • Attempt SSH connection to the VM from the TOE

- Verify that the bad password was not accepted, and the connection was unsuccessful
- Verify that the connection failed using Server logs

Public-key Authentication
Good credentials – SSH connection successful
ecdsa-sha2-nistp256

- Create a key pair on the TOE
- Copy the public key to the SSH server (VM)
- Attempt to connect to the VM from the TOE
- Verify the connection succeeds
- Verify the connection succeeds using server logs

rsa-sha2-256:

- Create a key pair on the TOE
- Copy the public key to the SSH server (VM)
- Attempt to connect to the VM from the TOE
- Verify the connection succeeds
- Verify the connection succeeds using server logs

rsa-sha2-512

- Create a key pair on the TOE
- Copy the public key to the SSH server (VM)
- Attempt to connect to the VM from the TOE
- Verify the connection succeeds
- Verify the connection succeeds using server logs

ecdsa-sha2-nistp384

- Create a key pair on the TOE
- Copy the public key to the SSH server (VM)
- Attempt to connect to the VM from the TOE
- Verify the connection succeeds
- Verify the connection succeeds using server logs

ecdsa-sha2-nistp521

- Create a key pair on the TOE
- Copy the public key to the SSH server (VM)
- Attempt to connect to the VM from the TOE
- Verify the connection succeeds
- Verify the connection succeeds using server logs

Bad Credentials – SSH connection unsuccessful
ecdsa-sha2-nistp256

- Create a key pair on the TOE but do not copy the public key on the Test VM
- Attempt to connect to the VM from the TOE
- Verify that the connection fails
- Verify that the connection failed using server logs

rsa-sha2-256:

	• Create a key pair on the TOE but do not copy the public key on the Test VM • Attempt to connect to the VM from the TOE • Verify that the connection fails • Verify that the connection failed using server logs rsa-sha2-512 • Create a key pair on the TOE but do not copy the public key on the Test VM • Attempt to connect to the VM from the TOE • Verify that the connection fails • Verify that the connection failed using server logs ecdsa-sha2-nistp384 • Create a key pair on the TOE but do not copy the public key on the Test VM • Attempt to connect to the VM from the TOE • Verify that the connection fails • Verify that the connection failed using server logs ecdsa-sha2-nistp521 • Create a key pair on the TOE but do not copy the public key on the Test VM • Attempt to connect to the VM from the TOE • Verify that the connection fails • Verify that the connection failed using server logs
Expected Test Results	The SSH server should allow TOE to connect only when good credentials are used.
Pass/Fail with Explanation	Pass. The SSH server allows TOE to connect only when the credentials are good. This satisfies the test requirements.

7.2.2.3 FCS_SSH_EXT.1.2Test#3

Item	**Data**
Test Assurance Activity	Test 3: [conditional] If the TOE is acting as SSH Client, the evaluator shall verify that the connection fails upon configuration mismatch as follows: a. The evaluator shall configure the Client with an authentication method not supported by the Server. b. The evaluator shall verify that the connection fails. If the Client supports only one authentication method, the evaluator can test this failure of connection by configuring the Server with an authentication method not supported by the Client. In order to facilitate this test, it is acceptable for the evaluator to configure an authentication method that is outside of the selections in the SFR.
Test Steps	• Configure the VM with an authentication method not supported by the TOE • Initiate a SSH session from the TOE • Verify the connection fails • Verify that the connection fails using logs
Expected Test Results	The TOE should not connect to the SSH Server if the authentication method is not supported by the TOE
Pass/Fail with Explanation	Pass. The TOE does not connect to the SSH Server if the authentication method is not supported by the TOE. This satisfies the test requirements.

7.2.2.4 FCS_SSH_EXT.1.3Test#1

Item	**Data**

Test Assurance Activity	Test 1: The evaluator shall demonstrate that the TOE accepts the maximum allowed packet size.
Test Steps	SSHS - Start the acumen-sshs tool on VM to send a packet of 262144 bytes to the TOE. - Using packet capture verifies that the connection succeeds. - Verify the connection was established using logs. SSHC - Use Acumen-sshc tool to make a connection with maximum allowed packet size. - Verify the connection was successful using a packet capture.
Expected Test Results	The TOE should allow SSH packet only up to the maximum size. (Here 262144 bytes)
Pass/Fail with Explanation	Pass. The TOE allowed SSH packet of maximum size it is configured for. This satisfies the test requirement.

7.2.2.5 FCS_SSH_EXT.1.3Test#2

Item	**Data**
Test Assurance Activity	Test 2: This test is performed to verify that the TOE drops packets that are larger than size specified in the component. a. The evaluator shall establish a successful SSH connection with the peer. b. Next the evaluator shall craft a packet that is ~~one byte~~ slightly larger than the maximum size specified in this component and send it through the established SSH connection to the TOE. The packet should not be greater than the maximum packet size + 16 bytes. If the packet is larger, the evaluator shall justify the need to send a larger packet. c. The ~~E~~evaluator shall verify that the packet was dropped by the TOE. The method of verification will vary by the TOE. Examples include ~~by~~ reviewing the TOE audit log for a dropped packet audit or observing the TOE terminates the connection. ***TD0732 Applied***
Test Steps	TOE as Server - Start the acumen-sshs tool on VM to send a packet larger than configured for TOE. - Verify that the TOE disconnects the SSH session using packet capture. - Verify that the TOE closes connection when it receives an error form the TOE using logs. TOE as Client - Start an SSH connection using Acumen SSHC tool. - Verify the connection failed using packet capture.
Expected Test Results	The TOE should drop connection when the packet size is larger than specified.
Pass/Fail with Explanation	Pass. The TOE drops connections when the packet size is larger than specified. The TOE satisfies the test requirements.

7.2.2.6 FCS_SSH_EXT.1.4Test#1

Item	**Data**

Test Assurance Activity	Test 1: The evaluator must ensure that only claimed algorithms and cryptographic primitives are used to establish an SSH connection. To verify this, the evaluator shall establish an SSH connection with a remote endpoint. The evaluator shall capture the traffic exchanged between the TOE and the remote endpoint during protocol negotiation (e.g. using a packet capture tool or information provided by the endpoint, respectively). The evaluator shall verify from the captured traffic that the TOE offers only the algorithms defined in the ST for the TOE for SSH connections. The evaluator shall perform one successful negotiation of an SSH connection and verify that the negotiated algorithms were included in the advertised set. If the evaluator detects that not all algorithms defined in the ST for SSH are advertised by the TOE or the TOE advertises additional algorithms not defined in the ST for SSH, the test shall be regarded as failed. The data collected from the connection above shall be used for verification of the advertised hashing and shared secret establishment algorithms in FCS_SSH_EXT.1.5 and FCS_SSH_EXT.1.6 respectively.
Test Steps	TOE as a server: • Initiate a SSH session from the Test VM. • Verify that the correct encryption algorithms were advertised during the connection process using packet capture. • Verify the connection was successful using TOE logs TOE as a client: • Initiate a SSH session from the TOE to the Test VM. • Verify that the correct encryption algorithms were advertised during the connection process using packet capture • Verify the connection was successful using server logs
Expected Test Results	TOE as Client: The TOE should be able to successfully connect to a SSH server using each of the encryption algorithm selected in the ST. TOE as Server: The TOE should allow a SSH client to connect when using each of the encryption algorithms selected in the ST.
Pass/Fail with Explanation	Pass. The TOE was able to use all claimed encryption algorithms. This satisfies the test requirements.

7.2.2.7 FCS_SSH_EXT.1.4Test#2

Item	Data
Test Assurance Activity	Test 2: For the connection established in Test 1, the evaluator shall terminate the connection and observe that the TOE terminates the connection.
Test Steps	TOE as a server: • End the SSH connection. • Verify the session terminated using packet capture. • Verify the session terminated using TOE logs. TOE as a client: • End the SSH connection. • Verify the session terminated using packet capture • Verify the session terminated using Server logs
Expected Test Results	The TOE should be able to terminate the connection.

Pass/Fail with Explanation	Pass. The TOE successfully terminates the connection. This satisfies the testing requirement.

7.2.2.8 FCS_SSH_EXT.1.4Test#3

Item	Data
Test Assurance Activity	Test 3: The evaluator shall configure the remote endpoint to only allow a mechanism that is not included in the ST selection. The evaluator shall attempt to connect to the TOE and observe that the attempt fails.
Test Steps	TOE as a client: • Configure the SSH server to allow encryption algorithms not allowed in the ST • Attempt a connection from the TOE to the VM • Verify the connection is not initiated • Verify the connection failed using server logs. TOE as a server: • Configure the SSH client to offer encryption algorithms not allowed in the ST • Attempt a SSH connection from the VM to the TOE • Verify the connection is not initiated • Verify the connection failed using TOE logs.
Expected Test Results	The TOE should not connect with any remote endpoint offering encryption algorithms other than the ones specified in ST
Pass/Fail with Explanation	Pass. The TOE was able to terminate the session. This satisfies the test requirements.

7.2.2.9 FCS_SSH_EXT.1.5Test#1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall use the test data collected in FCS_SSH_EXT.1.4, Test 1 to verify that appropriate mechanisms are advertised.
Test Steps	TOE as a Client: • Verify that appropriate mechanisms are advertised. • Verify the connection was successful using server logs TOE as a Server: • Verify that appropriate mechanisms are advertised. • Verify the connection was successful using TOE logs
Expected Test Results	The TOE should advertise appropriate hashing algorithms
Pass/Fail with Explanation	Pass. Appropriate hashing algorithms were advertised by the TOE. The TOE satisfies the test requirements.

7.2.2.10 FCS_SSH_EXT.1.5Test#2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall configure an SSH peer to allow only a hashing algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection and observe that the connection is rejected.
Test Steps	TOE as a Client: • Configure the VM to offer a hashing algorithm not included in the ST • Attempt to connect to the TOE from a SSH peer offering incorrect MAC algorithm. • Verify that the connection failed

	- Verify that the connection failed using server logs TOE as server: - Configure the VM to offer a hashing algorithm not included in the ST - Attempt to connect to the TOE from a SSH peer offering incorrect MAC algorithm. - Verify that the connection failed - Verify that the connection failed using TOE logs
Expected Test Results	The TOE should not connect to a SSH peer which offers hashing algorithm not included in ST.
Pass/Fail with Explanation	Pass. The TOE did not connect to a SSH peer which offers hashing algorithm not included in ST. This satisfies the test requirements.

7.2.2.11 FCS_SSH_EXT.1.6Test#1

Item	**Data**
Test Assurance Activity	Test 1: The evaluator shall use the test data collected in FCS_SSH_EXT.1.4, Test 1 to verify that appropriate mechanisms are advertised.
Test Steps	TOE as a client: - Verify that the appropriate key exchange algorithms were advertised. - Verify the connection was successful using server logs Toe as a server: - Verify that the appropriate key exchange algorithms were advertised. - Verify the connection was successful using TOE logs
Expected Test Results	The TOE should advertise key exchange algorithm that are mentioned in the ST.
Pass/Fail with Explanation	Pass. The TOE offers key-exchange algorithms only mentioned in the ST. This satisfies the test requirements.

7.2.2.12 FCS_SSH_EXT.1.6Test#2

Item	**Data**
Test Assurance Activity	Test 2: The evaluator shall configure an SSH peer to allow only a key exchange method that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection and observe that the connection is rejected.
Test Steps	TOE as a server: - Configure the VM to offer key exchange algorithm not included in the ST. - Attempt connection to TOE from the peer(VM) - Verify the connection attempt fails - Verify the attempt fails using TOE logs. TOE as a client: - Configure the Client to offer key exchange algorithm not mentioned in the ST. - Attempt a connection from the TOE to the VM - Verify the connection attempt fails - Verify the attempt fails using server logs.
Expected Test Results	The TOE should not connect to remote peer when a key-exchange method not included in ST is used.
Pass/Fail with Explanation	Pass. The TOE does not connect to remote peer when a key-exchange method not included in ST is used. This satisfies the test requirements.

7.2.2.13 FCS_SSH_EXT.1.8Test#1

Item	Data
Test Assurance Activity	The test harness needs to be configured so that its connection rekey or termination limits are greater than the limits supported by the TOE -- it is expected that the test harness should not be initiating the connection rekey or termination. Test 1: Establish an SSH connection. Wait until the identified connection rekey limit is met. Observed that a connection rekey or termination is initiated. This may require traffic to periodically be sent, or connection keep alive to be set, to ensure that the connection is not closed due to an idle timeout.
Pass/Fail with Explanation	Pass. *Refer to test FCS_SSH_EXT.1.8 Test#2 and FCS_SSH_EXT.1.8Test #3*

7.2.2.14 FCS_SSH_EXT.1.8Test#2

Item	Data
Test Assurance Activity	The test harness needs to be configured so that its connection rekey or termination limits are greater than the limits supported by the TOE -- it is expected that the test harness should not be initiating the connection rekey or termination. Test 2: Establish an SSH connection. Transmit data from the TOE until the identified connection rekey or termination limit is met. Observe that a connection rekey or termination is initiated.
Test Steps	Data: • Use the acumen-sshs tool to send traffic from the TOE to the VM • Verify that the rekeying occurs • Verify the connection was successful using TOE logs. Time: • Use the acumen-sshs tool to send traffic from the TOE to the VM for 1 hour • Verify that the rekeying occurs • Verify the connection was successful using TOE logs.
Expected Test Results	The TOE should properly rekey when the rekey limits are reached.
Pass/Fail with Explanation	Pass. The TOE properly rekeys when the rekey limits are reached. This passes the test requirements.

7.2.2.15 FCS_SSH_EXT.1.8Test#3

Item	Data
Test Assurance Activity	The test harness needs to be configured so that its connection rekey or termination limits are greater than the limits supported by the TOE -- it is expected that the test harness should not be initiating the connection rekey or termination. Test 3: Establish an SSH connection. Send data to the TOE until the identified connection rekey limit or termination is met. Observe that a connection rekey or termination is initiated.
Test Steps	Time • Use the acumen-sshc tool to cause a rekey based on time (1hr). • Initiate connection from TOE to the VM to start a SSH session. • Verify that the rekeying occurs.

	Data • Use the acumen-sshc tool to cause a rekey based on data. • Initiate connection from TOE to the VM to start a SSH session. • Verify that the rekeying occurs.
Expected Test Results	The TOE should properly rekey when the rekey limits are reached.
Pass/Fail with Explanation	Pass. The TOE properly rekeys for time and data. This satisfies the test requirements.

7.2.3 SSHC

7.2.3.1 FCS_SSHC_EXT.1.1Test#1

Item	**Data**
Test Assurance Activity	Test 1: [conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall configure the TOE with only a single host name and corresponding public key in the local database. The evaluator shall verify that the TOE can successfully connect to the host identified by the host name.
Test Steps	• Show the host key saved in /etc/ssh directory on the VM • Save the public host key on the TOE • Connect TOE to the VM • Verify via logs and debug messages
Expected Test Results	The TOE should successfully connect to the host identified by the host name.
Pass/Fail with Explanation	Pass. The TOE successfully connects to the host identified by the host name. This satisfies the test requirements.

7.2.3.2 FCS_SSHC_EXT.1.1Test#2

Item	Data
Test Assurance Activity	Test 2: [conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall configure the TOE with only a single host name and non-corresponding public key in the local database. The evaluator shall verify that the TOE fails to connect to a host not identified by the host name.
Test Steps	• Show host name and corresponding public-key saved on the TOE -Before -After • Configure the SSH server to offer a different host-key. • Initiate a SSH connection from the TOE to the VM: • Verify that the SSH host key verification fails:
Expected Test Results	The TOE should fail to connect to a host not identified by the host name.
Pass/Fail with Explanation	Pass. The TOE fails to connect to a host not identified by the host name. This satisfies the test requirements.

7.2.3.3 FCS_SSHC_EXT.1.1Test#3

Item	**Data**
Test Assurance Activity	Test 3: [conditional] If using a local database by associating each host name with its corresponding public key, the evaluator shall try to connect to a host not configured in

	the local database. The evaluator shall verify that the TOE either fails to connect to a host identified by the host name or there is a prompt provided to store the public key in the local database.
Test Steps	• Show the hosts and corresponding public key saved on the TOE • Try connecting TOE to the VM using VM's secondary IP address • Verify that it fails
Expected Test Results	The TOE should either fail to connect to the VM or provide a prompt to store the public key in local database.
Pass/Fail with Explanation	Pass. The TOE provides option to store public key in local database. This satisfies the test requirements.

7.2.3.4 FCS_SSHC_EXT.1.1Test#4

Item	**Data**
Test Assurance Activity	Test 4: [conditional] If using a list of trusted certification authorities, the evaluator shall configure the TOE with only a single trusted certification authority corresponding to the host. The evaluator shall verify that the TOE can successfully connect to the host identified by the host name.
Pass/Fail with Explanation	NA. TOE does not support certificate-based authentication.

7.2.3.5 FCS_SSHC_EXT.1.1Test#5

Item	**Data**
Test Assurance Activity	Test 5: [conditional] If using a list of trusted certification authorities, the evaluator shall configure the TOE with only a single trusted certification authority that does not correspond to the host. The evaluator shall verify that the TOE fails to the host identified by the host name.
Pass/Fail with Explanation	NA. TOE does not support certificate-based authentication.

7.2.4 SSHS

7.2.4.1 FCS_SSHS_EXT.1.1Test#1

Item	**Data**
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall use a suitable SSH Client to connect to the TOE and examine the list of server host key algorithms in the SSH_MSG_KEXINIT packet sent from the server to the client to determine that only the configured server authentication methods for the TOE were offered by the server. ***TD0682 Applied***
Test Steps	• Attempt a SSH connection to the TOE using the ecdsa-sha2-nistp256 public key algorithm: • Verify that the connection succeeded. • Verify that the correct host-key algorithms were displayed in the packet capture.
Expected Test Results	The TOE should be able to have a SSH client connect to it with all the supported authentication mechanism.

Pass/Fail with Explanation	Pass. The TOE was successfully able to connect to SSH client with all the supported authentication mechanism. This satisfies the test requirement.

7.2.4.2 FCS_SSHS_EXT.1.1Test#2

Item	**Data**
Test Assurance Activity	Test 2: The evaluator shall test for a successful configuration setting of each server authentication method as follows. The evaluator shall initiate a SSH session using the authentication method configured and verify that the session is successfully established. Repeat this process for each independently configurable server authentication method supported by the server. ***TD0682 Applied***
Test Steps	Password based Authentication: • Attempt a connection from VM to the TOE • Verify that the password was accepted • Verify the connection was successful Public-key Authentication ecdsa-sha2-nistp256 • Create a key pair on the VM • Copy the public key to the TOE • Attempt a connection to the TOE from the VM • Verify the connection succeeds rsa-sha2-256: • Create a key pair on the VM • Copy the public key to the TOE • Attempt a connection to the TOE from the VM • Verify the connection succeeds rsa-sha2-512 • Create a key pair on the VM • Copy the public key to the TOE • Attempt a connection to the TOE from the VM • Verify the connection succeeds ecdsa-sha2-nistp384 • Create a key pair on the VM • Copy the public key to the TOE • Attempt a connection to the TOE from the VM • Verify the connection succeeds ecdsa-sha2-nistp521 • Create a key pair on the VM • Copy the public key to the TOE • Attempt a connection to the TOE from the VM • Verify the connection succeeds
Expected Test Results	The TOE should be able to terminate the connection.

Pass/Fail with Explanation	Pass. The TOE was able to terminate sessions. This satisfies the test requirement.

7.2.4.3 FCS_SSHS_EXT.1.1Test#3

Item	Data
Test Assurance Activity	Test 3: ~~Next t~~The evaluator shall configure the ~~remote~~ peer to only allow an authentication mechanism that is not included in the ST selection. The evaluator shall attempt to connect to the TOE and observe that the ~~attempt fails~~ TOE sends a disconnect message. ***TD0682 Applied***
Test Steps	Password Authentication. • Configure the VM with an authentication method not supported by the TOE • Initiate a SSH session from the TOE • Verify the connection fails Public Key Authentications: • Configure the VM with an authentication method not supported by the TOE • Initiate a SSH session from the TOE • Verify the connection fails
Expected Test Results	The TOE should not connect with a peer configured with authentication mechanism not specified in the ST.
Pass/Fail with Explanation	Pass. The TOE does not connect with a peer configured with authentication mechanism not specified in the ST. The TOE pass the test requirement.

7.2.5 TLSC

7.2.5.1 FCS_TLSC_EXT.1.1Test#1

Item	Data
Test Assurance Activity	• **Test 1:** The evaluator will establish a TLS connection using each of the cipher suites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an EAP session. It is sufficient to observe the successful negotiation of a cipher suite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the cipher suite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Test Steps	For each TLS cipher suites selected in the ST • The evaluator started a TLS server on the test VM and specified the cipher suite • The evaluator attempted a connection from the TOE to the TLS server • The evaluator then verified that the correct cipher suite was used, and the connection was successful
Expected Test Results	The TOE should connect to a TLS server using each of the cipher suites selected in the ST.
Pass/Fail with Explanation	Pass. The TOE connects to a remote TLS server using the claimed ciphersuites. This meets the testing requirements.

7.2.5.2 FCS_TLSC_EXT.1.1Test#2

Item	Data
Test Assurance Activity	- **Test 2:** The evaluator will attempt to establish the connection using a server with a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field and verify that a connection is established. The evaluator will then verify that the client rejects an otherwise valid server certificate that lacks the Server Authentication purpose in the extendedKeyUsage field and a connection is not established. Ideally, the two certificates should be identical except for the extendedKeyUsage field.
Test Steps	Certificate containing Server Authentication purpose in the extendedKeyUsage field. - The evaluator started a TLS server on the test VM using a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection was successful Certificate not containing Server Authentication purpose in the extendedKeyUsage field. - The evaluator started a TLS server on the test VM using a server certificate that does not contain the Server Authentication purpose in the extendedKeyUsage field. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should connect to a TLS server that is using a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field and not connect to a TLS server that does not contain the Server Authentication purpose in the extendedKeyUsage field.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that does not have the Server Authentication purpose in the extendedKeyUsage field.

7.2.5.3 FCS_TLSC_EXT.1.1Test#3

Item	Data
Test Assurance Activity	- **Test 3:** The evaluator will send a server certificate in the TLS connection that does not match the server-selected cipher suite (for example, send a ECDSA certificate while using the TLS_RSA_WITH_AES_128_CBC_SHA cipher suite or send a RSA certificate while using one of the ECDSA cipher suites.) The evaluator will verify that the OS disconnects after receiving the server's Certificate handshake message.
Test Steps	- The evaluator created a TLS server on the test VM using the acumen-tlsc-v2.2e tool, using an RSA certificate and specifying an ECDSA cipher suite. - The evaluator then attempted to connect to the server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should disconnect from the TLS server when the server certificate does not match the server-selected cipher.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server whose certificate type does not match the cipher suite.

7.2.5.4 FCS_TLSC_EXT.1.1Test#4

Item	Data

Test Assurance Activity	- **Test 4:** The evaluator will configure the server to select the TLS_NULL_WITH_NULL_NULL cipher suite and verify that the client denies the connection.
Test Steps	- The evaluator started a TLS server on the test VM using the acumen-tlsc-2.2e tool. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server that is using the TLS_NULL_WITH_NULL_NULL cipher suite.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server using TLS_NULL_WITH_NULL_NULL cipher suite.

7.2.5.5 FCS_TLSC_EXT.1.1Test#5.1

Item	**Data**
Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.1:** Change the TLS version selected by the server in the Server Hello to a non-supported TLS version (for example 1.3 represented by the two bytes 03 04) and verify that the client rejects the connection**.**
Test Steps	- The evaluator started a TLS server using the acumen-tlsc-test tool on the test VM that will send an unsupported TLS version. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should reject a connection to a TLS server that has selected an unsupported version of TLS.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that is using an unsupported version of TLS.

7.2.5.6 FCS_TLSC_EXT.1.1Test#5.2

Item	**Data**
Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.2:** Modify at least one byte in the server's nonce in the Server Hello handshake message, and verify that the client rejects the Server Key Exchange handshake message (if using a DHE or ECDHE cipher suite) or that the server denies the client's Finished handshake message.
Test Steps	- The evaluator started the acumen-tlsc-test tool on the test VM to modify a byte in the servers nonce. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection when at least one byte in the server’s nonce in the Server Hello handshake message is modified.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that has had it’s nonce modified.

7.2.5.7 FCS_TLSC_EXT.1.1Test#5.3

Item	**Data**

Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.3:** Modify the server's selected cipher suite in the Server Hello handshake message to be a cipher suite not presented in the Client Hello handshake message. The evaluator will verify that the client rejects the connection after receiving the Server Hello.
Test Steps	- The evaluator used the acumen-tlsc-test tool to start a TLS server on the test VM that will change the cipher suite. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when the server's selected cipher suite in the Server Hello handshake message to be a cipher suite not presented in the Client Hello handshake message.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that has had it's cipher suite changed.

7.2.5.8 FCS_TLSC_EXT.1.1Test#5.4

Item	Data
Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.4:** If an ECDHE or DHE ciphersuite is selected, modify the signature block in the Server's Key Exchange handshake message, and verify that the client rejects the connection after receiving the Server Key Exchange message.
Test Steps	- The evaluator started the acumen-tlsc-test tool on the test VM that will modify the signature block. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when the signature block in the Server's Key Exchange handshake message is modified.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that has had it's signature block changed.

7.2.5.9 FCS_TLSC_EXT.1.1Test#5.5

Item	Data
Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.5:** Modify a byte in the Server Finished handshake message, and verify that the client sends a fatal alert upon receipt and does not send any application data.
Test Steps	- The evaluator started the acumen-tlsc-test tool in the test VM that will modify a byte in the server finished handshake message. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when a byte in the Server Finished handshake message is modified.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that has had it's server finished handshake message modified.

7.2.5.10 FCS_TLSC_EXT.1.1Test#5.6

Item	Data
Test Assurance Activity	- **Test 5:** The evaluator will perform the following modifications to the traffic: ○ **Test 5.6:** Send a garbled message from the Server after the Server has issued the Change Cipher Spec message and verify that the client denies the connection.
Test Steps	- The evaluator started the acumen-tlsc-test tool on the test VM so that a garbled message would be sent after the Change Cipher Spec message. - The evaluator then attempted to connect to the server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when is receives a garbled message from the Server after the Server has issued the Change Cipher Spec.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server that has sent a garbled message.

7.2.5.11 FCS_TLSC_EXT.1.2Test#1

Item	Data
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 1:** The evaluator will present a server certificate that does not contain an identifier in either the Subject Alternative Name (SAN) or Common Name (CN) that matches the reference identifier. The evaluator will verify that the connection fails.
Test Steps	- The evaluator started a TLS server on the test VM with a certificate that does not contain an identifier in either the Subject Alternative Name (SAN) or the Common Name (CN) that matches the reference identifier. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when the server certificate does not contain and identifier in either the SAN or CN that matched the reference identifier.
Pass/Fail with Explanation	Pass. The TOE rejects a connection if the CN and SAN have a value that does not match the reference identifier.

7.2.5.12 FCS_TLSC_EXT.1.2Test#2

Item	Data
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 2:** The evaluator will present a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. The evaluator shall verify that the connection fails. The evaluator will repeat this test for each supported SAN type.
Test Steps	IP: - The evaluator started a TLS server on the test VM with a certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. - The evaluator then attempted to connect to the server from the TOE.

	- The evaluator then verified that the connection failed. DNS: - The evaluator started a TLS server on the test VM with a certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. - The evaluator then attempted to connect to the server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should deny a connection to a TLS server when the server certificate contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier.
Pass/Fail with Explanation	Pass. The TOE rejects a connection if the CN matches and SAN does not match the reference identifier.

7.2.5.13 FCS_TLSC_EXT.1.2Test#3

Item	**Data**
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 3:** [conditional] If the TOE does not mandate the presence of the SAN extension, the evaluator will present a server certificate that contains a CN that matches the reference identifier and does not contain the SAN extension. The evaluator will verify that the connection succeeds. If the TOE mandates the presence of the SAN extension, this test shall be omitted.
Test Steps	DNS: - The evaluator started a TLS server on the test VM that uses a certificate that contains a CN that matches the reference identifier and does not contain the SAN extension. - The evaluator then attempted to connect to the TLS server from the TOE.
Expected Test Results	The TOE should connect to a TLS server with a server certificate that contains a CN that matches the reference identifier and does not contain the SAN extension.
Pass/Fail with Explanation	Pass. The TOE connects to a server if the CN matches the reference identifier and the SAN is missing.

7.2.5.14 FCS_TLSC_EXT.1.2Test#4

Item	**Data**
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 4:** The evaluator will present a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. The evaluator will verify that the connection succeeds.
Test Steps	IP: - The evaluator started a TLS server on the test VM with a certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. - The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection was successful. DNS:

	• The evaluator started a TLS server on the test VM with a certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. • The evaluator then attempted to connect to the TLS server from the TOE. • The evaluator then verified that the connection was successful.
Expected Test Results	The TOE should connect to a TLS server with a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches.
Pass/Fail with Explanation	Pass. The TOE connects to a server if the CN does not match the reference identifier and the SAN does match.

7.2.5.15 FCS_TLSC_EXT.1.2Test#5.1

Item	Data
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: • **Test 5:** The evaluator will perform the following wildcard tests with each supported type of reference identifier: ○ **Test 5.1:** The evaluator will present a server certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.example.com) and verify that the connection fails.
Test Steps	SAN: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g., foo.*.example.com). • The evaluator then attempted to connect to the server from the TOE. • The evaluator then verified that the connection fails. CN: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g., foo.*.example.com). • The evaluator then attempted to connect to the server from the TOE. • The evaluator then verified that the connection fails.
Expected Test Results	The TOE should not connect to a TLS server with a server a certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.example.com).
Pass/Fail with Explanation	Pass. The TOE rejects a connection if the server certificate contains a wildcard that is not in the left-most label.

7.2.5.16 FCS_TLSC_EXT.1.2Test#5.2

Item	Data
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: • **Test 5:** The evaluator will perform the following wildcard tests with each supported type of reference identifier: ○ **Test 5.2:** The evaluator will present a server certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g. *.example.com). The evaluator will configure the

	reference identifier with a single left-most label (e.g. foo.example.com) and verify that the connection succeeds. The evaluator will configure the reference identifier without a left-most label as in the certificate (e.g. example.com) and verify that the connection fails. The evaluator will configure the reference identifier with two left-most labels (e.g. bar.foo.example.com) and verify that the connection fails.
Test Steps	SAN: rhel.cctest.com identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *.example.com) • The evaluator then attempted to connect to the TLS server from the TOE. • The evaluator then verified that the connection succeeded. cctest.com reference identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *example.com) • The evaluator then attempted to connect to the server from the TOE. • The evaluator then verified that the connection failed. foo.rhel.cctest.com reference identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *example.com) • The evaluator then attempted to connect to the server from the TOE. • The evaluator then verified that the connection failed. CN: rhel.cctest.com identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *.example.com) • to connect to the TLS server from the TOE. • The evaluator then verified that the connection succeeded. cctest.com reference identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *example.com) • The evaluator then attempted to connect to the server from the TOE. • The evaluator then verified that the connection failed. foo.rhel.cctest.com reference identifier: • The evaluator started a TLS server on the test VM using a certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g., *example.com) • The evaluator then attempted to connect to the server from the TOE. a. The evaluator then verified that the connection failed.
Expected Test Results	The TOE should connect to a TLS server that is using a server certificate containing a wildcard in the left-most label but not preceding the public suffix (e.g. *.example.com) when the configured reference identifier is (e.g. foo.example.com) but not connect

	when the configured reference identifier is (e.g. example.com, and e.g. bar.foo.example.com).
Pass/Fail with Explanation	Pass. The TOE connects to a server that has a left-most wildcard (e.g. *.example.com) when using a reference identifier with a single left-most label(e.g. foo.example.com). And does not connect with a reference identifier without a left-most label (e.g. example.com) or a reference identifier with two left¬most labels (e.g. bar.foo.example.com).

7.2.5.17 FCS_TLSC_EXT.1.2Test#5.3

Item	**Data**
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: • **Test 5:** The evaluator will perform the following wildcard tests with each supported type of reference identifier: ○ **Test 5.3:** The evaluator will present a server certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com). The evaluator will configure the reference identifier with a single left-most label (e.g. foo.com) and verify that the connection fails. The evaluator will configure the reference identifier with two left-most labels (e.g. bar.foo.com) and verify that the connection fails.
Test Steps	SAN: cctest.com reference identifier: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com) • The evaluator then attempted to connect to the TLS server from the TOE. • The evaluator then verified that the connection failed. rhel.cctest.com reference identifier: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com) • The evaluator then attempted to connect to the TLS server from the TOE. • The evaluator then verified that the connection failed. CN: cctest.com reference identifier: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com) • The evaluator then attempted to connect to the TLS server from the TOE. • The evaluator then verified that the connection failed. rhel.cctest.com reference identifier: • The evaluator started a TLS server on the test VM with a certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com)

	- The evaluator then attempted to connect to the TLS server from the TOE. - The evaluator then verified that the connection failed.
Expected Test Results	The TOE should not connect to a TLS server that is using a server certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com) when the configured reference identifier is (e.g. foo.com) or (e.g. bar.foo.com).
Pass/Fail with Explanation	Yes. Pass. The TOE does not connect to a TLS server that is using a server certificate containing a wildcard in the left-most label immediately preceding the public suffix (e.g. *.com) when the configured reference identifier is (e.g. foo.com) or (e.g. bar.foo.com). This satisfies the testing requirement.

7.2.5.18 FCS_TLSC_EXT.1.2Test#6

Item	**Data**
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 6:** [conditional] If URI or Service name reference identifiers are supported, the evaluator will configure the DNS name and the service identifier. The evaluator will present a server certificate containing the correct DNS name and service identifier in the URIName or SRVName fields of the SAN and verify that the connection succeeds. The evaluator will repeat this test with the wrong service identifier (but correct DNS name) and verify that the connection fails.
Pass/Fail with Explanation	NA

7.2.5.19 FCS_TLSC_EXT.1.2Test#7

Item	**Data**
Test Assurance Activity	The evaluator will configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: - **Test 7:** [conditional] If pinned certificates are supported the evaluator will present a certificate that does not match the pinned certificate and verify that the connection fails.
Pass/Fail with Explanation	NA

7.2.5.20 FCS_TLSC_EXT.1.3Test#1

Item	**Data**
Test Assurance Activity	The evaluator will use TLS as a function to verify that the validation rules in FIA_X509_EXT.1.1 are adhered to and shall perform the following additional test: - **Test 1:** The evaluator will demonstrate that a peer using a certificate without a valid certification path results in an authenticate failure. Using the administrative guidance, the evaluator will then load the trusted CA certificate(s) needed to validate the peer's certificate, and demonstrate that the connection succeeds. The evaluator then shall delete one of the CA certificates, and show that the connection fails.
Pass/Fail Explanation	Satisfied by FIA_X509_EXT.1.1 Test 1.

7.2.5.21 FCS_TLSC_EXT.1.3Test#2

Item	Data
Test Assurance Activity	The evaluator will use TLS as a function to verify that the validation rules in FIA_X509_EXT.1.1 are adhered to and shall perform the following additional test: - **Test 2:** The evaluator will demonstrate that a peer using a certificate which has been revoked results in an authentication failure.
Pass/Fail with Explanation	Satisfied by FIA_X509_EXT.1.1 Test 1.

7.2.5.22 FCS_TLSC_EXT.1.3Test#3

Item	Data
Test Assurance Activity	The evaluator will use TLS as a function to verify that the validation rules in FIA_X509_EXT.1.1 are adhered to and shall perform the following additional test: - **Test 3:** The evaluator will demonstrate that a peer using a certificate which has passed its expiration date results in an authentication failure.
Pass/Fail with Explanation	Satisfied by FIA_X509_EXT.1.1 Test 1.

7.2.5.23 FCS_TLSC_EXT.1.3Test#4

Item	Data
Test Assurance Activity	The evaluator will use TLS as a function to verify that the validation rules in FIA_X509_EXT.1.1 are adhered to and shall perform the following additional test: - **Test 4:** the evaluator will demonstrate that a peer using a certificate which does not have a valid identifier shall result in an authentication failure.
Pass/Fail with Explanation	Satisfied by FIA_X509_EXT.1.1 Test 1.

7.2.5.24 FCS_TLSC_EXT.2.1Test#1

Item	Data
Test Assurance Activity	The evaluator will also perform the following test: - The evaluator will configure a server to perform ECDHE key exchange using each of the TOE's supported curves and shall verify that the TOE successfully connects to the server.
Test Steps	**secp256r1:** - Start a TLS server using a certificate that uses the curve - Attempt to connection to the TLS server from the TOE - Verify that the connection succeeds **secp384r1:** - Start a TLS server using a certificate that uses the curve - Attempt to connection to the TLS server from the TOE - Verify that the connection succeeds **secp521r1:**

	• Start a TLS server using a certificate that uses the curve • Attempt to connection to the TLS server from the TOE • Verify that the connection succeeds
Expected Test Results	The TOE should connect to a TLS server which each of the supported curves.
Pass/Fail with Explanation	Pass. The TOE successfully connects to a server using each of the claimed curves.

7.2.5.25 FCS_TLSC_EXT.3.1Test#1

Item	**Data**
Test Assurance Activity	The evaluator will also perform the following test: • The evaluator will configure the server to send a certificate in the TLS connection that is not supported according to the Client's HashAlgorithm enumeration within the signature_algorithms extension (for example, send a certificate with a SHA-1 signature). The evaluator will verify that the OS disconnects after receiving the server's Certificate handshake message.
Test Steps	• Configure End Entity certificate with SHA-1 signature (Endcert) • Start a TLS server using a certificate that uses SHA-1 signature • Attempt to connection to the TLS server from the TOE • Verify that the connection succeeds
Expected Test Results	The TOE should not connect to a TLS server with an unsupported signature algorithms extension
Pass/Fail with Explanation	Pass. The TOE does not connect to a TLS server with an unsupported signature algorithms extension.

7.2.5.26 FCS_TLSC_EXT.4.1Test#1

Item	**Data**
Test Assurance Activity	The evaluator will also perform the following test: • **Test 1:** The evaluator will establish a connection to a peer server that is not configured for mutual authentication (i.e. does not send Server's Certificate Request (type 13) message). The evaluator observes negotiation of a TLS channel and confirms that the TOE did not send Client's Certificate message (type 11) during handshake.
Pass/Fail with Explanation	NA.

7.2.5.27 FCS_TLSC_EXT.4.1Test#1

Item	**Data**
Test Assurance Activity	The evaluator will also perform the following test: • **Test 2:** The evaluator will establish a connection to a peer server with a shared trusted root that is configured for mutual authentication (i.e. it sends Server's Certificate Request (type 13) message). The evaluator observes negotiation of a

	TLS channel and confirms that the TOE responds with a non-empty Client's Certificate message (type 11) and Certificate Verify (type 15) messages.
Pass/Fail with Explanation	NA.

7.3 FDP – User Data Protection

7.3.1 ACF

7.3.1.1 FDP_ACF_EXT.1 Test #1

Item	Data
Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: • **Test 1:** The evaluator will authenticate to the system as the first user and create a file within that user's home directory. The evaluator will then log off the system and log in as the second user. The evaluator will then attempt to read the file created in the first user's home directory. The evaluator will ensure that the read attempt is denied.
Test Steps	• Log into the TOE as the first user • Create a file in the home directory • Log into the TOE as a second user • Attempt to read the file • Verify that the attempt fails
Expected Test Results	A user should not be able to read a file in another user’s home directory.
Pass/Fail with Explanation	Pass. The TOE prevented the user from reading another user's file. This meets testing requirements.

7.3.1.2 FDP_ACF_EXT.1 Test #2

Item	Data
Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: • **Test 2:** The evaluator will authenticate to the system as the first user and create a file within that user's home directory. The evaluator will then log off the system and log in as the second user. The evaluator will then attempt to modify the file created in the first user's home directory. The evaluator will ensure that the modification is denied.
Test Steps	• Log into the TOE as the first user • Create a file in the home directory • Log into the TOE as a second user • Attempt to modify the file and verify that the attempt fails.
Expected Test Results	A user should not be able to modify a file in another user’s home directory.
Pass/Fail with Explanation	Pass. The TOE prevented the user from modifying another user's file. This meets testing requirements.

7.3.1.3 FDP_ACF_EXT.1 Test #3

Item	Data

Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: • **Test 3:** The evaluator will authenticate to the system as the first user and create a file within that user's user directory. The evaluator will then log off the system and log in as the second user. The evaluator will then attempt to delete the file created in the first user's home directory. The evaluator will ensure that the deletion is denied.
Test Steps	• Log into the TOE as the first user • Create a file in the home directory • Log into the TOE as a second user • Attempt to delete the file • Verify that the attempt fails
Expected Test Results	A user should not be able to delete a file in another user’s home directory.
Pass/Fail with Explanation	Pass. The TOE prevented the user from deleting another user's file. This meets testing requirements.

7.3.1.4 FDP_ACF_EXT.1 Test #4

Item	Data
Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: • **Test 4:** The evaluator will authenticate to the system as the first user. The evaluator will attempt to create a file in the second user's home directory. The evaluator will ensure that the creation of the file is denied.
Test Steps	• Log into the TOE as the first user • Attempt to create a file in the second user’s home directory • Verify that it fails
Expected Test Results	A user should not be able to create a new file in another user’s home directory.
Pass/Fail with Explanation	Pass. The TOE prevented the user from creating a file in another user's home folder. This meets testing requirements.

7.3.1.5 FDP_ACF_EXT.1 Test #5

Item	Data
Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: • **Test 5:** The evaluator will authenticate to the system as the first user and attempt to modify the file created in the first user's home directory. The evaluator will ensure that the modification of the file is accepted.
Test Steps	• Log into the TOE as the first user • Create a file in the first user’s home directory • Attempt to modify that file as the first user • Verify that the modification attempt succeeds
Expected Test Results	A user should be able to modify a file that they created in their own home directory.
Pass/Fail with Explanation	Pass. The TOE allowed the user to read the user's file. This meets testing requirements.

7.3.1.6 FDP_ACF_EXT.1 Test #6

Item	Data
Test Assurance Activity	The evaluator will create two new standard user accounts on the system and conduct the following tests: Test 6: The evaluator will authenticate to the system as the first user and attempt to delete the file created in the first user's directory. The evaluator will ensure that the deletion of the file is accepted.
Test Steps	• Log into the TOE as the first user • Create a file in the home directory • Attempt to delete the file while logged into the first user account • Verify that it succeeds
Expected Test Results	A user should be able to delete a file that they created in their own home directory.
Pass/Fail with Explanation	Pass. The TOE allowed the user to delete the user's file. This meets testing requirements.

7.4 FIA – Identification and Authentication

7.4.1 AFL

7.4.1.1 FIA_AFL.1.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will set an administrator-configurable threshold for failed attempts, or note the ST-specified assignment. The evaluator will then (per selection) repeatedly attempt to authenticate with an incorrect password, PIN, or certificate until the number of attempts reaches the threshold. Note that the authentication attempts and lockouts must also be logged as specified in FAU_GEN.1.
Expected Test Results	The TOE should properly react (as selected in the ST) to a user that has reached the configured limit for consecutive failed authentication attempts and create an audit log.
Pass/Fail with Explanation	Pass. The TOE behaves as configured when a user makes multiple invalid login attempts.

7.4.1.2 FIA_AFL.1.2 Test#1

Item	Data
Test Assurance Activity	**Test 1:** The evaluator will attempt to authenticate repeatedly to the system with a known bad password. Once the defined number of failed authentication attempts has been reached the evaluator will ensure that the account that was being used for testing has had the actions detailed in the assignment list above applied to it. The evaluator will ensure that an event has been logged to the security event log detailing that the account has had these actions applied.
Test Steps	• Set user login unsuccessful authentication attempts. (deny = 3) • Start an SSH session with the TOE and attempt to login thrice (3 in this case) with wrong password and lock the user. • Verify the user is lockout for configured time with logs. • Attempt to open another connection and attempt to login with valid password before the lockout period expires. • Verify with logs the attempt failed due to lockout account.

Expected Test Results	The TOE should properly react (as selected in the ST) to a user that has reached the configured limit for consecutive failed authentication attempts and create an audit log.
Pass/Fail with Explanation	Pass. The TOE behaves as configured when a user makes multiple invalid login attempts.

7.4.1.3 FIA_AFL.1.2 Test#2

Item	**Data**
Test Assurance Activity	**Test 2:** The evaluator will attempt to authenticate repeatedly to the system with a known bad certificate. Once the defined number of failed authentication attempts has been reached the evaluator will ensure that the account that was being used for testing has had the actions detailed in the assignment list above applied to it. The evaluator will ensure that an event has been logged to the security event log detailing that the account has had these actions applied.
Pass/Fail with Explanation	Pass. *The TOE does not support certificate-based authentication.*

7.4.1.4 FIA_AFL.1.2 Test#3

Item	**Data**
Test Assurance Activity	**Test 3:** The evaluator will attempt to authenticate repeatedly to the system using both a bad password and a bad certificate. Once the defined number of failed authentication attempts has been reached the evaluator will ensure that the account that was being used for testing has had the actions detailed in the assignment list above applied to it. The evaluator will ensure that an event has been logged to the security event log detailing that the account has had these actions applied.
Pass/Fail with Explanation	Pass. The TOE does not support certificate-based authentication. For bad passwords please refer FIA_AFL.1.2 Test#1

7.4.2 UAU

7.4.2.1 FIA_UAU.5.1 Username and Password Test#1

Item	**Data**
Test Assurance Activity	If user name and password authentication is selected, the evaluator will configure the OS with a known user name and password and conduct the following tests: • **Test 1:** The evaluator will attempt to authenticate to the OS using the known user name and password. The evaluator will ensure that the authentication attempt is successful. **TD0649 Applied**
Test Steps	• Create a new username and password. • Attempt to login with correct username/password. • Verify the authentication attempt is successful.
Expected Test Results	The TOE should allow the evaluator to login when current user credentials are used.
Pass/Fail with Explanation	Pass. The TOE will allow user to login when current user credentials are used.

7.4.2.2 FIA_UAU.5.1 Username and Password Test#2

Item	Data
Test Assurance Activity	If user name and password authentication is selected, the evaluator will configure the OS with a known user name and password and conduct the following tests: • **Test 2:** The evaluator will attempt to authenticate to the OS using the known user name but an incorrect password. The evaluator will ensure that the authentication attempt is unsuccessful.
Test Steps	• Attempt to login with correct username and incorrect password . • Verify the authentication attempt is unsuccessful.
Expected Test Results	The TOE should not allow the evaluator to login when incorrect user credentials are used. (here incorrect password)
Pass/Fail with Explanation	Pass. The TOE did not allow user to login when incorrect user credentials were used.

7.4.2.3 FIA_UAU.5.2 Test#1

Item	Data
Test Assurance Activity	The evaluator will verify that configuration guidance for each authentication mechanism is addressed in the AGD guidance. • **Test 1:** For each authentication mechanism selected, the evaluator will enable that mechanism and verify that it can be used to authenticate the user at the specified authentication factor interfaces.
Test Steps	**Console:** Password based Authentication: • Start a console session and input correct username and password. • Verify if the connection succeeds with logs. **SSH:** Password based Authentication: • Start a SSH session from the VM to the TOE and input correct username and password. • Verify if the SSH session has started with logs. Public key-based Authentication: • Create a private/public key on the VM. • Copy the public key and store it in the authorized-keys file. • Start a SSH session from the VM to the TOE. • Verify the logs on the TOE.
Expected Test Results	The TOE should allow a user to authenticate with each of the authentication mechanisms selected.
Pass/Fail with Explanation	Pass. The TOE allows a user to authenticate with each of the authentication mechanisms selected.

7.4.2.4 FIA_UAU.5.2 Test#2

Item	Data
Test Assurance Activity	The evaluator will verify that configuration guidance for each authentication mechanism is addressed in the AGD guidance. • **Test 2:** For each authentication mechanism rule, the evaluator will ensure that the authentication mechanism(s) behave as documented in the TSS.

Pass/Fail with Explanation	Pass. *Test satisfied by FIA_UAU.5.2Test#1.*

7.4.3 X509

7.4.3.1 FIA_X509_EXT.1.1 Test#1

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 1:** The evaluator shall demonstrate that validating a certificate without a valid certification path results in the function failing, for each of the following reasons, in turn: o by establishing a certificate path in which one of the issuing certificates is not a CA certificate, o by omitting the basic Constraints field in one of the issuing certificates, o by setting the basic Constraints field in an issuing certificate to have CA=False, o by omitting the CA signing bit of the key usage field in an issuing certificate, and o by setting the path length field of a valid CA field to a value strictly less than the certificate path. The evaluator shall then establish a valid certificate path consisting of valid CA certificates and demonstrate that the function succeeds. The evaluator shall then remove trust in one of the CA certificates and show that the function fails.
Test Steps	Establishing a certificate path in which one of the issuing certificates is not a CA certificate. **--** **---** Omitting the basicConstraints field in one of the issuing certificates. • Generate a chain of 4 certificates with one of the certificates missing basicConstraints field. • Attempt a connection from the TOE with the OpenSSL server (10.1.4.61) and verify the connection fails. • Verify the connection with Packet capture. -- The basicConstraints field in an issuing certificate to have CA=False. • Generate a chain of 4 certificates with one of the certificates have CA=False. • Attempt a connection from the TOE with the OpenSSL server (10.4.61) and verify the connection fails. • Verify the connection with Packet capture. -- Omitting the CA signing bit of the key usage field in an issuing certificate. • Generate a chain of 4 certificates with certificates missing CA signing bit of the key usage field in an issuing certificate.

	• Attempt a connection from the TOE with the OpenSSL server (10.1.4.61) and verify the connection fails. • Verify the connection with Packet capture. -- Setting the path length field of a valid CA field to a value strictly less than the certificate path. • Generate a chain of 4 certificates with certificates have CA field to a value strictly less than the certificate path. • Attempt a connection from the TOE with the OpenSSL server (10.1.4.61) and verify the connection fails. • Verify the connection with Packet capture. -- Establish a valid certificate path consisting of valid CA certificates and remove trust in one of the CA certificates and verify connection succeeded and failed respectively. • The evaluator generated a chain of 4 certificates. (Root_CA->ICA1->ICA2->Endcert) • Delete ICA2 certificate from the TOE System keychain • Attempt a connection from the TOE with the OpenSSL server and verify the connection fails. • Load the missing ICA2 certificate on the TOE keychain. • Attempt a connection from the TOE with the OpenSSL server and verify the connection is successful.
Expected Test Results	The TOE should not connect to a TLS server when using a certificate path in which one of the issuing certificates is not a CA certificate. The TOE should not connect to a TLS server when omitting the basicConstraints field in one of the issuing certificates. The TOE should not connect to a TLS server when setting the basicConstraints field in an issuing certificate to have CA=False. The TOE should not connect to a TLS server when omitting the CA signing bit of the key usage field in an issuing certificate. The TOE should not connect to a TLS server when setting the path length field of a valid CA field to a value strictly less than the certificate path. The TOE should connect to a TLS server when a valid certificate path consisting of valid CA certificates is used. The TOE should not connect to a TLS server when trust in one of the CA certificates is removed
Pass/Fail with Explanation	Pass. The TOE properly denied or connected to a server based on the requirements above.

7.4.3.2 FIA_X509_EXT.1.1 Test#2

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extended Key Usage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 2:** The evaluator will demonstrate that validating an expired certificate results in the function failing.

Test Steps	- Start a TLS server with a certificate that has expired. - Attempt to connect to the TLS server from the TOE. - Verify that the connection fails.
Expected Test Results	The TOE should not connect to a TLS server if the server certificate has expired.
Pass/Fail with Explanation	Pass. The TOE does not connect to a server whose certificate has expired.

7.4.3.3 FIA_X509_EXT.1.1 Test#3

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. - **Test 3:** [Conditional, to be performed for use cases identified in exceptions that cannot be configured to allow revocation checking]: The evaluator will test that the OS can properly handle revoked certificates - conditional on whether CRL, OCSP, OCSP stapling, or OCSP multi-stapling is selected; if multiple methods are selected, then a test shall be performed for each method. The evaluator will test revocation of the node certificate and revocation of the intermediate CA certificate (i.e. the intermediate CA certificate should be revoked by the root CA). If OCSP stapling per RFC 6066 is the only supported revocation method, testing revocation of the intermediate CA certificate is omitted. The evaluator will ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails. If the exceptions are configurable, the evaluator shall attempt to configure the exceptions to allow revocation checking for each function indicated in FIA_X509_EXT.2. ***TD0715 Applied.***
Test Steps	Revoked End Certificate: - Start a TLS server with a revoked certificate. - Attempt to connect to the TLS server from the TOE. - Verify that the connection fails. Unrevoked End Certificate: - Start a TLS server with an unrevoked certificate. - Attempt to connect to the TLS server from the TOE. - Verify the connection is successful. Unrevoked ICA2 Certificate: - Start a TLS server with a revoked certificate. - Attempt to connect to the TLS server from the TOE. - Verify that the connection fails. Revoked ICA2 Certificate: - Start a TLS server with an unrevoked certificate. - Attempt to connect to the TLS server from the TOE.

	- Verify the connection is successful.
Expected Test Results	The TOE should not connect to the TLS server when either the node certificate or intermediate certificate is revoked.
Pass/Fail with Explanation	Pass. The TOE properly handles a revoked server or intermediate CA cert and connects when the certs are not revoked.

7.4.3.4 FIA_X509_EXT.1.1 Test#4

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 4:** If any OCSP option is selected, the evaluator shall configure the OCSP server or use a man-in-the-middle tool to present a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.
Test Steps	• Start a TLS server with a certificate that does not have the cRLsign key usage bit. • Attempt to connect to the TLS server from the TOE. • Verify that the connection fails.
Expected Test Results	The TOE should not connect to the TLS server if the CA that signs the CRL does not have the cRLsign key.
Pass/Fail with Explanation	Pass. The TOE does not connect allow a connection when a CA certificate signs a CRL when it does not have the CRLsign key usage bit set.

7.4.3.5 FIA_X509_EXT.1.1 Test#5

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 5:** The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)
Test Steps	• Start the acumen-tlsc tool so it will modify any byte in the first eight bytes of the certificate. • Attempt to connect to the tool from the TOE. • Verify that the connection fails
Expected Test Results	The TOE should not connect to a TLS server with a server certificate that had the first eight bytes modified.
Pass/Fail with Explanation	Pass. The TOE denies the connection to a remote TLS server when the server certificate has been modified

7.4.3.6 FIA_X509_EXT.1.1 Test#6

Item	Data
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 6:** The evaluator shall modify any byte in the last byte of the certificate and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
Test Steps	• Start the acumen-tlsc tool so it will modify any byte in the last eight bytes of the certificate. • Attempt to connect to the tool from the TOE. • Verify that the connection fails
Expected Test Results	The TOE should not connect to a TLS server with a server certificate that had the last eight bytes modified.
Pass/Fail with Explanation	Pass. The TOE denies the connection to a remote TLS server when the server certificate has been modified

7.4.3.7 FIA_X509_EXT.1.1 Test#7

Item	Data
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. • **Test 7:** The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The signature of the certificate will not validate.)
Test Steps	• Start the acumen-tlsc tool so it will modify any byte in the public key of the certificate. • Attempt to connect to the tool from the TOE. • Verify that the connection fails
Expected Test Results	The TOE should not connect to a TLS server with a server certificate that had the public key modified.
Pass/Fail with Explanation	Pass. The TOE denies the connection to a remote TLS server when the server certificate has been modified

7.4.3.8 FIA_X509_EXT.1.1 Test#8a

Item	Data
Test Assurance Activity	(Conditional on support for EC certificates as indicated in FCS_COP.1(3)). The evaluator shall establish a valid, trusted certificate chain consisting of an EC leaf certificate, an EC Intermediate CA certificate not designated as a trust anchor, and an EC certificate designated as a trusted anchor, where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.
Test Steps	• Configure the EC root CA certificate (here Root_CA) • Configure the EC intermediate CA certificate (here ICA1 and ICA2)

	- Configure the EC node certificate (here Endcert) - Start a TLS server that is using all EC certificates. - Attempt to connect to the TLS server from the TOE using an EC certificate. - Verify that the connection succeeds.
Expected Test Results	The TOE should connect to a TLS server when using an EC certificate.
Pass/Fail with Explanation	Pass. The TOE connects to a server when all certificates are using a named EC curve.

7.4.3.9 FIA_X509_EXT.1.1 Test#8b

Item	**Data**
Test Assurance Activity	The evaluator shall replace the intermediate certificate in the certificate chain for Test 8a with a modified certificate, where the modified intermediate CA has a public key information field where the EC parameters uses an explicit format version of the Elliptic Curve parameters in the public key information field of the intermediate CA certificate from Test 8a, and the modified Intermediate CA certificate is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.
Test Steps	- Modify the ICA2 certificate from FIA_X509_EXT.1.1 Test #8a using x509-mod tool. - Start a TLS server using the new explicit certificate. - Attempt to connect to the TLS server from the TOE. - Verify that the connection fails
Expected Test Results	The TOE should not connect to a TLS server that has one of the intermediate EC certificates modified to have the public key information field where the EC parameters uses an explicit format version of the Elliptic Curve parameters in the public key information field.
Pass/Fail with Explanation	Pass. The TOE rejects a connection to a remote server using an intermediate CA having explicit public key information.

7.4.3.10 FIA_X509_EXT.1.2 Test#1

Item	**Data**
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. - Test 1: The evaluator will construct a certificate path, such that the certificate of the CA issuing the OS's certificate does not contain the basicConstraints extension. The validation of the certificate path fails.
Pass/Fail with Explanation	Pass. *Covered in FIA_X509_EXT.1.1 Test #1, as the TOE will not validate a certificate with missing basicConstraints inside an issuer's certificate, but it will accept that same certificate when it has the full CA chain with the basicConstraints field defined in the issuing certificates. Incomplete certificates (without the basicConstraints extension) fail to validate and are rejected. This meets the testing requirements.*

7.4.3.11 FIA_X509_EXT.1.2 Test#2

Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The

	evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. - Test 2: The evaluator will construct a certificate path, such that the certificate of the CA issuing the OS's certificate has the CA flag in the basicConstraints extension not set. The validation of the certificate path fails
Pass/Fail with Explanation	Pass. *Covered in FIA_X509_EXT.1.1 Test #1 as the TOE will not validate a certificate with missing CA flag inside an issuer's certificate, but it will accept that same certificate when it has the full CA chain with the CA flag set inside the issuing certificates.*

7.4.3.12 FIA_X509_EXT.1.2 Test#3

Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services evaluation activities, including the functions in FIA_X509_EXT.2.1. The evaluator will create a chain of at least four certificates: the node certificate to be tested, two Intermediate CAs, and the self-signed Root CA. - Test 3: The evaluator will construct a certificate path, such that the certificate of the CA issuing the OS's certificate has the CA flag in the basicConstraints extension set to TRUE. The validation of the certificate path succeeds.
Pass/Fail with Explanation	Pass*. Covered in FCS_TLSC_EXT.1.1.The TOE will validate certificate with CA flag set to TRUE.*

7.4.3.13 FIA_X509_EXT.2.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will acquire or develop an application that uses the OS TLS mechanism with an X.509v3 certificate. The evaluator will then run the application and ensure that the provided certificate is used to authenticate the connection. The evaluator will repeat the activity for any other selections listed.
Pass/Fail with Explanation	Pass. *Covered in FCS_TLSC_EXT.1.1.*

7.5 FMT – Security Management

7.5.1 MOF

7.5.1.1 FMT_MOF_EXT.1 Test #1

Item	**Data**
Test Assurance Activity	- **Test 1:** For each function that is indicated as restricted to the administrator, the evaluation shall perform the function as an administrator, as specified in the Operational Guidance, and determine that it has the expected effect as outlined by the Operational Guidance and the SFR. The evaluator will then perform the function (or otherwise attempt to access the function) as a non-administrator and observe that they are unable to invoke that functionality.
Test Steps	**Note – here 'admin' is administrator and 'user' is a non – administrative user.* 1) Configure local audit storage capacity Successful - admin

	• Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 2) Configure minimum password length Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 3) Configure minimum number of special characters in password Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 4) Configure minimum number of numeric characters in password Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 5) Configure minimum number of uppercase characters in password

Successful - admin

- Before
- Attempt changes
- After
- Logs

Unsuccessful - user

- Before
- Attempt changes
- After
- Logs

6) Configure minimum number of lowercase characters in password

Successful - admin

- Before
- Attempt changes
- After
- Logs

Unsuccessful - user

- Before
- Attempt changes
- After
- Logs

7) Configure host-based firewall

Successful - admin

- Before
- Attempt changes
- After
- Logs

Unsuccessful - user

- Before
- Attempt changes
- After
- Logs

8) Configure name/address of audit/logging server to which to send audit/logging records

Successful - admin

- Before
- Attempt changes
- After
- Logs

Unsuccessful - user

- Before
- Attempt changes
- After

	• Logs 9) Configure audit rules Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 10) Configure name/address of network time server Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs 11) Enable/disable automatic software update Disable- Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before • Attempt changes • After • Logs Enable- Successful - admin • Before • Attempt changes • After • Logs Unsuccessful - user • Before

	• Attempt changes • After • Logs
Expected Test Results	The TOE should allow an admin to perform admin functions and restrict a non-admin from performing the admin functions.
Pass/Fail with Explanation	Pass. The TOE restricts configuration changes to privileged users. This satisfies the testing requirements.

7.5.2 SMF

7.5.2.1 FMT_SMF_EXT.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will test the OS's ability to provide the management functions by configuring the operating system and testing each option selected from above. The evaluator is expected to test these functions in all the ways in which the ST and guidance documentation state the configuration can be managed.
Pass/Fail with Explanation	Pass. *Test satisfied by FMT_MOF_EXT.1.*

7.6 FPT – Protection of the TSF

7.6.1 ACF

7.6.1.1 FPT_ACF_EXT.1.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 1:** The evaluator will attempt to modify all kernel drivers and modules.
Test Steps	/boot • Log into the TOE as an unprivileged user • Attempt to modify all kernel drivers and modules • Verify that the attempts fail /usr/lib/firmware: • Log into the TOE as an unprivileged user • Attempt to modify all kernel drivers and modules • Verify that the attempts fail **/**usr/lib/modules: • Log into the TOE as an unprivileged user • Attempt to modify all kernel drivers and modules • Verify that the attempts fail
Expected Test Results	The TOE should not allow a user to modify the kernel drivers
Pass/Fail with Explanation	Pass. The TOE prevented an unprivileged user from modifying kernel files. This satisfies the test requirement.

7.6.1.2 FPT_ACF_EXT.1.1 Test#2

Item	**Data**

Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 2:** The evaluator will attempt to modify all security audit logs generated by the logging subsystem.
Test Steps	Privileged User • Log into the TOE as a privileged user. • Show contents of /var/log directory • Log file before modification. (using file 'secure' for example) • Attempt to modify all security audit logs. • Verify that the modifications were applied. (using file 'secure' to compare) Unprivileged User • Log into the TOE as an unprivileged user. • Log file before modification. • Attempt to modify all security audit logs. • Verify that the attempt fails.
Expected Test Results	The TOE should not allow a non-privileged user to modify TOE logs.
Pass/Fail with Explanation	Pass.

7.6.1.3 FPT_ACF_EXT.1.1 Test#3

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 3:** The evaluator will attempt to modify all shared libraries that are used throughout the system.
Test Steps	/usr/lib: • Log into the TOE as an unprivileged user • Attempt to modify all shared libraries • Verify that the attempts fail /usr/lib64: • Log into the TOE as an unprivileged user • Attempt to modify all shared libraries • Verify that the attempts fail
Expected Test Results	The TOE should not allow an unprivileged user to modify the shared libraries.
Pass/Fail with Explanation	Pass. The TOE prevented an unprivileged user from modifying shared libraries. This satisfies the testing requirement.

7.6.1.4 FPT_ACF_EXT.1.1 Test#4

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 4:** The evaluator will attempt to modify all system executables.
Test Steps	/usr/bin • Log into the TOE as an unprivileged user

	- Attempt to modify all system executables - Verify that the attempts fail /usr/libexec - Log into the TOE as an unprivileged user - Attempt to modify all system executables - Verify that the attempts fail /usr/sbin - Log into the TOE as an unprivileged user - Attempt to modify all system executables - Verify that the attempts fail
Expected Test Results	The TOE should not allow a user to modify the system executables.
Pass/Fail with Explanation	Pass. The TOE prevented an unprivileged user from modifying all system executables. This satisfies the test requirement.

7.6.1.5 FPT_ACF_EXT.1.1 Test#5

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 5:** The evaluator will attempt to modify all system configuration files.
Test Steps	- Log into the TOE as an unprivileged user - Attempt to modify all system executables - Verify that the attempts fail
Expected Test Results	The TOE should not allow a user to modify the configuration files.
Pass/Fail with Explanation	Pass. The TOE prevented an unprivileged user from modifying all system configuration files. This passes the test requirement.

7.6.1.6 FPT_ACF_EXT.1.1 Test#6

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 6:** The evaluator will attempt to modify any additional components selected.
Pass/Fail with Explanation	NA. There are no additional components selected.

7.6.1.7 FPT_ACF_EXT.1.2 Test#1

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 1:** The evaluator will attempt to read security audit logs generated by the auditing subsystem
Test Steps	- Log into the TOE as an unprivileged user - Attempt to read security audit logs

	- Verify that the attempt fails
Expected Test Results	The TOE should not allow an unprivileged user to read the security audit logs.
Pass/Fail with Explanation	Pass. The TOE prevented the user from reading the security audit logs. This meets testing requirements.

7.6.1.8 FPT_ACF_EXT.1.2 Test#2

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 2:** The evaluator will attempt to read system-wide credential repositories
Test Steps	- Log into the TOE as an unprivileged user - Attempt to read system-wide credential repositories - Verify that the attempt fails
Expected Test Results	The TOE should not allow an unprivileged user to read the system-wide credential repositories
Pass/Fail with Explanation	Pass.

7.6.1.9 FPT_ACF_EXT.1.2 Test#3

Item	Data
Test Assurance Activity	The evaluator will create an unprivileged user account. Using this account, the evaluator will ensure that the following tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 3:** The evaluator will attempt to read any other object specified in the assignment
Pass/Fail with Explanation	NA. No other object is specified.

7.6.2 ASLR

7.6.2.1 FPT_ASLR_EXT.1 Test 1

Item	Data
Test Assurance Activity	The evaluator will select 3 executables included with the TSF. If the TSF includes a web browser it must be selected. If the TSF includes a mail client it must be selected. For each of these apps, the evaluator will launch the same executables on two separate instances of the OS on identical hardware and compare all memory mapping locations. The evaluator will ensure that no memory mappings are placed in the same location. If the rare chance occurs that two mappings are the same for a single executable and not the same for the other two, the evaluator will repeat the test with that executable to verify that in the second test the mappings are different. This test can also be completed on the same hardware and rebooting between application launches.
Test Steps	- Select 3 executables included with the TSF - Launch the executables and check the memory mapping for each one - Reboot the system - Launch the executables and check the memory mapping for each one again

	- Verify that the mappings are different
Expected Test Results	The TOE should not have executables running in the same memory locations.
Pass/Fail with Explanation	Pass

7.6.3 SBOP

7.6.3.1 FPT_SBOP_EXT.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will also preform the following test: • **Test 1:** The evaluator will inventory the kernel, libraries, and application binaries to determine those that do not implement stack-based buffer overflow protections. This list should match up with the list provided in the TSS.
Test Steps	/usr/lib64: • Execute a command to list all files in "exec_list.txt". (Note: exec_list.txt is used as parameter in cande_wrapper.py) • Run the script to check buffer overflow protection in /usr/lib64 directory: • Edit the exec_list file and add files that are mentioned in TSS section for lib64 • Run the script again and verify the files don't have stack-based protection enabled /usr/sbin: • Run the script to check buffer overflow protection in /usr/sbin directory: • Edit the exec_list.txt file and add /sbin files that are mentioned in the TSS that do not implement stack-based buffer protection. / usr/bin: • Run the script to check buffer overflow protection in /usr/bin directory: • Edit the exec_file and add list of files under /usr/lib directory and verify that the stack-based protection is not enabled on them. Run the script for /usr/libexec/os-prober/newns and verify the stack-based overflow protection is not implemented.
Expected Test Results	The TOE should be implementing stack-based buffer overflow protections.
Pass/Fail with Explanation	Pass

7.6.4 SRP

7.6.4.1 FPT_SRP_EXT.1.1 Test#1

Item	Data
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 1:** The evaluator will configure the OS to only allow code execution from the core OS directories. The evaluator will then attempt to execute code from a directory that is in the allowed list. The evaluator will ensure that the code they attempted to execute has been executed.
Test Steps	• Configure the TOE to only allow code execution from the core OS directories

	- Attempt to execute code from a directory that is in the allowed list - Ensure that the execution succeeds
Expected Test Results	The TOE should allow code to execute in directories where it is allowed.
Pass/Fail with Explanation	Pass. The TOE allows code execution in an allowed directory. This meets the testing requirement.

7.6.4.2 FPT_SRP_EXT.1.1 Test#2

Item	**Data**
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 2:** The evaluator will configure the OS to only allow code execution from the core OS directories. The evaluator will then attempt to execute code from a directory that is not in the allowed list. The evaluator will ensure that the code they attempted to execute has not been executed.
Test Steps	- Configure the TOE to only allow code execution from the core OS directories - Attempt to execute code from a directory that is not in the allowed list and ensure that the execution fails - Ensure that the execution succeeds
Expected Test Results	The TOE should not allow code execution in directories outside of the allowed list.
Pass/Fail with Explanation	Pass. The TOE does not allow code execution in directories outside of the allowed list. This meets the testing requirement.

7.6.4.3 FPT_SRP_EXT.1.1 Test#3

Item	**Data**
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 3:** The evaluator will configure the OS to only allow code that has been signed by the OS vendor to execute. The evaluator will then attempt to execute code signed by the OS vendor. The evaluator will ensure that the code they attempted to execute has been
Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.4.4 FPT_SRP_EXT.1.1 Test#4

Item	**Data**
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): - **Test 4:** The evaluator will configure the OS to only allow code that has been signed by the OS vendor to execute. The evaluator will then attempt to execute code signed by another digital authority. The evaluator will ensure that the code they attempted to execute has not been executed.
Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.4.5 FPT_SRP_EXT.1.1 Test#5

Item	Data
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 5:** The evaluator will configure the OS to allow execution of a specific application based on version. The evaluator will then attempt to execute the same version of the application. The evaluator will ensure that the code they attempted to execute has been executed.
Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.4.6 FPT_SRP_EXT.1.1 Test#6

Item	Data
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 6:** The evaluator will configure the OS to allow execution of a specific application based on version. The evaluator will then attempt to execute an older version of the application. The evaluator will ensure that the code they attempted to execute has not been executed.
Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.4.7 FPT_SRP_EXT.1.1 Test#7

Item	Data
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 7:** The evaluator will configure the OS to allow execution based on the hash of the application executable. The evaluator will then attempt to execute the application with the matching hash. The evaluator will ensure that the code they attempted to execute has been executed.
Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.4.8 FPT_SRP_EXT.1.1 Test#8

Item	Data
Test Assurance Activity	For each selection specified in the ST, the evaluator will ensure that the corresponding tests result in a negative outcome (i.e., the action results in the OS denying the evaluator permission to complete the action): • **Test 8:** The evaluator will configure the OS to allow execution based on the hash of the application executable. The evaluator will modify the application in such a way that the application hash is changed. The evaluator will then attempt to execute the application with the matching hash. The evaluator will ensure that the code they attempted to execute has not been executed.

Pass/Fail with Explanation	NA. The TOE supports only file path.

7.6.5 TST

7.6.5.1 FPT_TST_EXT.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will also preform the following test: - **Test 1:** The evaluator will perform actions to cause TSF software to load and observe that the integrity mechanism does not flag any executables as containing integrity errors and that the OS properly boots.
Test Steps	- Reboot the TOE - Verify that there are no errors in the log
Expected Test Results	There should be no errors in the boot log.
Pass/Fail with Explanation	Pass. The TOE boots properly and does not show any integrity errors.

7.6.5.2 FPT_TST_EXT.1 Test#2

Item	**Data**
Test Assurance Activity	The evaluator will also preform the following test: - **Test 2:** The evaluator will modify a TSF executable that is part of the bootchain verified by the TSF (i.e. Not the first-stage bootloader) and attempt to boot. The evaluator will ensure that an integrity violation is triggered and the OS does not boot (Care must be taken so that the integrity violation is determined to be the cause of the failure to load the module, and not the fact that in such a way to invalidate the structure of the module.).
Test Steps	- Modify a boot file - Reboot the TOE - Verify that the boot fails
Expected Test Results	The TOE should not boot if a boot file is modified.
Pass/Fail with Explanation	Pass. The TOE detects an integrity violation and does not boot. This satisfies the test requirement.

7.6.5.3 FPT_TST_EXT.1 Test#3

Item	**Data**
Test Assurance Activity	The evaluator will also preform the following test: - ***Test 3[conditional]***: If the ST author indicates that the integrity verification is performed using a public key *in an X509 certificate*, the evaluator will verify that the boot integrity mechanism includes a certificate validation according to FIA_X509_EXT.1 or all certificates in the chain from the certificate used for boot integrity to a certificate in the trust store that are not themselves in the trust store. This means that, for each X509 certificate in this chain that is not a trust store element, the evaluator must ensure that revocation information is available to the TOE during the bootstrap mechanism (before the TOE becomes fully operational). ***TD0463 Applied***

Pass/Fail with Explanation	NA. Not indicated in ST

7.6.6 TUD

7.6.6.1 FPT_TUD_EXT.1.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will check for an update using procedures described in the documentation and verify that the OS provides a list of available updates. Testing this capability may require installing and temporarily placing the system into a configuration in conflict with secure configuration guidance which specifies automatic update. The evaluator is also to ensure that the response to this query is authentic by using a digital signature scheme specified in FCS_COP.1(3). The digital signature verification may be performed as part of a network protocol as described in FTP_ITC_EXT.1.) If the signature verification is not performed as part of a trusted channel, the evaluator shall send a query response with a bad signature and verify that the signature verification fails. The evaluator shall then send a query response with a good signature and verify that the signature verification is successful. ***TD0463 Applied***
Test Steps	• Check for updates from a source that has a good signature • Verify that the check succeeds • Modify the signature • Check for update from the source that has its signature modified • Verify that the check fails
Expected Test Results	The TOE should properly check from updates from a source that has a good signature. The TOE should not be able to check for updates from a source that has a bad signature.
Pass/Fail with Explanation	Pass. The TOE checks for update from a good source while its not able to check it from a source that has a bad signature.

7.6.6.2 FPT_TUD_EXT.1.2 Test#1

Item	**Data**
Test Assurance Activity	For the following tests, the evaluator will initiate the download of an update and capture the update prior to installation. The download could originate from the vendor's website, an enterprise-hosted update repository, or another system (e.g. network peer). All supported origins for the update must be indicated in the TSS and evaluated. • **Test 1:** The evaluator will ensure that the update has a digital signature belonging to the vendor prior to its installation. The evaluator will modify the downloaded update in such a way that the digital signature is no longer valid. The evaluator will then attempt to install the modified update. The evaluator will ensure that the OS does not install the modified update. ***TD0463 Applied***
Test Steps	• Remove the OS update • Download an authentic OS update

	- Modify the update file so the signature is no longer valid - Attempt to install the update and verify that the update fails
Expected Test Results	The TOE should not install an OS update that has the signature modified.
Pass/Fail with Explanation	Pass. The TOE rejects an update if the RPM package has been modified.

7.6.6.3 FPT_TUD_EXT.1.2 Test#2

Item	**Data**
Test Assurance Activity	For the following tests, the evaluator will initiate the download of an update and capture the update prior to installation. The download could originate from the vendor's website, an enterprise-hosted update repository, or another system (e.g. network peer). All supported origins for the update must be indicated in the TSS and evaluated. - **Test 2:** The evaluator will ensure that the update has a digital signature belonging to the vendor. The evaluator will then attempt to install the update (or permit installation to continue). The evaluator will ensure that the OS successfully installs the update. ***TD0463 Applied***
Test Steps	- Verify checksum value for package on Red Hat Customer Portal - Download the same package and verify the checksum value matches. - Install the package and verify it successfully installed.
Expected Test Results	The TOE should be able to install an authentic OS update.
Pass/Fail with Explanation	Pass. The TOE allows an update to be installed with a valid digital signature.

7.6.6.4 FPT_TUD_EXT.2.1 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will check for updates to application software using procedures described in the documentation and verify that the OS provides a list of available updates. Testing this capability may require temporarily placing the system into a configuration in conflict with secure configuration guidance which specifies automatic update. The evaluator is also to ensure that the response to this query is authentic by using a digital signature scheme specified in FCS_COP.1(3). The digital signature verification may be performed as part of a network protocol as described in FTP_ITC_EXT.1.) If the signature verification is not performed as part of a trusted channel, the evaluator shall send a query response with a bad signature and verify that the signature verification fails. The evaluator shall then send a query response with a good signature and verify that the signature verification is successful. ***TD0463 Applied***
Pass/Fail with Explanation	Same as test 1.1 test1

7.6.6.5 FPT_TUD_EXT.2.2 Test#1

Item	**Data**
Test Assurance Activity	The evaluator will initiate an update to an application. This may vary depending on the application, but it could be through the application vendor's website, a

	commercial app store, or another system. All origins supported by the OS must be indicated in the TSS and evaluated. However, this only includes those mechanisms for which the OS is providing a trusted installation and update functionality. It does not include user or administrator-driven download and installation of arbitrary files. • **Test 1:** The evaluator will ensure that the update has a digital signature which chains to the OS vendor or another trusted root managed through the OS. The evaluator will modify the downloaded update in such a way that the digital signature is no longer valid. The evaluator will then attempt to install the modified update. The evaluator will ensure that the OS does not install the modified update. ***TD0463 Applied***
Test Steps	• Download the authentic application update file. • Modify the authentic application update file. • Attempt to Install the modified update file. • Verify that the installation fails.
Expected Test Results	The TOE should not install an OS update that has the signature modified.
Pass/Fail with Explanation	Pass. The TOE rejects an update if the RPM package has been modified.

7.6.6.6 FPT_TUD_EXT.2.2 Test#2

Item	Data
Test Assurance Activity	The evaluator will initiate an update to an application. This may vary depending on the application, but it could be through the application vendor's website, a commercial app store, or another system. All origins supported by the OS must be indicated in the TSS and evaluated. However, this only includes those mechanisms for which the OS is providing a trusted installation and update functionality. It does not include user or administrator-driven download and installation of arbitrary files. • **Test 2:** The evaluator will ensure that the update has a digital signature belonging to the OS vendor or another trusted root managed through the OS. The evaluator will then attempt to install the update. The evaluator will ensure that the OS successfully installs the update. ***TD0463 Applied***
Test Steps	• Check the checksum value of package on TOE's customer portal site. • Download the update package and verify that the checksum value matches with the package. • Initiate an update and verify the package is installed successfully.
Expected Test Results	The TOE should install an authentic application update.
Pass/Fail with Explanation	Pass. The TOE allows an update for a package to be installed with a valid signature.

7.7 FTA – TOE Access

7.7.1 TAB

7.7.1.1 FTA_TAB.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will configure the OS, per instructions in the OS manual, to display the advisory warning message "TEST TEST Warning Message TEST TEST". The evaluator

	will then log out and confirm that the advisory message is displayed before logging in can occur.
Test Steps	• Update the advisory warning message • Verify that the update succeeded SSH: Console:
Expected Test Results	The TOE should properly be able to update the warning message.
Pass/Fail with Explanation	Pass. The TOE allows an administrator to set a login banner and displays the banner prior to login.

7.8 FTP

7.8.1 ITC

7.8.1.1 FTP_ITC.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will configure the OS to communicate with another trusted IT product as identified in the second selection. The evaluator will monitor network traffic while the OS performs communication with each of the servers identified in the second selection. The evaluator will ensure that for each session a trusted channel was established in conformance with the protocols identified in the first selection. ***TD0649 Applied***
Pass/Fail with Explanation	Pass. Test satisfied by FCS_TLSC_EXT.1.1 Test 1, FCS_SSHC_EXT.1.1 and FCS_SSHS_EXT.1.1.

7.8.2 TRP

7.8.2.1 FTP_TRP.1 Test#1

Item	Data
Test Assurance Activity	The evaluator will also perform the following tests: • **Test 1:** The evaluator will ensure that communications using each remote administration method is tested during the course of the evaluation, setting up the connections as described in the operational guidance and ensuring that communication is successful.
Pass/Fail with Explanation	Pass. Test satisfied with FCS_SSHS_EXT.1.

7.8.2.2 FTP_TRP.1 Test#2

Item	Data
Test Assurance Activity	The evaluator will also perform the following tests: • **Test 2:** For each method of remote administration supported, the evaluator will follow the operational guidance to ensure that there is no available interface that can be used by a remote user to establish a remote administrative sessions without invoking the trusted path.
Test Steps	• The evaluator checked the open ports on the TOE and verified that only port 22 (SSH) was open. • The evaluator tried using telnet to access the TOE and verified that it failed.

Expected Test Results	The TOE should not have an available interface that can be invoked by a remote user without using a trusted path.
Pass/Fail with Explanation	Pass. The TOE only allows the trusted path to be use for remote administration. This passes the test requirement.

7.8.2.3 FTP_TRP.1 Test#3

Item	**Data**
Test Assurance Activity	The evaluator will also perform the following tests: • **Test 3:** The evaluator will ensure, for each method of remote administration, the channel data is not sent in plaintext.
Test Steps	• The evaluator connected to the TOE via SSH. • The evaluator then verified that the channel data is not sent in plaintext.
Expected Test Results	The channel data for remote administration should not be in plaintext.
Pass/Fail with Explanation	Pass. The channel data for remote administration is not sent in plaintext. This passes the test requirement.

7.8.2.4 FTP_TRP.1 Test#4

Item	**Data**
Test Assurance Activity	The evaluator will also perform the following tests: • **Test 4:** The evaluator will ensure, for each method of remote administration, modification of the channel data is detected by the OS.
Test Steps	• Establish a successful SSH session from the VM to the TOE to select a reference string constant. • Start Acumen MTIM tool and modify the reference bits to "ffffffffffff". • Initiate a SSH session from the VM while the tool is sniffing and verify it fails. • Verify that the TOE did not connect via packet capture. • Verify the same via logs.
Expected Test Results	The TOE should be able to detect if there has been a modification of channel data for each method of remote administration.
Pass/Fail with Explanation	Pass. The TOE detects if there has been a modification of channel data for each method of remote administration. This passes the test requirement.

8 Appendix A

8.1 Certificates Table

This section provides a table that lists all SFRs for which a CAVP certificate is claimed, the CAVP algorithm list name, and the CAVP certificate number.

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
FCS_CKM.1.1	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	OpenSSL (64 bit) (SHA_ASM)	RSA KeyGen (FIPS186-4)	A1823
	ECC schemes using “NIST curves” [selection: P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4	OpenSSL (64 bit) (SHA_ASM)	ECDSA KeyGen (FIPS186-4)	A1823
	FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526	N/A	-	No NIST CAVP, CCTL has performed all assurance/evaluation activities and documented in the ETR and AAR accordingly.
	FFC Schemes using safe primes that meet the following: ‘NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes	N/A	-	No NIST CAVP, CCTL has performed all assurance/evaluation activities and documented in the ETR and AAR accordingly.
FCS_CKM.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	OpenSSL (64 bit) (SHA_ASM)	KAS-ECC-SSC Sp800-56Ar3	A1823

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
	Logarithm Cryptography”			
	Finite field-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”	OpenSSL (64 bit) (FFC_DH)	KAS-FFC-SSC Sp800-56Ar3	A1834
	Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526	N/A	-	No NIST CAVP, CCTL has performed all assurance/evaluation activities and documented in the ETR and AAR accordingly.
FCS_COP.1.1(1)/ Encryption/Decryption (Refined)	AES-CBC (as defined in NIST SP 800-38A) cryptographic key sizes [128-bit, 256-bit]	OpenSSL (64 bit) (AESASM)	AES-CBC	A1794
	AES-CTR (as defined in NIST SP 800-38A) cryptographic key sizes [128-bit, 256-bit]	OpenSSL (64 bit) (AESASM)	AES-CTR	A1794
	AES-GCM (as defined in NIST SP 800-38D),] and cryptographic key sizes [128-bit, 256-bit]	OpenSSL (64 bit) (AESGCM_ASM_ASM)	AES-GCM	A2781
		OpenSSL (64 bit) (AESASM_ASM)	AES-GCM	A1816
FCS_COP.1.1(3) – Signing	RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 4,	OpenSSL (64 bit) (SHA_ASM)	RSA SigGen (FIPS186-4)	A1823
		OpenSSL (64 bit) (SHA_ASM)	RSA SigVer (FIPS186-4)	A1823
	ECDSA schemes using "NIST curves" P-256, P-384 and [P-521] that meet the following: FIPS	OpenSSL (64 bit) (SHA_ASM)	ECDSA SigGen (FIPS186-4)	A1823

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
	PUB 186-4, "Digital Signature Standard (DSS)", Section 5	OpenSSL (64 bit) (SHA_ASM)	ECDSA SigVer (FIPS186-4)	A1823
FCS_COP.1.1.(2) – Hashing	SHA-256, SHA-384, SHA-512] and message digest sizes 160 bits and [256 bits, 384 bits, 512 bits]	OpenSSL (64 bit) (SHA_ASM)	SHA2-256	A1823
		Red Hat Enterprise Linux 8 Kernel Crypto API Cryptographic Module	SHA2-256	A4710
		OpenSSL (64 bit) (SHA_ASM)	SHA2-384	A1823
		Red Hat Enterprise Linux 8 Kernel Crypto API Cryptographic Module	SHA2-384	A4710
		OpenSSL (64 bit) (SHA_ASM)	SHA2-512	A1823
		Red Hat Enterprise Linux 8 Kernel Crypto API Cryptographic Module	SHA2-512	A4710
FCS_COP.1.1(4) – KeyedHash	SHA-256, SHA-384, SHA-512] with key sizes [256 bits, 384 bits, 512 bits] and message digest sizes [256 bits, 384 bits, 512 bits]	OpenSSL (64 bit) (SHA_ASM)	HMAC-SHA2-256	A1823
		OpenSSL (64 bit) (SHA_ASM)	HMAC-SHA2-384	A1823
		OpenSSL (64 bit) (SHA_ASM)	HMAC-SHA2-512	A1823
		Red Hat Enterprise Linux 8 Kernel Crypto API Cryptographic Module	HMAC-SHA2-512	A4710
FCS_RBG_EXT.1	CTR_DRBG (AES), 256 bits of entropy	OpenSSL (64 bit) (AESASM)	Counter DRBG	A1794
	HMAC_DRBG (SHA-512)	Red Hat Enterprise Linux 8 Kernel Crypto API Cryptographic Module	Counter DRBG	A4710

9 Conclusion

The testing shows that all test cases required for conformance have passed testing.

End of Document