Assurance Activity Report for
Junos OS 20.3R3 for NFX350

Security Target Junos 20.3R3 for NFX350
Version 1.2

collaborative Protection Profile for Network Devices, Version 2.2e
collaborative Protection Profile Module for Stateful Traffic Filter Firewalls, Version 1.4e
PP-Module for Intrusion Prevention Systems (IPS), Version 1.0
PP-Module for Virtual Private Network (VPN) Gateways, Version 1.1

AAR Version 1.2
20 June 2022

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:
Juniper Networks, Inc.

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

The TOE Evaluation was Sponsored by:
Juniper Networks, Inc.

Evaluation Personnel:
Shaunak Shah
Aruna Shaju K
Yogesh Pawar
Pratheek Menon

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	07/09/2021	Initial Release.
0.2	04/01/2022	Updated as per AGD and ST updates.
1.0	05/06/2022	Updated for checkout submission.
1.1	06/13/2022	Updated based on ECR comments.
1.2	06/20/2022	Updated based on ECR comments.

Contents

`

1 TOE Overview

The Target of Evaluation (TOE) is Juniper Networks, Inc. Junos OS 20.3R3 for NFX350 Network Services Platform. The NFX350 is a network device that integrates routing, switching, and security functions on a single platform.

The NFX350 supports the definition of, and enforces, information flow policies among network nodes, also providing for stateful inspection of every packet that traverses the network and central management to manage the network security policy. All information flow from one network node to another passes through an instance of the TOE. Information flow is controlled on the basis of network node addresses, protocol, type of access requested, and services requested. In support of the information flow security functions, the TOE ensures that security-relevant activity is audited, that their own functions are protected from potential attacks, and provides the security tools to manage all of the security functions. The TOE provides multi-site virtual private network (VPN) gateway functionality, and also implements Intrusion Prevention System functionality, capable of monitoring information flows to detect potential attacks based on pre-defined attack signature and anomaly characteristics in the traffic.

The deployment of the Junos OS 20.3R3 for NFX350 TOE includes a hypervisor, which runs a virtual machine (VM) on an NFX350 series hardware model:

- NFX350-S1
- NFX350-S2
- NFX350-S3

1.1 TOE Description

This section provides an overview of the TOE architecture, including physical boundaries, security functions, and relevant TOE documentation and references.

The TOE includes a Linux Operating System (OS), Junos Control Plane (JCP), a Juniper Device Manager (JDM) and an Open vSwitch (OVS) bridge. NFX350 supports the installation of 3rd party VMs and containers, but installation of 3rd party VMs and containers is not allowed in the evaluated configuration.
Figure 1 below shows the general architecture for the NFX350.

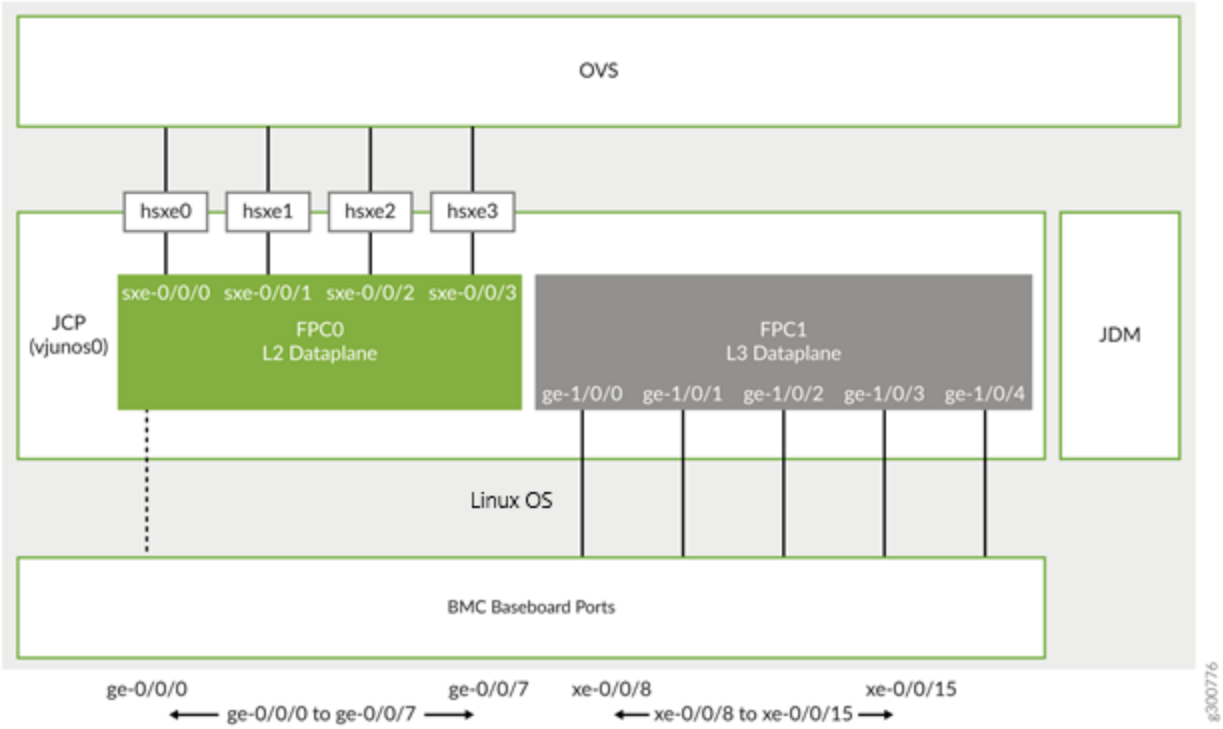


Figure 1 NFX350 Architecture

1.1.1 Linux OS

NFX350 is running on Wind River Linux 8 as its host OS. The host OS functions as a hypervisor and runs natively on an Intel Xeon D processor.

1.1.2 Junos Control Plane

Junos Control Plane (JCP) is the Junos VM running on the host OS. JCP is used to configure the network ports of the NFX350 device, and JCP runs by default as vjunos0 on NFX350. The JCP functions as the single point of management for all the components. The JCP supports:

- Layer 2 to Layer 3 routing services
- Layer 3 to Layer 4 security services
- Layer 4 to Layer 7 advanced security services

In addition, the JCP enables virtualized network functions (VNF) lifecycle management. VNF is a virtualized implementation of a network device and its functions. In the NFX350 NextGen architecture, Linux functions as the hypervisor, and it creates and runs the VNFs. The VNFs include functions such as firewalls, routers, and WAN accelerators.
The JCP VM is the single administration point for the NFX350 platform. It is the front-end for all functionality provided by the NFX350 software. Logging in via console of SSH take the user to a CLI prompt on the JCP VM. This CLI is the single point of configuration for all NFX350 services.

1.1.2.1 L2 Data Plane

L2 data plane manages the Layer 2 traffic. The L2 data plane forwards the LAN traffic to the OVS bridge. The L2 data plane is mapped to the virtual FPC0 on the JCP.

1.1.2.2 L3 Data Plane

L3 data plane provides data path functions for the Layer 3 to Layer 7 services. The L3 data plane is mapped to the virtual FPC1 on the JCP.

1.1.3 Juniper Device Manager (JDM)

JDM is an application container that manages VNFs and provides infrastructure services. The JDM functions in the background. JDM is a low-footprint Linux container that provides these functions:

- Virtual Machine (VM) lifecycle management
- Device management and isolation of host OS from user installations
- NIC, single-root I/O virtualization (SR-IOV), and virtual input/output (VirtIO) interface provisioning
- Inventory and resource management
- Internal network and image management
- Service chaining—provides building blocks such as virtual interfaces and bridges for users to implement service chaining polices
- Virtual console access to VNFs including vSRX and vjunos

1.1.4 Open vSwitch (OVS) bridge

The OVS bridge is a VLAN-aware system bridge that acts as the network functions virtualization backplane to which the VNFs, FPC1, and FPC0 connect.

1.1.5 NFX350 Hardware

The hardware model specifications are described in the table below.

Table 1 – TOE Hardware Specifications

Specification	NFX350-S1	NFX350-S2	NFX350-S3
Dimensions (H x W x D)	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm)	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm))	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm))
Rack units (U)	1 U	1 U	1 U
Weight	18.5 lb (8.4 kg)	18.6 lb (8.45 kg)	18.6 lb (8.45 kg)
Airflow	Front-to-back (AFO) forced cooling	Front-to-back (AFO) forced cooling	Front-to-back (AFO) forced cooling
Acoustics	61 dBA	61 dBA	61 dBA
Power	650W hot-swappable AC-DC/DC-DC	650W hot-swappable AC-DC/DC-DC	650W hot-swappable AC-DC/DC-DC
CPU	Intel Xeon D-2146NT 8 Core	Intel Xeon D-2166NT 12 Core	Intel Xeon D-2187NT 16 Core
Micro-Architecture	Skylake	Skylake	Skylake
Memory	32 GB DDR4	64 GB DDR4	128 GB DDR4
Storage	100 GB SSD[1]	100 GB SSD[2]	100 GB SD[2]
Software	Wind River Linux 8	Wind River Linux 8	Wind River Linux 8

[1] Raw capacity; actual capacity will be lower due to overprovisioning.

Specification	NFX350-S1	NFX350-S2	NFX350-S3
Network interfaces	• 8 x 10/100/ 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE SFP+ LAN or WAN ports • 1 x 10/100/ 1000BASE-T RJ-45 management port	• 8 x 10/100/ 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE SFP+ LAN or WAN ports • 1 x 10/100/ 1000BASE-T RJ-45 management port	• 8 x 10/100/ • 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE • SFP+ LAN or WAN ports • 1 x 10/100/ • 1000BASE-T RJ-45 management port
Managed Secure Router[2]	12 Gbps	20 Gbps	30 Gbps
Managed Security	12 Gbps	20 Gbps	30 Gbps
IPsec	2.5 Gbps	5 Gbps	7.5 Gbps
Out-of-band interfaces	RJ-45 console port Mini USB console port USB 2.0 port	RJ-45 console port Mini USB console port USB 2.0 port	RJ-45 console port Mini USB console port USB 2.0 port
Maximum number of VNFs	8	10	12

The JCP's FreeBSD kernel uses the kernel-based virtual machine (KVM) as a virtualization infrastructure. KVM is part of the standard NFX350 distribution and can be used to create multiple virtual machines (VMs) and to install security and networking appliances. The TOE uses Open vSwitch as a backplane between these VMs. However, in the TOE evaluated configuration, only a single VM is running and no security or networking appliances may be installed. Therefore, in the evaluated configuration the KVM functions simply as a pass-through layer.

The interfaces on the NFX350 devices include physical interfaces and virtual interfaces.

The physical interfaces represent the physical ports on the NFX350 chassis. The physical interfaces include network and management ports:

- Network ports NFX350 chassis —
  - 8 x 10/100/1000BASE-T RJ-45 LAN or WAN ports
  - 8 x 1GbE/10GbE SFP+ LAN or WAN ports
  - 1 x 10/100/1000BASE-T RJ-45 management port
- Management port – NFX350 device has a dedicated management ports which functions as the out-of-band management interface –
  - RJ-45 console port
  - Mini USB console port
  - 2 x USB 3.0 port

Each physical NIC port has sixteen virtual functions (VFs) enabled by default.

The virtual interfaces on the NFX350 device include the following:

[2] Maximum throughput mode

- Virtual layer 3 interfaces – used to configure layer 3 features such as routing protocols and QoS
- Virtual SXE interfaces – four SXE ports (static interfaces) connect the layer 2 data plane

Physical ports on the front panel of the NFX350 device can be mapped to layer 2 or layer 3 interfaces or Virtualized Network Functions (VNF)s. There is a dedicated IPsec VPN interface (FPC1).

The JCP and PFE perform their primary tasks independently, while constantly communicating with each other. This arrangement provides streamlined forwarding and routing control and the capability to run Internet-scale networks at high speeds.

NFX350 supports numerous routing standards for flexibility and scalability as well as IETF IPSec protocols. These functions can all be managed through the Junos OS software, either from a connected console on the management interface or via a network connection. Network management can be secured using IPsec and SSH protocols. All management, whether from a user connecting to a console or from the network, requires successful authentication. **In the evaluated deployment, network management (using the CLI) is secured using the SSH protocol, which can be tunnelled over IPsec.**

The TOE supports intrusion detection and prevention functionality, which allows it to detect and react to potential attacks in real time. The detection component of the IPS can be based on attack signatures which specify the characteristics of the potentially malicious traffic based on a variety of packet headers payload data attributes. Anomaly detection based on deviation of the monitored traffic from expected values is also supported.

In the evaluated configuration the TOE is managed and configured via Command Line Interface either via a directly connected console or using SSH connections (optionally tunnelled over IPsec).

1.1.6 Physical Boundaries

The physical boundaries of the TOE is the NFX350 series hardware running Junos OS 20.3R3.

The Junos OS 20.3R3 for NFX350 software includes the KVM Hypervisor as well as the JCP and JDM. Hence the TOE is contained within the physical boundary of each server specified above. The TOE is delivered as a single device with the Junos OS software installed. The TOE model number can be verified through the shipping label and device front panel. The software version can be verified by the show version command once the device is configured.

The Management platform and external syslog server are outside the boundary of the TOE.

1.1.7 Logical Boundary

The logical boundary of the TOE includes the following security functionality:

Table 2 – TOE Logical Boundary Security Functionality

Protected Communications	The TOE provides an SSH server to support protected communications for administrators to establish secure sessions and to connect to external syslog servers. The TOE also supports IPsec connections to provide multi-site virtual private network (VPN) gateway functionality and also as a tunnel for remote administrate SSH connections. The TOE requires that applications exchanging information with it are successfully authenticated prior to any exchange (i.e. applications connecting over SSH and IPsec). Telnet, File Transfer Protocol (FTP), and Secure Socket Layer (SSL) are out of scope. The TOE includes cryptographic modules that provide the underlying cryptographic services, including key management and protection of stored keys, algorithms, random bit generation and crypto-

	administration. The cryptographic modules provide confidentiality and integrity services for authentication and for protecting communications with adjacent systems.
Administrator Authentication	Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.
Correct Operation	The TOE provides for both cryptographic and non-cryptographic self-tests and is capable of automated recovery from failure states.
Trusted Update	The administrator can initiate update of the TOE software. The integrity of any software updates is verified prior to installation of the updated software.
Audit	TOE auditable events are stored in the syslog **files in the VM filesystem** and can be sent to an external log server (via Netconf over SSH). Auditable events include start-up and shutdown of the audit functions, authentication events, service requests, IPS events, as well as the events listed in Table 12 and Table 13 of the ST. Audit records include the date and time, event category, event type, username, and the outcome of the event (success or failure). Local (VM) syslog storage limits are configurable and are monitored. In the event of storage limits being reached the oldest logs will be overwritten.
Management	The TOE provides a Security Administrator role that is responsible for: • the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product • the regular review of all audit data; • initiation of trusted update function; • administration of VPN, IPS and Firewall functionality; • all administrative tasks (e.g., creating the security policy). The devices are managed through a Command Line Interface (CLI). The CLI is accessible through local (serial) console connection or remote administrative (SSH) session. The Security Administrator role includes the capability to manage all NFX350 services. Access to manage the device's FreeBSD host can only be gained through the JCP.
Packet Filtering/Stateful Traffic Filtering	The TOE provides stateful network traffic filtering based on examination of network packets and the application of information flow rules.
Intrusion Prevention	The TOE can be configured to analyze IP-based network traffic forwarded to the TOE's interfaces and detect violations of administratively-defined IPS policies. The TOE is capable of initiating a proactive response to terminate/interrupt an active potential threat, and to initiate a response in real time that would cause interruption of the suspicious traffic flow.
User Data Protection/Information Flow Control	The TOE is designed to forward network packets (i.e., information flows) from source network entities to destination network entities based on available routing information **using Virtual Routers**. This information is either provided directly by TOE users or indirectly from other network

	entities (outside the TOE) configured by the TOE users. The TOE has the capability to regulate the information flow across its interfaces; traffic filters can be set in accordance with the presumed identity of the source, the identity of the destination, the transport layer protocol, the source service identifier, and the destination service identifier (TCP or UDP port number).

1.1.8 Non-TOE hardware/software/firmware

The TOE relies on the provision of the following items in the network environment:

- Syslog server supporting SSHv2 connections to send audit logs;
- SSHv2 client for remote administration;
- Serial connection client for local administration.
- IPsec peer

1.1.9 Security Functions Provided by the TOE

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v2.2e or NDcPP.

1.1.9.1 Security Audit

TOE stored auditable event files locally but can be sent to an external log server (over SSH vis Netconf). The TOE provides appropriate start-up and shut-down functions, authentication events, service requests, IPS events and as well events listed in Table 12 and Table 13 of the Security Target.

Audit records include the date and time, event category, event type, username, and the outcome of the event (success or failure). Local (VM) syslog storage limits are configurable and are monitored. In the event of storage limits being reached the oldest logs will be overwritten.

1.1.9.2 Cryptographic Support

The TOE includes cryptographic modules that provide the underlying cryptographic services, including key management and protection of stored keys, algorithms, random bit generation and crypto-administration. The cryptographic modules provide confidentiality and integrity services for authentication and for protecting communications with adjacent systems.

1.1.9.3 Identification and Authentication

Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.

1.1.9.4 Security Management

Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.

The TOE provides a Security Administrator role that is responsible for:

- the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product

- the regular review of all audit data;
- initiation of trusted update function;
- administration of VPN, IPS and Firewall functionality;
- all administrative tasks (e.g., creating the security policy).

The devices are managed through a Command Line Interface (CLI). The CLI is accessible through local (serial) console connection or remote administrative (SSH) session. The Security Administrator role includes the capability to manage all NFX350 services.

1.1.9.5 Protection of the TSF

The TOE prevents reading of all pre-shared keys, symmetric keys and private keys. Its administrative passwords are store in non-plaintext form to prevent the reading of them. The TOE also runs a suite of self-tests during initial start-up which include the following:

- Power on test
- File Integrity test
- Crypto integrity test
- Authentication test
- Algorithm know answer tests

These self-tests are executed through a TSF provided cryptographic service specified in FCS_COP.1/SigGen.

1.1.9.6 TOE Access

The TOE provides capabilities for controlling session termination, locking and banner display.

1.1.9.7 Trusted Path/Channels

The TOE provides an SSH server to support protected communications for administrators to establish secure sessions and to connect to external syslog servers.

1.1.9.8 Firewall

The TOE uses Stateful Traffic Filtering on network packets that are processed which can permit or drop the capability to the log operations. These Stateful Traffic Filtering rules (found under FFW_RUL.EXT.1) are assigned to a distinct network interface.

1.1.9.9 VPN

The TOE also supports IPsec connections to provide multi-site virtual private network (VPN) gateway functionality and also as a tunnel for remote administrative SSH connections. The TOE requires that applications exchanging information with it are successfully authenticated prior to any exchange (i.e. applications connecting over SSH and IPsec).

1.1.9.10 IPS

The TOE includes functionality for policy rules such as white listing via packet monitoring and traffic detection. The TOE performs IP-based analysis of the network traffic in order to detect unusual activity not defined by the administrator. A list of known-good and known-bad IP addresses by source and destination are kept. Additionally a list of known-good and known-bad rules following IPS policy elements is also maintained.

2 Assurance Activities Identification

The Assurance Activities contained within this document include all those defined within the NDcPP 2.2e based upon the core SFRs and those implemented based on selections within the PPs/EPs.

3 Test Equivalency Justification

This testing equivalency analysis is for the Juniper Networks, Inc. Junos OS 20.3R3 for NFX350 Network Services Platform TOE. This analysis provides an explanation of the differences between each of the hardware models included within the TOE boundary and provides an analysis of the impact each of the differences have on the TSF functionality.

3.1 Architectural Description

The TOE is Juniper Networks, Inc. Junos OS 20.3R3 for NFX350 Network Services Platform. The NFX350 integrates routing, switching, and security functions on a single platform.

The NFX350 supports the definition of, and enforces, information flow policies among network nodes, also providing for stateful inspection of every packet that traverses the network and central management to manage the network security policy. All information flow from one network node to another passes through an instance of the TOE. Information flow is controlled on the basis of network node addresses, protocol, type of access requested, and services requested. In support of the information flow security functions, the TOE ensures that security-relevant activity is audited, that their own functions are protected from potential attacks, and provides the security tools to manage all of the security functions. The TOE provides multi-site virtual private network (VPN) gateway functionality, and also implements Intrusion Prevention System functionality, capable of monitoring information flows to detect potential attacks based on pre-defined attack signature and anomaly characteristics in the traffic.

The hardware model specifications are described in the table below.

Specification	NFX350-S1	NFX350-S2	NFX350-S3
Dimensions (H x W x D)	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm)	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm))	1.72 x 17.32 x 20.86 in (4.37 x 44.0 x 53.0 cm))
Rack units (U)	1 U	1 U	1 U
Weight	18.5 lb (8.4 kg)	18.6 lb (8.45 kg)	18.6 lb (8.45 kg)
Airflow	Front-to-back (AFO) forced cooling	Front-to-back (AFO) forced cooling	Front-to-back (AFO) forced cooling
Acoustics	61 dBA	61 dBA	61 dBA
Power	650W hot-swappable AC-DC/DC-DC	650W hot-swappable AC-DC/DC-DC	650W hot-swappable AC-DC/DC-DC
CPU	Intel Xeon D-2146NT 8 Core	Intel Xeon D-2166NT 12 Core	Intel Xeon D-2187NT 16 Core
Micro-Architecture	Skylake	Skylake	Skylake
Memory	32 GB DDR4	64 GB DDR4	128 GB DDR4
Storage	100 GB SSD[3]	100 GB SSD	100 GB SSD
Software	Wind River Linux 8	Wind River Linux 8	Wind River Linux 8
Network interfaces	• 8 x 10/100/ 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE SFP+ LAN or WAN ports	• 8 x 10/100/ 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE SFP+ LAN or WAN ports	• 8 x 10/100/ • 1000BASE-T RJ-45 LAN or WAN ports • 8 x 1GbE/10GbE • SFP+ LAN or WAN ports

[3] Raw capacity; actual capacity will be lower due to overprovisioning.

Specification	NFX350-S1	NFX350-S2	NFX350-S3
	• 1 x 10/100/ 1000BASE-T RJ-45 management port	• 1 x 10/100/ 1000BASE-T RJ-45 management port	• 1 x 10/100/ • 1000BASE-T RJ-45 management port
Managed Secure Router[4]	12 Gbps	20 Gbps	30 Gbps
Managed Security	12 Gbps	20 Gbps	30 Gbps
IPsec	2.5 Gbps	5 Gbps	7.5 Gbps
Out-of-band interfaces	RJ-45 console port Mini USB console port USB 2.0 port	RJ-45 console port Mini USB console port USB 2.0 port	RJ-45 console port Mini USB console port USB 2.0 port
Maximum number of VNFs	8	10	12

Table 3 TOE Hardware Specifications

3.1.1 Software Comparison

The Junos OS 20.3R3 firmware runs on all three models of NFX350. There is no difference in the firmware image.

3.1.2 Processor Comparison

All CPUs are manufactured by Intel and are part of the XEON D family of CPUs. All processors include the Skylake microarchitecture with Intel® AVX2 instruction set extensions. All CPUs support AES-NI, Secure Key, and the Execute Disable Bit. All models are using the same CPU family including microarchitecture; therefore, the hardware models are considered equivalent.

3.1.3 Platform Differences

Table 1 above provides a description of the physical differences between hardware models. None of the listed hardware differences have any impact of the security functionality provided by the TSF. All operate identically.

3.2 **Equivalency Analysis**

The following equivalency analysis provides a per category analysis of key areas of differentiation for each hardware model to determine the minimum subset to be used in testing. The areas examined will use the areas and analysis description provided in the supporting documentation for the NDcPP.

3.2.1 Platform/Hardware Differences

The TOE boundary is inclusive of all hardware required by the TOE. The hardware platforms do not provide any of the TSF functionality. For the hardware appliances, the hardware within the TOE only differs by configuration and performance. There are no hardware specific dependencies of the product.
Result:

- There are no hardware dependencies.
- All devices are equivalent.

3.2.2 Processor Differences

Across appliance platforms, the same processor family including microarchitecture is used.
The XEON D series include the Skylake microarchitecture with Intel® AVX2 instruction set extensions. CPU supports AES-NI, Secure Key, and the Execute Disable Bit.

[4] Maximum throughput mode.

3.2.3 Software/OS Dependencies

The source code for the TSF is identical across all models.

Result:

- The same software is used across all models.
- All NFX350 devices are equivalent.

3.2.4 Differences in Libraries Used to Provide TOE Functionality

All software binaries compiled in the TOE software are identical including the version of the library regardless of the platform for which the software is compiled. There are no differences between the included libraries.
Result:

- There are no differences in the included libraries.
- All NFX350 devices are equivalent.

3.2.5 TOE Management Interface Differences

TOE management interface is provided by secure SSH v2 session or via a local console connection. The management interface is the same across all platforms, and the protocol used for secure remote management is SSH v2.
Result:

- There are no differences in the user interface amongst platforms.
- All NFX350 devices are equivalent.

3.2.6 TOE Functional Differences

There are no functional differences.
The TOE implements the following security functionality throughout all of the models.

Protected Communications	The TOE provides an SSH server to support protected communications for administrators to establish secure sessions and to connect to external syslog servers. The TOE also supports IPsec connections to provide multi-site virtual private network (VPN) gateway functionality and also as a tunnel for remote administrate SSH connections. The TOE requires that applications exchanging information with it are successfully authenticated prior to any exchange (i.e. applications connecting over SSH and IPsec). Telnet, File Transfer Protocol (FTP), and Secure Socket Layer (SSL) are out of scope. The TOE includes cryptographic modules that provide the underlying cryptographic services, including key management and protection of stored keys, algorithms, random bit generation and crypto-administration. The cryptographic modules provide confidentiality and integrity services for authentication and for protecting communications with adjacent systems.
Administrator Authentication	Administrative users must provide unique identification and authentication data before any administrative access to the system is granted. Authentication data entered and stored on the TOE is protected. The TOE can be configured to terminate interactive user sessions and to present an access banner with warning messages prior to authentication.

Correct Operation	The TOE provides for both cryptographic and non-cryptographic self-tests and is capable of automated recovery from failure states.
Trusted Update	The administrator can initiate update of the TOE software. The integrity of any software updates is verified prior to installation of the updated software.
Audit	TOE auditable events are stored in the syslog **files in the VM filesystem** and can be sent to an external log server (via Netconf over SSH). Auditable events include start-up and shutdown of the audit functions, authentication events, service requests, IPS events, as well as the events listed in Table 12 and Table 13 of the ST. Audit records include the date and time, event category, event type, username, and the outcome of the event (success or failure). Local (VM) syslog storage limits are configurable and are monitored. In the event of storage limits being reached the oldest logs will be overwritten.
Management	The TOE provides a Security Administrator role that is responsible for: • the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product • the regular review of all audit data; • initiation of trusted update function; • administration of VPN, IPS and Firewall functionality; • all administrative tasks (e.g., creating the security policy). The devices are managed through a Command Line Interface (CLI). The CLI is accessible through local (serial) console connection or remote administrative (SSH) session. The Security Administrator role includes the capability to manage all NFX350 services. Access to manage the device's FreeBSD host can only be gained through the JCP.
Packet Filtering/Stateful Traffic Filtering	The TOE provides stateful network traffic filtering based on examination of network packets and the application of information flow rules.
Intrusion Prevention	The TOE can be configured to analyze IP-based network traffic forwarded to the TOE's interfaces and detect violations of administratively-defined IPS policies. The TOE is capable of initiating a proactive response to terminate/interrupt an active potential threat, and to initiate a response in real time that would cause interruption of the suspicious traffic flow.
User Data Protection/Information Flow Control	The TOE is designed to forward network packets (i.e., information flows) from source network entities to destination network entities based on available routing information **using Virtual Routers**. This information is either provided directly by TOE users or indirectly from other network entities (outside the TOE) configured by the TOE users. The TOE has the capability to regulate the information flow across its interfaces; traffic filters can be set in accordance with the presumed identity of the source, the identity of the destination, the transport layer protocol, the source service identifier, and the destination service identifier (TCP or UDP port number).

Result:

- There are no security functional differences between platforms in a series.
- All NFX350 devices are equivalent.

3.3 Recommendations/Conclusion

The similarities and differences between the NFX350 devices have been reviewed above. This analysis has shown:

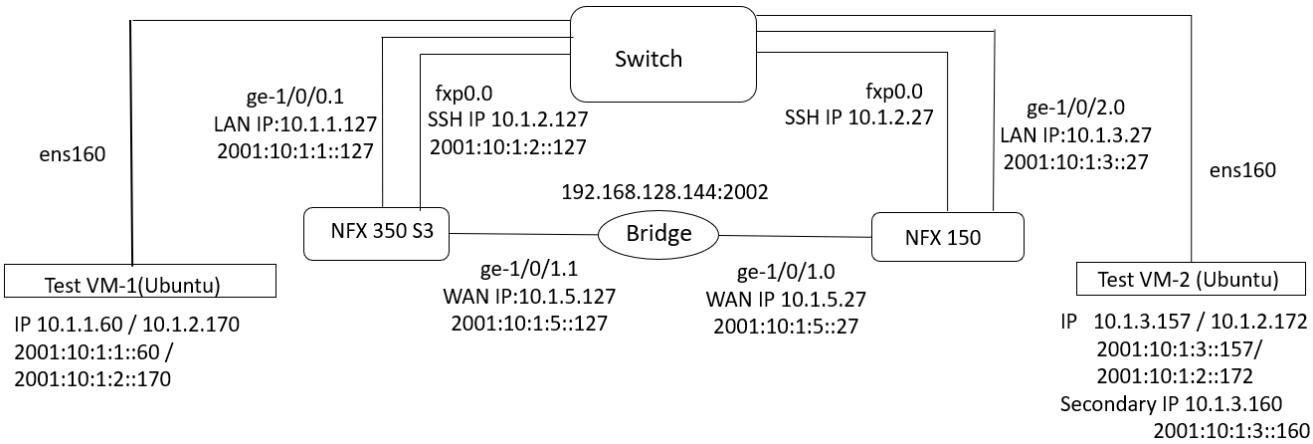
- The same CPU and firmware images are used across the models and hence do not affect security functionality. One model was tested for entropy, CAVP, and NDcPP testing.
- All other criteria are the same for these devices.

Between the NFX350 devices the hardware, software, libraries, management interfaces, and functionality are the same. Based on the analysis above, the following will sufficiently test the TOE:

- NFX350-S3 will be used for testing

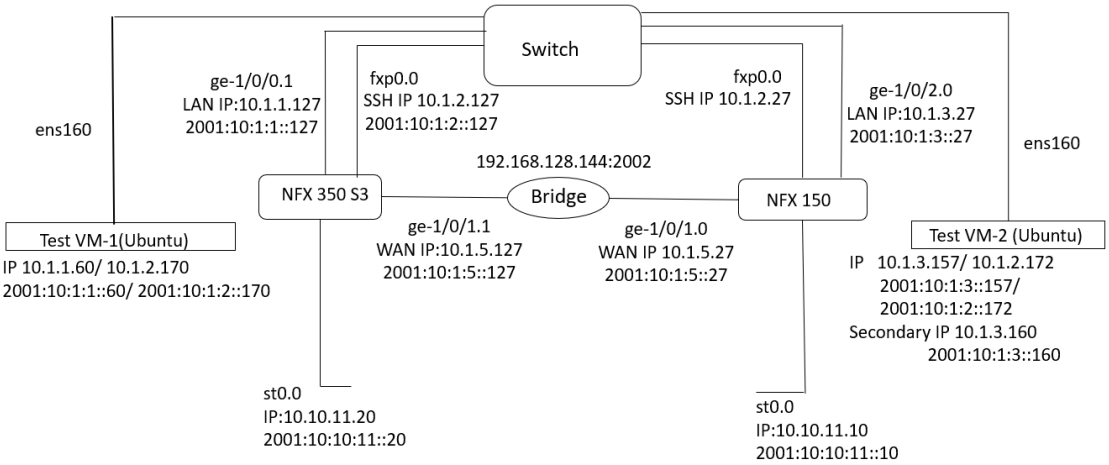
4 Test Bed Descriptions

4.1 Audit / Auth / SSHS / Update /IPS Audit



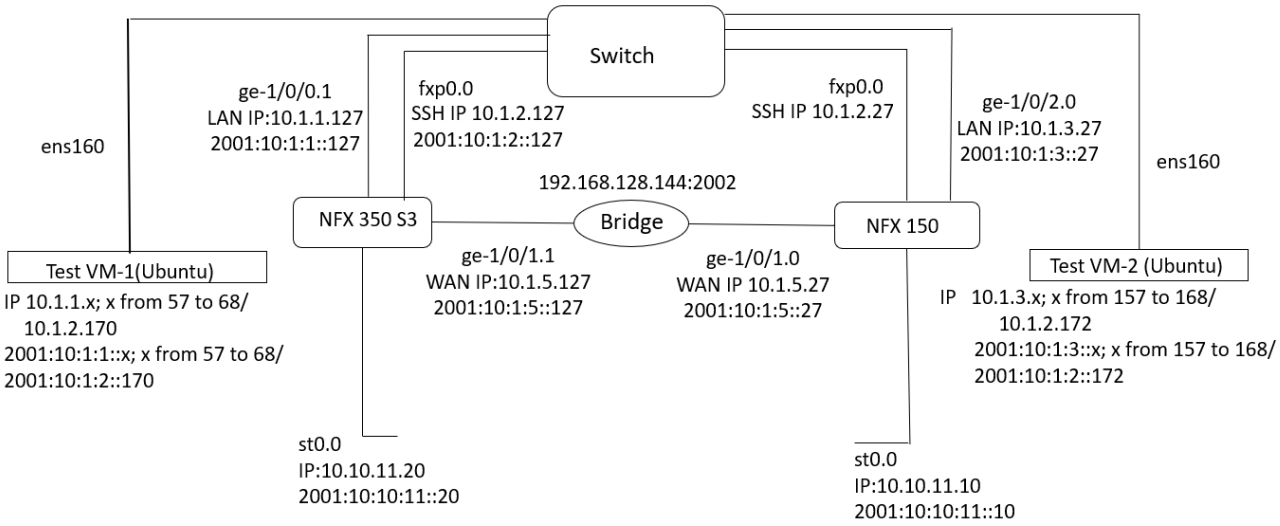
Name	OS with Version	Function	Protocols	Time	Tools (version)
NFX350-S3	JUNOS 20.3R3-S3.1	TOE	SSH IPSEC	Manually Set and Verified	NA
NFX150	JUNOS 20.3R3-S3.1	PEER	SSH IPSEC	Manually Set and Verified	NA
Test VM-1	Ubuntu 18.04	TOE Test VM	SSH	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) acumen-sshs netconfd (2.10-1) Python (3.6.9)
Test VM-2	Ubuntu 18.04	PEER Test VM	SSH	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Python (3.6.9)
Bridge	Raspbian GNU/Linux 9.4 (stretch)	Packet capture	NA	Manually Set and Verified	NA

4.2 X509 / IPSec / VPN Auth / Firewall



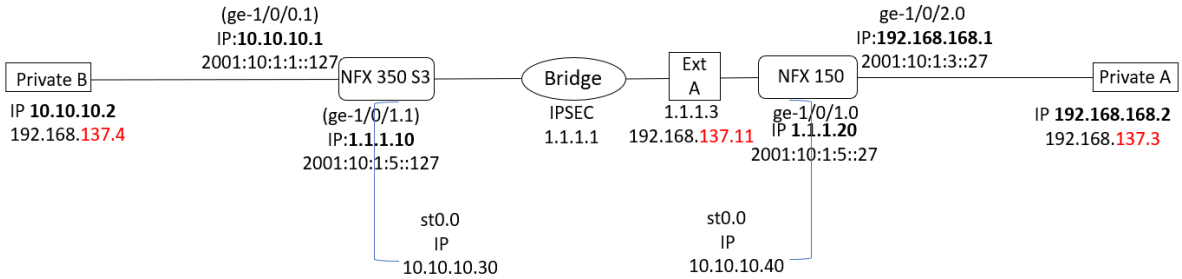
Name	OS with Version	Function	Protocols	Time	Tools (version)
NFX350-S3	JUNOS 20.3R3-S3.1	TOE	SSH IPsec	Manually Set and Verified	NA
NFX150	JUNOS 20.3R3-S3.1	PEER	SSH IPsec	Manually Set and Verified	NA
Test VM-1	Ubuntu 18.04	TOE Test VM	SSH IPsec	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Acumen x509-mod (1.1) Python (3.6.9) Strongswan (5.7.2)
Test VM-2	Ubuntu 18.04	PEER Test VM	SSH	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Python (3.6.9)
Bridge	Raspbian GNU/Linux 9.4 (stretch)	Packet Capture	NA	Manually Set and Verified	NA

4.3 IPS Policies / VPN Filter / Firewall



Name	OS with Version	Function	Protocols	Time	Tools (version)
NFX350-S3	JUNOS 20.3R3-S3.1	TOE	SSH IPSEC	Manually Set and Verified	NA
NFX150	JUNOS 20.3R3-S3.1	PEER	SSH IPSEC	Manually Set and Verified	NA
Test VM-1	Ubuntu 18.04	TOE Test VM	SSH IPsec	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Acumen x509-mod (1.1) Python (3.6.9) Strongswan (5.7.2)
Test VM-2	Ubuntu 18.04	PEER Test VM	SSH	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Python (3.6.9)
Bridge	Raspbian GNU/Linux 9.4 (stretch)	Packet capture	NA	Manually Set and Verified	NA

4.4 IPsec/X509



Name	OS with Version	Function	Protocols	Time	Tools (version)
NFX350-S3	JUNOS 20.3R3-S3.1	TOE	SSH IPSEC	Manually Set and Verified	NA
NFX150	JUNOS 20.3R3-S3.1	PEER	SSH IPSEC	Manually Set and Verified	NA
Private B	Ubuntu 18.04.5	TOE Test VM	SSH IPsec	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Acumen x509-mod (1.1) Python (3.6.9) Strongswan (5.7.2)
Private A	Ubuntu 18.04.5	PEER Test VM	SSH	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Python (3.6.9)
External A	Ubuntu 18.04.4	Strongswan peer Packet Capture	SSH IPsec	Manually Set and Verified	Scapy (2.3.3) Tcpdump (4.9.3) Python (3.6.9) Strongswan (5.7.2)
IPsec Bridge	Ubuntu (18.04.4)	Packet Capture	NA	Manually Set and Verified	tcpdump (4.9.3) libpcap (1.8.1)

5 Detailed Test Cases (TSS and Guidance Activities)

5.1 TSS and Guidance Activities (Auditing)

5.1.1 FAU_GEN.1

5.1.1.1 FAU_GEN.1 TSS 1

Objective	For the administrative task of generating/import of, changing, or deleting of cryptographic keys as defined in FAU_GEN.1.1c, the TSS should identify what information is logged to identify the relevant key.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to determine the verdict of this assurance activity. The evaluator confirmed that within this section it identified the following information that was logged in order to identify the relevant key in relation to import/generation, changing, or deletion of cryptographic keys: **In order to identify the key being operated on, the following details are recorded for all administrative actions relating to cryptographic keys (generating, importing, changing and deleting keys):** • **PKID – certificate id will be recorded when generating or deleting a key pair** • **IKE SPI – IP address of the initiator and responder recorded, together with the SPI, will be recorded when generating a key pair. The IP address of the initiator and responder will provide the unique link to the key identifier (SPI) of the key that has been destroyed in the session termination** • **SSH session keys– key reference provided by process id** • **SSH key configured for SSH public key authentication –the hash of the public key that is to be used for authentication is recorded in syslog** **For SSH (ephemeral) session keys the PID is used as the key reference to relate the key generation and key destruction audit events.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.1.2 FAU_GEN.1 TSS 3 (VPNGWMod)

Objective	The evaluator shall verify that the TSS describes how the TSF can be configured to log network traffic associated with applicable rules. Note that this activity may be addressed in conjunction with the TSS Evaluation Activities for FPF_RUL_EXT.1.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TSF can be configured to log network traffic associated with applicable rules. Upon investigation, the evaluator found that the TSS states that: **Traffic will be logged in accordance with 'log' operations defined in the rules. Source and destination addresses, source and destination ports, transport layer protocol, and TOE Interface are recorded in each log record.** **This activity has been further addressed with a combination of the TSS assurance activities for FPF_RUL_EXT.1** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.1.1.3 FAU_GEN.1 TSS 4 (VPNGWMod)

Objective	The evaluator shall verify that the TSS describes how the TOE behaves when one of its interfaces is overwhelmed by network traffic. It is acceptable for the TOE to drop packets that it cannot process, but under no circumstances is the TOE allowed to pass packets that do not satisfy a rule that allows the permit operation or belong to an allowed established session. It may not always be possible for the TOE to audit dropped packets due to implementation limitations. These limitations and circumstances in which the event of dropped packets is not audited shall be described in the TSS.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE behaves when one of its interfaces is overwhelmed by network traffic. Upon investigation, the evaluator found that the TSS states that: **Because of the nature of IPS event logs, log generation often happens in bursts and can generate a large volume of messages during an attack. To manage the volume of log messages, Junos supports log suppression, which suppresses multiple instances of the same log occurring from the same or similar sessions over the same period of time. IPS log suppression is enabled by default and can be customized based on the following configurable attributes:** • **Source/destination addresses;** • **Number of log occurrences after which log suppression begins;** • **Maximum number of logs that log suppression can operate on;** • **Time after which suppressed logs are reported.** **Suppressed logs are reported as single log entries containing the count of occurrences.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.1.4 FAU_GEN.1 TSS 5 (VPNGWMod)

Objective	The evaluator also verifies that the TSS describes the auditable events for IPsec peer session establishment that are required by the PP-Module.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the auditable events for IPsec peer session establishment that are required by the PP-Module. Upon investigation, the evaluator found that the TSS states that: **Junos OS creates and stores audit records for the following events (the detail of content recorded for each audit event is detailed in Table 12 and Table 13. Auditing is implemented using syslog.** • **Initiation/termination/failure of an IPsec trusted channel, including Session Establishment with peer** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.1.5 FAU_GEN.1 Guidance 1

Objective	The evaluator shall check the guidance documentation and ensure that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, shall be provided from the actual audit record).
Evaluator Findings	The evaluator examined the section titled **Table 2: Audit Records** in the AGD to verify that it provides an example of each auditable event required by FAU_GEN.1. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.1.6 FAU_GEN.1 Guidance 2

Objective	The evaluator shall also make a determination of the administrative actions related to TSF data related to configuration changes. The evaluator shall examine the guidance documentation and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP. The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding guidance documentation satisfies the requirements related to it.
Evaluator Findings	The evaluator examined the AGD to verify that it identifies administrative commands, including subcommands, scripts, and configuration files, that are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP. The evaluator first examined the entirety of AGD to determine what administrative commands are associated with each administrative activity. Upon investigation, the evaluator found that the following are applicable:

Administrative Activity	**Method (Command/GUI Configuration)**	**Section**
Startup	A series of CLI commands are provided for the configuration of audit logging	Configuring Audit Log Options
Shutdown	A series of CLI commands are provided for disabling audit logging	Configuring Audit Log Options
Logout	Users may terminate their sessions	Login and Logout Events Using SSH
Generating Keys (certificates)	Request security pki generate-key-pair	Configuring VPNs
Display system information	Show version	Installing Software on Network Services Platform with a Single Routing Engine

Creating Users	A series of CLI commands are provided for configuring an authorized administrator	Configuring Administrative Credentials and Privileges
Configuring Cas	request security pki ca-certificate load ca-profile ca-profile-ipsec filename /var/tmp/ca.cert	Configuring VPNs
Configuring Revocation Servers	request security pki crl load ca-profile ca-profile-ipsec filename /var/tmp/revoke.crl	Configuring IPsec VPN with RSA Signature as IKE Authentication on the Initiator
Generating CSRs	request security pki generate-certificate-request certificate-id ms-cert subject "CN=john doe,CN=10.1.1.2,OU=sales,O=example, L=Sunnyvale,ST=CA,C=US" email user@example.net filename ms-cert-req	Configuring IPsec VPN with RSA Signature as IKE Authentication on the Initiator
Performing Software Updates	request vmhost software add /<image-path/<junos package> no-copy no-validate reboot	Installing Software on Network Services Platform with a Single Routing Engine
Setting the Time	set date YYYYMMDDHHMM.ss	Configuring the time and date
Configuring Admin Timeout	set system login idle-timeout	Configuring the user session idle timeout
Configuring the Audit Server	A series of CLI commands are provided for configuration of the Audit Server	Configuring the Remote Syslog Server
Configuring Access Banner	Set system login message login-message-banner-text	Configuring a System Login Message and Announcement
Setting Password Length	Set system login password minimum length	Configuring Administrative Credentials and Privileges
Resetting the TOE	Request system zeroize	Operational Commands
Configuring SSH	A series of CLI commands are provided for configuration for SSH	Configuring SSH on the Evaluated Configuration for NDcPP
Configuring IKE/IPsec	A series of CLI commands are provided for configuration for IPsec	Configuring VPN on a Device Running Junos OS
Setting firewall rules	A series of commands are provided for configuring traffic filtering rules	Configuring traffic filtering rules

Next, the evaluator examined each of the test cases and identified test cases which exercised the above referenced functionality. The audit record associated with the configuration was captured. The following table identifies the test cases in which audit records for those configurations can be found.

Administrative Activity	Method (Command/GUI Configuration)	Test Case(s)
Startup	set system login class monitor permissions trace set system login user syslog-mon class monitor authentication ssh-rsa “ssh-rsa xxxxx syslog-monitor key pair” set system services netconf ssh set system syslog file <filename> any any	FPT_STG_EXT.1 T1
Shutdown	Deleting any of the configuration items from the Startup row	FAU_STG_EXT.1 T1
Login	Via SSH or console	FTA_TAB.1 T1
Logout	exit	FTA_SSL.4 T1 and T2
Generating Keys (certificates)	Request security pki generate-key-pair	FIA_X509_EXT.1.2/Rev T1
Display system information	Show version	FPT_TUD_EXT.1 T1
Creating Users	Set system login class set system login user set system login password format sha256	FAU_STG_EXT.1 T1
Configuring CAs	request security pki ca-certificate load ca-profile ca-profile-ipsec filename /var/tmp/ca.cert	FIA_X509_EXT.1.2/Rev T1
Configuring Revocation Servers	request security pki crl load ca-profile ca-profile-ipsec filename /var/tmp/revoke.crl	FIA_X509_EXT.2 T1
Generating CSRs	request security pki generate-certificate-request certificate-id ms-cert subject "CN=john doe,CN=10.1.1.2,OU=sales,O=example, L=Sunnyvale,ST=CA,C=US" email user@example.net filename ms-cert-req	FIA_X509_EXT.3 T1
Performing Software Updates	Request system software add /<image-path/<junos package> no-copy no-validate reboot	FPT_TUD_EXT.1 T1
Setting the Time	set date YYYYMMDDHHMM.ss	FPT_STM.1 T1
Configuring Admin Timeout	Set system login idle-timeout	FTA_SSL_EXT.1.1 T1

	Configuring the Audit Server	set system login class monitor permissions trace set system login user syslog-mon class monitor authentication ssh-rsa "ssh-rsa xxxxx syslog-monitor key pair" set system services netconf ssh set system syslog file <filename> any any	FAU_STG_EXT.1 T1
	Configuring Access Banner	Set system login message login-message-banner-text	FTA_TAB.1 T1
	Setting Password Length	Set system login password minimum length	FIA_PMG_EXT.1.1 T1
	Configuring SSH	Set system services ssh hostkey-algorithm <> set system services ssh key-exchange <> set system services ssh macs <> set system services ssh ciphers <>	FCS_SSHS_EXT.1.2 T1
	Configuring IKE/IPsec	Set security ike <> set security ipsec <> set security policies <>	FCS_IPSEC_EXT.1.1 T1
	Configuring firewall rules	A series of commands are provided for configuring traffic filtering rules	FFW_RUL_EXT.1
	Based on these findings, this assurance activity is considered satisfied.		
Verdict	Pass.		

5.1.1.7 FAU_GEN.1 Guidance 3 (VPNGWMod)

Objective	The evaluator shall verify that the operational guidance describes how to configure the TSF to result in applicable network traffic logging. Note that this activity may be addressed in conjunction with the guidance Evaluation Activities for FPF_RUL_EXT.1.
Evaluator Findings	The evaluator examined the section titled **Configuring Audit Log Options in the Evaluated Configuration** in the AGD to verify describes how to configure the TSF to result in applicable network traffic logging. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.1.8 FAU_GEN.1 Guidance 4 (FWMod)

Objective	In addition to the Evaluation Activities specified in the Supporting Document for the Base-PP, the evaluator shall check the guidance documentation to ensure that it describes the audit records specified in Table 2 of the PP-Module in addition to those required by the Base-PP. If the optional SFR FFW_RUL_EXT.2 is claimed by the TOE, the evaluator shall also check the

	guidance documentation to ensure that it describes the relevant audit record specified in Table 3 of the PP-Module.
Evaluator Findings	The evaluator examined **Table 2: Audit Records** in the AGD to verify that it describes the audit records specified in Table 2 of the PP-Module in addition to those required by the Base-PP, along with the relevant audit record specified in Table 3 of the PP-Module for the optional SFR FFW_RUL_EXT.2 . Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.2 FAU_GEN.1/IPS

5.1.2.1 FAU_GEN.1/IPS TSS 1

Objective	The evaluator shall verify that the TSS describes how the TOE can be configured to log IPS data associated with applicable policies. The evaluator shall verify that the TSS describes what (similar) IPS event types the TOE will combine into a single audit record along with the conditions (e.g., thresholds and time periods) for so doing. The TSS shall also describe to what extent (if any) that may be configurable. For IPS_SBD_EXT.1, for each field, the evaluator shall verify that the TSS describes how the field is inspected and if logging is not applicable, any other mechanism such as counting that is deployed.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE can be configured to log IPS data associated with applicable policies. Upon investigation, the evaluator found that the TSS states that: **Junos OS creates and stores audit records for the following events (the detail of content recorded for each audit event is detailed in Table 12 and Table 13. Auditing is implemented using syslog** • **Start-up and shut-down of the IPS functions** • **All dissimilar IPS events and reactions** • **Totals of similar events and reactions occurring within a specified time period** • **Modification of an IPS policy element** • **Modification of which IPS policies are active on a TOE interface** • **Enabling/disabling a TOE interface with IPS policies applied** • **Modification of which mode(s) is/are active on a TOE interface** • **Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy** • **Inspected traffic matches a signature-based IPS policy with logging enabled** • **Inspected traffic matches an anomaly-based IPS policy** In addition, the evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes what IPS event types the TOE will combine into a single audit record along with the conditions for so doing. Upon investigation, the evaluator found that the TSS states that:

	Because of the nature of IPS event logs, log generation often happens in bursts and can generate a large volume of messages during an attack. To manage the volume of log messages, Junos supports log suppression, which suppresses multiple instances of the same log occurring from the same or similar sessions over the same period of time. IPS log suppression is enabled by default and can be customized based on the following configurable attributes: • **Source/destination addresses;** • **Number of log occurrences after which log suppression begins;** • **Maximum number of logs that log suppression can operate on;** • **Time after which suppressed logs are reported.** **Suppressed logs are reported as single log entries containing the count of occurrences.** In addition, the evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the field is inspected and if logging is not applicable, any other mechanism such as counting that is deployed. Upon investigation, the evaluator found that the TOE logs for all applicable audit events. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.2.2 FAU_GEN.1/IPS Guidance

Objective	The evaluator shall verify that the operational guidance describes how to configure the TOE to result in applicable IPS data logging. The evaluator shall verify that the operational guidance provides instructions for any configuration that may be done in regard to logging similar events (e.g., setting thresholds, defining time windows, etc.).
Evaluator Findings	The evaluator examined the section titled **Configuring the IDP Extended Package** in the AGD to verify that it describes how to configure the TOE to result in applicable IPS data logging. Upon investigation, the evaluator found that the AGD states that how to configure audit log options in the evaluated configuration. In addition, the evaluator examined the section titled **Configuring the IDP Extended Package** in the AGD to verify that it provides instructions for any configuration that may be done in regard to logging similar events. Upon investigation, the evaluator found that the AGD states that IPS log suppression is enabled by default and can be customized based on the following configurable attributes: • Source/destination addresses; • Number of log occurrences after which log suppression begins; • Maximum number of logs that log suppression can operate on; • Time after which suppressed logs are reported. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3 FAU_STG_EXT.1

5.1.3.1 FAU_STG_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to ensure it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided. Upon investigation, the evaluator found that the TSS states that: **Syslog can be configured to store the audit logs locally, and optionally to send them to one or more syslog log servers in real time via Netconf over SSH** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.2 FAU_STG_EXT.1 TSS 2

Objective	The evaluator shall examine the TSS to ensure it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access. Upon investigation, the evaluator found that the TSS states that: **Only a Security Administrator can read log files or delete log and archive files through the CLI interface or through direct access to the filesystem having first authenticated as a Security Administrator. The syslogs are automatically deleted locally according to configurable limits on storage volume. The default maximum size is 1Gb. The default maximum size can be modified by the user, using the "size" argument for the "set system syslog" CLI command.** **The Junos OS defines an active log file and a number of "archive" files (10 by default, but configurable from 1 to 1000). When the active log file reaches its maximum size, the logging utility closes the file, compresses it, and names the compressed archive file 'logfile.0.gz'. The logging utility then opens and writes to a new active log file. When the new active log file reaches the configured maximum size, 'logfile.0.gz' is renamed 'logfile.1.gz', and the active log file is closed, compressed, and renamed 'logfile.0.gz'. When the maximum number of archive files is reached and when the size of the active file reaches the configured maximum size, the contents of the oldest archived file are deleted so the current active file can be archived.** **A 1Gb syslog file takes approximately 0.25Gb of storage when archived** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.3 FAU_STG_EXT.1 TSS 3

Objective	The evaluator shall examine the TSS to ensure it describes whether the TOE is a standalone TOE that stores audit data locally or a distributed TOE that stores audit data locally on each TOE component or a distributed TOE that contains TOE components that cannot store audit data locally on themselves but need to transfer audit data to other TOE components that can store audit data locally. The evaluator shall examine the TSS to ensure that for distributed TOEs it contains a list of TOE components that store audit data locally. The evaluator shall examine the TSS to ensure that for distributed TOEs that contain components which do not store audit data locally but transmit their generated audit data to other components it contains a mapping between the transmitting and storing TOE components.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes whether the TOE is a standalone TOE that stores audit data locally or a distributed TOE that stores audit data locally on each TOE component or a distributed TOE that contains TOE components that cannot store audit data locally on themselves but need to transfer audit data to other TOE components that can store audit data locally. Upon investigation, the evaluator found that the TSS states that: **The TOE is a standalone device wherein syslog can be configured to store the audit logs locally, and optionally to send them to one or more syslog log servers in real time via Netconf over SSH. Local audit log are stored in /var/log/ in the underlying filesystem.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.4 FAU_STG_EXT.1 TSS 4

Objective	The evaluator shall examine the TSS to ensure that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE shall also be detailed in the TSS.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details the behavior of the TOE when the storage space for audit data is full. Upon investigation, the evaluator found that the TSS states that: **The syslogs are automatically deleted locally according to configurable limits on storage volume. The default maximum size is 1Gb. The default maximum size can be modified by the user, using the "size" argument for the "set system syslog" CLI command.** **The Junos OS defines an active log file and a number of "archive" files (10 by default, but configurable from 1 to 1000). When the active log file reaches its maximum size, the logging utility closes the file, compresses it, and names the compressed archive file 'logfile.0.gz'. The logging utility then opens and writes to a new active log file. When the new active log file reaches the configured maximum size, 'logfile.0.gz' is renamed 'logfile.1.gz', and the active log file is closed, compressed, and renamed 'logfile.0.gz'. When the maximum number of archive files is reached and when the size of the active file reaches the configured maximum size, the contents of the oldest archived file are deleted so the current active file can be archived** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.1.3.5 FAU_STG_EXT.1 TSS 5

Objective	The evaluator shall examine the TSS to ensure that it details whether the transmission of audit information to an external IT entity can be done in realtime or periodically. In case the TOE does not perform transmission in realtime the evaluator needs to verify that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details whether the transmission of audit information to an external IT entity can be done in realtime or periodically. Upon investigation, the evaluator found that the TSS states that: **Syslog can be configured to store the audit logs locally, and optionally to send them to one or more syslog log servers in real time via Netconf over SSH** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.6 FAU_STG_EXT.1 Guidance 1

Objective	The evaluator shall also examine the guidance documentation to ensure it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.
Evaluator Findings	The evaluator examined the section titled **Configuring the Remote Syslog Server** in the guidance documentation to verify that it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server. Upon investigation, the evaluator found that the AGD states that the TOE securely sends traffic to an external audit server via SSH and IPSEC. Next, the evaluator found that AGD provides instructions for configuring the secure connection between the TOE and the remote audit server via CLI. The evaluator found that AGD defines the following requirements for audit server, the syslog server must have an SSH client with NETCONF configured to receive the streamed syslog messages. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.7 FAU_STG_EXT.1 Guidance 2

Objective	The evaluator shall also examine the guidance documentation to determine that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. For example, when an audit event is generated, is it simultaneously sent to the external server and the local store, or is the local store used as a buffer and "cleared" periodically by sending the data to the audit server.
Evaluator Findings	The evaluator examined the section titled **Configuring the remote syslog server** in the guidance documentation to verify that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. Upon investigation, the

	evaluator found that the AGD states that that AGD describes the relationship between local and external audit data is as follows: **A secure Junos OS environment requires auditing of events and storing them in a local audit file. The recorded events are simultaneously sent to an external syslog server. A syslog server receives the syslog messages streamed from the NFX350 device. The syslog server must have an SSH client with NETCONF support configured to receive the streamed syslog messages.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.1.3.8 FAU_STG_EXT.1 Guidance 3

Objective	The evaluator shall also ensure that the guidance documentation describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behavior of the TOE for each possible configuration. The description of possible configuration options and resulting behavior shall correspond to those described in the TSS.
Evaluator Findings	The evaluator examined the section titled **Configuring Audit Log Options** in the guidance documentation to verify that it describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behavior of the TOE for each possible configuration. Upon investigation, the evaluator found that the AGD describes how to specify the number of files to be archived, the file in which to log data, the size of the files to be archived, and the system message format. Next, the evaluator compared the exhausted local audit handling description found in the AGD to the description provided by the TSS of the ST. The descriptions of the behavior found in AGD and ST are consistent. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2 TSS and Guidance Activities (Cryptographic Support)

Note that Test activities in the SD that are typically addressed by referencing CAVP certs are addressed in this section and are identified as "Test/CAVP" activities.

5.2.1 FCS_CKM.1

5.2.1.1 FCS_CKM.1 TSS 1

Objective	The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE. 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 **TOE Summary Specification** in the Security Target to verify that the TSS identifies the key sizes supported by the TOE. Upon investigation, the evaluator found that the TSS states that: **The TOE's cryptographic module generates asymmetric keys. The asymmetric keys produced are:** • **RSA 2048, 4096 bit** • **ECC (P-256, P-384, P-521)** • **DH group 14 (2048 bits)**

Usage of the keys in protocols is specified in Table 4

Table 16 of ST states that:

Table 4 – Protocol Usage of Cryptographic Algorithms

Protocol	Key Exchange	Auth	Cipher	Integrity
IKEv1	Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384)	RSA 2048 ECDSA P-256 ECDSA P-384 Pre-Shared Key	AES CBC 128 AES-CBC-192 AES CBC 256	HMAC-SHA-256-128 HMAC-SHA-384-192
IKEv2	Group 14 (modp 2048) Group 19 (P-256) Group 20 (P-384)	RSA 2048 ECDSA P-256 ECDSA P-384 Pre-Shared Key	AES CBC 128 AES-CBC-192 AES CBC 256 AES GCM 128 AES GCM 192 AES GCM 256	HMAC-SHA-256-128 HMAC-SHA-384-192
IPsec ESP	IKEv1 with optional: • Group 14 (modp 2048) • Group 19 (P-256) • Group 20 (P-384)	IKEv1	AES CBC 128 AES-CBC-192 AES CBC 256	HMAC-SHA-256-128
	IKEv2 with optional: • Group 14 (modp 2048) • Group 19 (P-256) • Group 20 (P-384)	IKEv2	AES CBC 128 AES-CBC-192 AES CBC 256 AES GCM 128 AES-GCM-192 AES GCM 256	HMAC-SHA-256-128
SSHv2	Diffie-Hellman Group 14 (modp 2048) ECDH-sha2-nistp256 ECDH-sha2-nistp384 ECDH-sha2-nistp521	ECDSA P-256 ECDSA P-384 ECDSA P-521	AES CTR 128 AES CTR 256 AES CBC 128 AES CBC 256	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512

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

Verdict	Pass

5.2.1.2 FCS_CKM.1 Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.
Evaluator Findings	The evaluator examined the section titled **Configuring Event Logging to a Remote Server when Initiating the Connection from the Remote Server** and **Configuring an IPsec VPN with an RSA Signature for IKE 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 cryptographic protocols defined in the Security Target. Upon investigation, the evaluator found that the AGD states the CLI commands for configuring the appropriate key generation scheme and key size on the TOE. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.1.3 FCS_CKM.1 Test/CAVP 1

Objective	The evaluator shall verify the key generation mechanisms supported by the TOE.
Evaluator Findings	CAVP Certs: #A2574, #A2575, #A2576, #A2577 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.2 FCS_CKM.1/IKE

5.2.2.1 FCS_CKM.1/IKE TSS 1

Objective	The evaluator shall check to ensure that the TSS describes how the key-pairs are generated. In order to show that the TSF implementation complies with FIPS PUB 186-4, the evaluator shall ensure that the TSS contains the following information: • The TSS shall list all sections of Appendix B to which the TOE complies. • For each applicable section listed in the TSS, for all statements that are not "shall" (that is, "shall not", "should", and "should not"), if the TOE implements such options it shall be described in the TSS. If the included functionality is indicated as "shall not" or "should not" in the standard, the TSS shall provide a rationale for why this will not adversely affect the security policy implemented by the TOE; • For each applicable section of Appendix B, any omission of functionality related to "shall" or “should” statements shall be described; Any TOE-specific extensions, processing that is not included in the Appendices, or alternative Implementations allowed by the Appendices that may impact the security requirements the TOE is to enforce shall be described.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the key-pairs are generated. Upon investigation, the evaluator found that the TSS states that: **Asymmetric keys are generated in accordance with NIST SP 800-56A and FIPS PUB 186-4 for IKE with IPSec. The TOE complies with section 5.6 of NIST SP 800-56A regarding asymmetric key pair generation. The TOE implements all of the "shall" and "should" requirements and none of the "shall not" or "should not" from FIPS PUB 186-4 Appendix B.3.3 and B.4.2.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.2.2 FCS_CKM.1/IKE Guidance 1

Objective	The evaluator shall check that the operational guidance describes how the key generation functionality is invoked, and describes the inputs and outputs associated with the process for

	each signature scheme supported. The evaluator shall also check that guidance is provided regarding the format and location of the output of the key generation process.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it describes how the key generation functionality is invoked, and describes the inputs and outputs associated with the process for each signature scheme supported. Upon investigation, the evaluator found that the AGD states the CLI commands for invoking the key generation functionality which includes the inputs and outputs associated with the same. The evaluator examined the section titled **Configuring Roles and Authentication Methods** in the AGD to verify that it describes the format and location of the output of the key generation process. Upon investigation, the evaluator found that the AGD states that:

CSP	Description	Method of storage	Storage location	Zeroization Method
IKE Private Host key	Private authentication key used in IKE. RSA 2048, RSA 4096, ECDSA P-256, ECDSA P-384	Plaintext	Disk (mapped to SDD)/Memory	'clear security IKE security-association' command or reboot the box. Private keys stored in flash are not zeroized unless an explicit "request system zeroize" is executed.
IKE-SKEYID	IKE master secret used to derive IKE and IPsec ESP session keys	Plaintext	Memory	'clear security IKE security-association' command or reboot the box
IKE Session Keys	IKE session key. AES, HMAC	Plaintext	Memory	'clear security IKE security-association' command or reboot the box
ESP Session Key	ESP session keys. AES, HMAC	Plaintext	Memory	'clear security ipsec security-association' or reboot the box.
IKE-DH Private Exponent	Ephemeral DH private exponent used in IKE. DH N = 224 bit or N = 256 bit, ECDH P-256, or ECDH P-384	Plaintext	Memory	'clear security IKE security-association' command or reboot the box.
IKE-PSK	Pre-shared authentication key used in IKE.	Hashed	Disk (mapped to SDD)/Memory	'clear security IKE security-association' command or reboot the box. Key values stored in flash are not zeroized unless an explicit "request system zeroize" is executed.

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

Verdict	Pass.

5.2.2.3 FCS_CKM.1/IKE Test/CAVP 1

Objective	The evaluator shall verify the key generation mechanisms supported by the TOE.
Evaluator Findings	CAVP Cert: #A2577. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.3 FCS_CKM.2

5.2.3.1 FCS_CKM.2 TSS 1 **[TD0580]**

Objective	The evaluator shall 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. It is sufficient to provide the scheme, SFR, and service in the TSS.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. Upon investigation, the evaluator found that the TSS states that: **Asymmetric keys are also generated in accordance with FIPS PUB 186-4 Appendix B.3.3 for RSA Schemes and Appendix B.4.2 for ECC Schemes for SSH and IPsec communications. The TOE implements Diffie-Hellman group 14, using the modulus and generator specified by Section 3 of RFC3526. Usage of key agreement in protocols is specified in Table 4.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.3.2 FCS_CKM.2 Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s).
Evaluator Findings	The evaluator examined the section titled **Configuring SSH on the Evaluated Configuration for NDcPP** in the AGD to verify that it instructs the administrator how to configure the TOE to use the selected key establishment scheme(s). Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the appropriate key establishment scheme on the TOE. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.3.3 FCS_CKM.2 Test/CAVP 1

Objective	The evaluator shall verify the key establishment mechanisms supported by the TOE.
Evaluator Findings	CAVP Certs: #A2577 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.4 FCS_CKM.4

5.2.4.1 FCS_CKM.4 TSS 1

Objective	The evaluator examines the TSS to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case. For the purpose of this Evaluation Activity the relevant keys are those keys that are relied upon to support any of the SFRs in the Security Target. The evaluator confirms that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for2). In particular, if a TOE claims not to store plaintext keys in non-volatile memory then the evaluator checks that this is consistent with the operation of the TOE.
Evaluator Findings	The evaluator examined the section titled **Table 19-Key Destructions** in the Security Target to verify that the TSS lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case and that the TSS description of keys and storage locations is consistent with the functions carried out by the TOE.. Upon investigation, the evaluator found that the TSS states that:

Table 5 – Key Descriptions

Keys/CSPs	Purpose	Method of Storage	Storage Location	Method of Zeroization
SSH Private Host Key	The first time SSH is configured the set of Host keys is generated. Used to identify the host. ecdsa-sha2-nistp256 (ECDSA P-256, ECDSA P-384, ECDSA P-521) and/or ssh-rsa (RSA 2048)	Plaintext	File format on Disk (mapped to SDD)	When the appliance is recommissioned, the config files (including CSP files such as SSH keys) are removed using the Linux `shred` command to wipe the underlying persistent storage media.
SSH Private Host Key	Loaded into memory to complete session establishment	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)
SSH Session Key	Session keys used with SSH, AES 128, 256,	Plaintext	Memory	Memory free() operation is performed by

		hmac-sha-1, hmac-sha2-256 or hmac-sha2-512 key (160, 256 or 512), DH Private Key (2048 or elliptic curve 256/384/521-bits)			Junos upon session termination (when released by the Junos VM, the hypervisor releases the memory and places it in the free pool)
	User Password	Plaintext value as entered by user	Plaintext as entered	Processed in Memory	Memory free() operation is performed by Junos (when released by the Junos VM, the hypervisor releases the memory and places it in the free pool)
			Hashed when stored (HMAC-sha1, sha256, sha512)	Stored on disk (mapped to SDD)	When the appliance is recommissioned, the config files (including the obfuscated password) are removed using the "request system zeroize" option.
	RNG State	Internal state and seed key of RNG	Plaintext	Memory	Handled by kernel, overwritten with zero's at reboot.
	IKE Private Host key	Private authentication key used in IKE. RSA 2048, RSA 4096, ECDSA P-256, ECDSA P-384	Plaintext	Disk (mapped to SDD)/Memory	'clear security IKE security-association' command or reboot the box. Private keys stored in flash are not zeroized unless an explicit "request system zeroize" is executed.
	IKE-SKEYID	IKE master secret used to derive IKE and IPsec ESP session keys	Plaintext	Memory	'clear security IKE security-association' command or reboot the box
	IKE Session Keys	IKE session key. AES, HMAC	Plaintext	Memory	'clear security IKE security-association'

				command or reboot the box
ESP Session Key	ESP session keys. AES, HMAC	Plaintext	Memory	'clear security ipsec security-association' or reboot the box.
IKE-DH Private Exponent	Ephemeral DH private exponent used in IKE. DH N = 224 bit or N = 256 bit, ECDH P-256, or ECDH P-384	Plaintext	Memory	'clear security IKE security-association' command or reboot the box.
IKE-PSK	Pre-shared authentication key used in IKE.	Hashed	Disk (mapped to SDD)/Memory	'clear security IKE security-association' command or reboot the box. Key values stored in flash are not zeroized unless an explicit "request system zeroize" is executed.
ecdh private keys	Loaded into memory to complete key exchange in session establishment	Plaintext	Memory	Memory free() operation is performed by Junos upon session termination (when released by the Junos VM, the WRL hypervisor erases the released memory before it is placed in the free pool)

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

Verdict	Pass

5.2.4.2 FCS_CKM.4 TSS 2

Objective	The evaluator shall check to ensure the TSS identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** and **Table 19** in the Security Target to verify that the TSS identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys. Upon investigation, the evaluator found that the TSS states that:

	Table 5 of the Security Target lists all relevant keys as well as their origin, storage location, situations in which keys are destroyed and key destruction method used. The TOE stores plaintext keys in volatile and non-volatile memory. Keys listed are consistent with the functions carried out by the TOE. There are no configurations that do not conform to the key destruction requirement. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.4.3 FCS_CKM.4 TSS 3

Objective	Where the TSS identifies keys that are stored in a non-plaintext form, the evaluator shall check that the TSS identifies the encryption method and the key-encrypting-key used, and that the key-encrypting-key is either itself stored in an encrypted form or that it is destroyed by a method included under FCS_CKM.4.
Evaluator Findings	The evaluator examined the Security Target to verify that keys are not stored in non-plaintext form under FCS_CKM.4. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.4.4 FCS_CKM.4 TSS 4

Objective	The evaluator shall check that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below). Note that reference may be made to the Guidance Documentation for description of the detail of such cases where destruction may be prevented or delayed.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement. Upon investigation, the evaluator found that the TSS states that: **There are no configurations that do not conform to the key destruction requirement.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.4.5 FCS_CKM.4 TSS 5

Objective	Where the ST specifies the use of “a value that does not contain any CSP” to overwrite keys, the evaluator examines the TSS to ensure that it describes how that pattern is obtained and used, and that this justifies the claim that the pattern does not contain any CSPs.
Evaluator Findings	The ST does not select “a value that does not contain any CSP”. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.4.6 FCS_CKM.4 Guidance 1

Objective	A TOE may be subject to situations that could prevent or delay key destruction in some cases. The evaluator shall 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 supporting information used). The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer.
Evaluator Findings	The evaluator checked the AGD and did not discover any configuration or circumstances that do not conform to the key destruction requirement. The evaluator determined that this description is consistent with the TSS. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.5 FCS_COP.1/DataEncryption

5.2.5.1 FCS_COP.1/DataEncryption TSS 1

Objective	The evaluator shall examine the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption.
Evaluator Findings	The evaluator examined the section titled **Table 16** in the Security Target to verify that the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.5.2 FCS_COP.1/DataEncryption Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it provides guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the required encryption/decryption mode and key size. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.5.3 FCS_COP.1/DataEncryption Test/CAVP 1

Objective	The evaluator shall verify the implementation of encryption supported by the TOE.
Evaluator Findings	CAVP AES Certs: #A2576, #A2577. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.6 FCS_COP.1/SigGen

5.2.6.1 FCS_COP.1/SigGen TSS 1

Objective	The evaluator shall examine the TSS to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS to ensure it specifies the cryptographic algorithm and key size supported by the TOE for signature services. Upon investigation, the evaluator found that the TSS states that: **The TOE performs cryptographic signature services (generation and verification) using the following cryptographic algorithms:** • **RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits or 4096 bits]** • **Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256-bits, 384-bits, 512-bits]** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.6.2 FCS_COP.1/SigGen Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services.
Evaluator Findings	The evaluator examined the section titled **Configuring an IPsec VPN with an RSA Signature for IKE Authentication** and **Configuring an IPsec VPN with an ECDSA Signature for IKE Authentication** in the AGD to verify that it provides guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services. Upon investigation, the evaluator found that the AGD states the steps to configure the TOE to use the appropriate cryptographic algorithm and key size for signature services. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.6.3 FCS_COP.1/SigGen Test/CAVP 1

Objective	The evaluator shall verify the implementation of signature generation and verification supported by the TOE.
Evaluator Findings	CAVP RSA SigGen&SigVer (186-4) Certs: #A2576, #A2577 CAVP ECDSA&SigVer SigGen (186-4) Certs: #A2576, #A2577 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.7 FCS_COP.1/Hash

5.2.7.1 FCS_COP.1/Hash TSS 1

Objective	The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS documents the association of the hash function with other TSF cryptographic functions. Upon investigation, the evaluator found that the TSS states that:

Hash functions are used in support of protocols as specified in Table 4 and Table 6. SHA-1, SHA-256 and SHA-512 are also used for password hashing.

Table 6 – CAVP Algorithm Certificate References

Crypto Module/ Library	FIPS PUB	Algorithm, Mode, Keysize, Function, Hashing, Usage	Certificate Number	Processor
OpenSSL libcrypto – OpenSSL IPSec Daemon	FIPS 197, SP 800-38D	AES-GCM (128, 192, 256) (Encrypt, Decrypt, AEAD)	#A2577	Intel® Xeon D-2146NT (Skylake)
	FIPS 197, SP 800-38A	AES-CBC (128, 192, 256) (Encrypt, Decrypt)	#A2577	
	FIPS 180-4	SHS: SHA (256) Byte Oriented (Message Digest Generation)	#A2577	
	FIPS 198-1	HMAC-SHA (256) Byte Oriented (Message Authentication)	#A2577	
Junos OS quicksec - ipsec-7 – IKED Daemon	FIPS 197, SP 800-38A	AES-CBC (128, 192, 256) (Encrypt, Decrypt)	A2576	
	FIPS 197, SP 800-38D	AES-GCM (128, 256) (Encrypt, Decrypt, AEAD)	A2576	
	FIPS 180-4	SHS: SHA (256, 384) Byte Oriented (Message Digest Generation)	A2576	
	FIPS 198-1	HMAC-SHA (256, 384)	A2576	
	FIPS 186-4	RSA PKCS1_V1_5 (n=2048, 4096 (SHA 256) (SigGen, SigVer)	A2576	

			FIPS 186-4	ECDSA (P-256 w/ SHA-256) ECDSA (P-384 w/ SHA-384) (KeyGen, SigGen, SigVer)	A2576		
		Junos OS quicksec - ipsec-7 – All Daemons	SP 800-90A	DRBG (HMAC-SHA-2-256) (Random Bit Generation)	A2576		
		OpenSSL libcrypto – IKED Daemon	SP 800-56A	KAS ECC SSC (P-256, P-384, P-521)	#A2577		
		OpenSSL libcrypto – SSHD Daemon and PKID Daemon	FIPS 197, SP800-38A	AES-CBC/CTR (128, 256) (Encrypt, Decrypt)	#A2577		
			FIPS 180-4	SHS: SHA (1, 256, 384) Byte Oriented (Message Digest Generation, SSH KDF Primitive)	#A2577		
			FIPS 180-4	SHS: SHA-512 Byte Oriented (Message Digest Generation)	#A2577		
			FIPS 198-1	HMAC-SHA (1, 256), (512) Byte Oriented (Message Authentication DRBG Primitive)	#A2577		
			SP 800-56A	KAS ECC SSC (P-256, P-384, P-521)	#A2577		
		OpenSSL libcrypto – All Daemons	SP 800-90A	DRBG (HMAC-SHA-2-256) (Random Bit Generation)	#A2577		

		FIPS 186-4	RSA KeyGen (n=2048, 4096 (SHA-256)	#A2577	
		FIPS 186-4	RSA PKCS1_V1_5 (n=2048, 4096 (SHA-256) (SigGen, SigVer)	#A2577	
		FIPS 186-4	ECDSA [P-256 (SHA-256)], [P-384 (SHA-384)], [P-521 (SHA-521)]] (SigGen, SigVer, KeyGen, KeyVer)	#A2577	
	Junos OS libmd – MGD Daemon, Password Hashing	FIPS 198-1	HMAC-SHA (1, 256) Byte Oriented (Message Authentication DRBG Primitive)	A2575	
		FIPS 180-4	SHS: SHA (256, 512) Byte Oriented (Message Digest Generation)	A2575	
	Junos OS Kernel – Veriexec	FIPS 180-4	SHS: SHA (1, 256) Byte Oriented (Message Digest Generation)	A2574	
	Junos OS Kernel – kernel-hmac drbg	FIPS 198-1	HMAC-SHA (256) Byte Oriented (DRBG Primitive)	A2574	
	Junos OS Kernel – kernel-hmac drbg	SP800-90A	DRBG (HMAC-2-SHA-256) (Random Bit Generation)	A2574	
	Based on these findings, this assurance activity is considered satisfied.				
Verdict	Pass				

5.2.7.2 FCS_COP.1/Hash Guidance 1

Objective	The evaluator checks the AGD documents to determine that any configuration that is required to configure the required hash sizes is present.
Evaluator Findings	The evaluator examined the section titled Configuring SSH on the Evaluated Configuration , "Configuring VPN on a Device Running Junos OS", and "Configuring a Network Device Collaborative Protection Profile Authorized Administrator" in the AGD to verify that it presents any configuration that is required to configure the required hash sizes. Upon investigation, the evaluator found that the AGD states that the following hash sizes configurations are required: • Password hashing – SHA-256 or SHA-512 • SSH Key Exchange – SHA-1, SHA-256, or SHA-384 • SSH Authentication – HMAC-SHA-1, HMAC-SHA-256, or HMAC-SHA-512 • IKE HMAC – HMAC-SHA-256 or HMAC-SHA-384 • ESP HMAC - HMAC-SHA-1 or HMAC-SHA-256 • The firmware integrity and digital signature hash uses are automatically selected based on cryptographic usage. Additionally, the evaluator compared the instructions listed in AGD to the actual usage of the TOE during testing and found that the listed configuration covers each way the hash functions can be configured for the TOE.
Verdict	Pass

5.2.7.3 FCS_COP.1/Hash Test/CAVP 1

Objective	The evaluator shall verify the implementation of hashing supported by the TOE.
Evaluator Findings	CAVP SHS Certs: #A2574, #A2575, #A2576, #A2577. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.8 FCS_COP.1/KeyedHash

5.2.8.1 FCS_COP.1/KeyedHash TSS 1

Objective	The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used. Upon investigation, the evaluator found that the TSS states that: **The following table states the key lengths, hash functions, block sizes and output MAC lengths supported by the TOE.**

HMAC-SHA	-1	-256	-384	-512
Key Length	**160 bits**	**256 bits**	**384 bits**	**512 bits**

	<table><tr><th>Hash function</th><th>SHA-1</th><th>SHA-256</th><th>SHA-384</th><th>SHA-512</th></tr><tr><td>Block Size</td><td>512 bits</td><td>512 bits</td><td>1024 bits</td><td>1024 bits</td></tr><tr><td>Output MAC</td><td>160 bits</td><td>256 bits</td><td>384 bits</td><td>512 bits</td></tr></table> Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.8.2 FCS_COP.1/KeyedHash Guidance 1

Objective	The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.
Evaluator Findings	The evaluator examined the section titled **Configuring SSH on the Evaluated Configuration for NDcPP** in the AGD to verify how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function. Upon investigation, the evaluator found that the AGD states the CLI command used to configure to use the appropriate HMAC function and associated functions. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.8.3 FCS_COP.1/KeyedHash Test/CAVP 1

Objective	The evaluator shall verify the implementation of MACing supported by the TOE.
Evaluator Findings	CAVP HMAC Certs: #A2574, #A2575, #A2576, #A2577. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.9 FCS_RBG_EXT.1

5.2.9.1 FCS_RBG_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min-entropy contained in the combined seed value.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min-entropy contained in the combined seed value. Upon investigation, the evaluator found that the TSS states that: **Junos OS performs random bit generation in accordance with NIST Special Publication 800-90 using HMAC_DRBG, SHA-256. The RBG for the NFX does not require any configuration and is seeded with 256 bits of entropy from hardware-based noise sources of entropy,**

	RANDOM_INTERRUPT, RANDOM_ATTACH, as well as a software-based noise source, namely RANDOM_NET_ETHER, RANDOM_SWI, RANDOM_FS_ATIME: • **RANDOM_INTERRUPT: This hardware source of entropy is provided by devices whose hardware interrupts are known to provide some amount of entropy, such as hard drive controllers. The timings are fed into kernel HMAC DRBG (Juniper kernel DRBG) along with a CPU cycle counter. This source of provides entropy during the boot-up process and during steady state operations.** • **RANDOM_NET_ETHER: This source of entropy is associated with network activity. Timings (CPU counter values at the time of the event) together with internal representation of network packets are used to construct extra entropy that is fed into Kernel HMAC DRBG. This source will only provide entropy after the device has booted and has started processing network packets.** • **RANDOM_SWI: This source of entropy is associated with software thread interrupts. Timing of software interrupts are combined with event and thread pointers to construct extra entropy that is fed into the Kernel HMAC DRBG. This source of provides entropy during the boot-up process and during steady state operations.** • **RANDOM_FS_ATIME: This source of entropy is associated with temporary file storage. An internal representation of the file system node of a file in temporary storage is hashed and used to construct entropy that is fed into the Kernel HMAC DRBG. This source of provides entropy during the boot-up process and during steady state operations.** • **RANDOM_ATTACH: This source of entropy is associated with attaching devices. The timing delta for the time to attach a device is used to construct entropy that is fed into the Kernel HMAC DRBG. This source of entropy provides entropy only during the boot-up process.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.9.2 FCS_RBG_EXT.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.
Evaluator Findings	The evaluator examined the section titled **Understanding FIPS Self-Tests** in the AGD to verify that it contains appropriate instructions for configuring the RNG functionality. Upon investigation, the evaluator found that the AGD states that **DRBG does not require any configuration, and initialized on startup**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.2.9.3 FCS_RBG_EXT.1.1 Test/CAVP 1

Objective	The evaluator shall verify the implementation of SP 800-90A DRBG supported by the TOE.
Evaluator Findings	CAVP DRBG Certs: #A2574, #A2576, #A2577. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3 TSS and Guidance Activities (IPsec)

5.3.1 FCS_IPSEC_EXT.1

5.3.1.1 FCS_IPSEC_EXT.1.1 TSS 1

Objective	The evaluator shall examine the TSS and determine that it describes what takes place when a packet is processed by the TOE, e.g., the algorithm used to process the packet. The TSS describes how the SPD is implemented and the rules for processing both inbound and outbound packets in terms of the IPsec policy. The TSS describes the rules that are available and the resulting actions available after matching a rule. The TSS describes how those rules and actions form the SPD in terms of the BYPASS (e.g., no encryption), DISCARD (e.g., drop the packet), and PROTECT (e.g., encrypt the packet) actions defined in RFC 4301.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes what takes place when a packet is processed by the TOE. Upon investigation, the evaluator found that the TSS states that: **Each packet is compared against entries in the security policy ruleset in sequential order until one is found that matches the specification in the policy, or until the end of the rule set is reached, in which case the implicit default policy is implemented and the packet is discarded. When a packet is processed by the TOE, the route is checked to see if it meets a defined security policy. If the packet meets the security policy, it is processed according to the rules of that policy.** **The following modes can be defined for a security flow policy:** • **Bypass – The Permit option directs traffic traversing the device through the stateful firewall inspection, but not through the IPsec VPN tunnel.** • **Discard – The Deny option inspects and drops all packets that do not match any Permit policies.** • **Protect – The traffic is routed through an IPsec tunnel based on a combination of route lookup and Permit policy inspection.** • **Log – This option logs traffic and session information for all modes mentioned above.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.2 FCS_IPSEC_EXT.1.1 TSS 2

Objective	As noted in section 4.4.1 of RFC 4301, the processing of entries in the SPD is non-trivial and the evaluator shall determine that the description in the TSS is sufficient to determine which rules will be applied given the rule structure implemented by the TOE. For example, if the TOE allows specification of ranges, conditional rules, etc., the evaluator shall determine that the description of rule processing (for both inbound and outbound packets) is sufficient to determine the action that will be applied, especially in the case where two different rules may apply. This description shall cover both the initial packets (that is, no SA is established on the interface or for that particular packet) as well as packets that are part of an established SA.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS is sufficient to determine which rules will be applied given the rule structure implemented by the TOE. Upon investigation, the evaluator found that the TSS states that:

	Each packet is compared against entries in the security policy ruleset in sequential order until one is found that matches the specification in the policy, or until the end of the rule set is reached, in which case the implicit default policy is implemented and the packet is discarded. When a packet is processed by the TOE, the route is checked to see if it meets a defined security policy. If the packet meets the security policy, it is processed according to the rules of that policy. **For inbound traffic, the TOE looks up the SA by using the destination IP address, security protocol, and security parameter index (SPI) value. For outbound VPN traffic, the policy invokes the SA associated with the VPN tunnel. If a packet arrives and there is not an active SA for that tunnel, the packet is dropped. The TOE will then begin to establish a tunnel, so that when the packet is resent, the SA is active. After the SA is established all subsequent packets in the session will use the IPsec tunnel.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.3 FCS_IPSEC_EXT.1.1 Guidance 1

Objective	The evaluator shall examine the guidance documentation to verify it instructs the Administrator how to construct entries into the SPD that specify a rule for processing a packet. The description includes all three cases – a rule that ensures packets are encrypted/decrypted, dropped, and flow through the TOE without being encrypted. The evaluator shall determine that the description in the guidance documentation is consistent with the description in the TSS, and that the level of detail in the guidance documentation is sufficient to allow the administrator to set up the SPD in an unambiguous fashion. This includes a discussion of how ordering of rules impacts the processing of an IP packet.
Evaluator Findings	The evaluator examined the section titled **Configuring Security Flow Policies** in the AGD to verify that it instructs the Administrator how to construct entries into the SPD that specify a rule for processing a packet. Upon investigation, the evaluator found that the AGD states the CLI commands to add entries into the SPD and to specify rules for processing a packet under three modes i.e. bypass, discard and protect. The evaluator also found that the description in the guidance documentation is consistent with the description in the TSS Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.4 FCS_IPSEC_EXT.1.3 TSS 1

Objective	The evaluator checks the TSS to ensure it states that the VPN can be established to operate in transport mode and/or tunnel mode (as identified in FCS_IPSEC_EXT.1.3).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS states that the VPN can be established to operate in transport mode and/or tunnel mode (as identified in FCS_IPSEC_EXT.1.3). Upon investigation, the evaluator found that the TSS states that: **The TOE supports tunnel mode only**. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.3.1.5 FCS_IPSEC_EXT.1.3 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains instructions on how to configure the connection in each mode selected.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it contains instructions on how to configure the connection in each mode selected. The ST mentions that the TOE only supports tunnel mode. Upon investigation, the evaluator found that the AGD states that **TOE only operates in tunnel mode by default. No separate configuration is needed for IPSec**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.6 FCS_IPSEC_EXT.1.4 TSS 1

Objective	The evaluator shall examine the TSS to verify that the selected algorithms are implemented. In addition, the evaluator ensures that the SHA-based HMAC algorithm conforms to the algorithms specified in FCS_COP.1/KeyedHash Cryptographic Operations (for keyed-hash message authentication) and if the SHA-based HMAC function truncated output is utilized it must also be described.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS states that the selected algorithms are implemented. Upon investigation, the evaluator found that the TSS states that: **The TOE supports AES-GCM-128, and AES-GCM-256, and AES-CBC-128, AES-CBC-192 or AES-CBC-256 using HMAC SHA-256 for ESP protection.** **IKEv1 and IKEv2 are implemented. IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109 and RFC 4868 for hash functions; IKEv2 as defined in RFCs 5996 (with no support for NAT traversal) and RFC 4868 for hash functions. IKEv1 aggressive mode is not supported.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.7 FCS_IPSEC_EXT.1.4 Guidance 1

Objective	The evaluator checks the guidance documentation to ensure it provides instructions on how to configure the TOE to use the algorithms selected.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it provides instructions on how to configure the TOE to use the algorithms selected. Upon investigation, the evaluator found that the AGD states the CLI commands to configure the TOE to use the selected algorithms as part of the IPsec proposal. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.8 FCS_IPSEC_EXT.1.5 TSS 1

Objective	The evaluator shall examine the TSS to verify that IKEv1 and/or IKEv2 are implemented.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies whether IKEv1 and/or IKEv2 are implemented. Upon investigation, the evaluator found that the TSS states that: **IKEv1 and IKEv2 are implemented. IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109 and RFC 4868 for hash functions; IKEv2 as defined in RFCs 5996 (with no support for NAT traversal) and RFC 4868 for hash functions. IKEv1 aggressive mode is not supported.** **The TOE supports AES-CBC-128, AES-CBC-192, and AES-CBC-256 for payload protection in IKEv1 and IKEv2. The TOE also supports AES-GCM-128 and AES-GCM-256 for the payload protection in IKEv2.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.9 FCS_IPSEC_EXT.1.5 TSS 2

Objective	For IKEv1 implementations, the evaluator shall examine the TSS to ensure that, in the description of the IPsec protocol, it states that aggressive mode is not used for IKEv1 Phase 1 exchanges, and that only main mode is used. It may be that this is a configurable option.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS states that aggressive mode is not used for IKEv1 Phase 1 exchanges, and that only main mode is used. Upon investigation, the evaluator found that the TSS states that: **IKEv1 aggressive mode is not supported.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.10 FCS_IPSEC_EXT.1.5. Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure it instructs the administrator how to configure the TOE to use IKEv1 and/or IKEv2 (as selected), and how to configure the TOE to perform NAT traversal (if selected).
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it instructs the administrator how to configure the TOE to use IKEv1 and/or IKEv2 (as selected), and how to configure the TOE to perform NAT traversal (if selected). Upon investigation, the evaluator found that the AGD states the CLI command used to configure the TOE to use IKEv1/IKEv2 as part of the IKE gateway configuration. The ST does not select NAT traversal, so no configuration instructions for NAT traversal are necessary. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.11 FCS_IPSEC_EXT.1.5. Guidance 2

Objective	If the IKEv1 Phase 1 mode requires configuration of the TOE prior to its operation, the evaluator shall check the guidance documentation to ensure that instructions for this configuration are contained within that guidance.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it contains any necessary instructions for IKEv1 Phase 1 mode configuration. Upon

	investigation, the evaluator found that the AGD states the CLI commands needed for IKEv1 Phase 1 configuration. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.12 FCS_IPSEC_EXT.1.6 TSS 1

Objective	The evaluator shall ensure the TSS identifies the algorithms used for encrypting the IKEv1 and/or IKEv2 payload, and that the algorithms chosen in the selection of the requirement are included in the TSS discussion.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the algorithms used for encrypting the IKEv1 and/or IKEv2 payload, and that the algorithms chosen in the selection of the requirement are included in the TSS discussion. Upon investigation, the evaluator found that the TSS states that: **IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109 and RFC 4868 for hash functions; IKEv2 as defined in RFCs 5996 (with no support for NAT traversal) and RFC 4868 for hash functions. IKEv1 aggressive mode is not supported.** **The TOE supports AES-CBC-128, AES-CBC-192, and AES-CBC-256 for payload protection in IKEv1 and IKEv2. The TOE also supports AES-GCM-128 and AES-GCM-256 for the payload protection in IKEv2.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.13 FCS_IPSEC_EXT.1.6 Guidance 1

Objective	The evaluator ensures that the guidance documentation describes the configuration of all selected algorithms in the requirement.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it describes the configuration of all selected algorithms in the requirement. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the TOE to use the selected algorithms as part of the IKE proposal. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.14 FCS_IPSEC_EXT.1.7 TSS 1

Objective	The evaluator shall ensure the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 1 SA lifetime and/or the IKEv2 SA lifetime. The evaluator shall verify that the selection made here corresponds to the selection in FCS_IPSEC_EXT.1.5.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 1 SA lifetime and/or the IKEv2 SA lifetime and that information corresponds to the selection in FCS_IPSEC_EXT.1.5. Upon investigation, the evaluator found that the TSS states that: **In the evaluated configuration, the TOE permits configuration of the:**

	• **IKEv1 Phase 1 and IKEv2 SA lifetimes in terms of length of time (180 to 86400 seconds),** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.15 FCS_IPSEC_EXT.1.7 Guidance 1 **[TD0633]**

Objective	The evaluator shall verify that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before the maximum SA lifetime of 24 hours is exceeded (e.g. configure a time value of 23h 45min to ensure the actual rekey is performed no later than 24h). The evaluator shall verify that the guidance documentation allows the Administrator to configure the Phase 1 SA value of 24 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 24 hours. It is not permitted to configure a value of 24 hours if that leads to an actual rekey after more than 24hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.
Evaluator Findings	The evaluator examined the section titled **Configuring VPN on a Device Running Junos OS** in the AGD to verify that it includes instructions for configuring values for SA lifetimes. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure time-based SA lifetime for Phase 1 in accordance with the ST selection. Phase 1 SA lifetime based on number of bytes is not supported as per the ST. The evaluator also verified that it was possible to configure Phase 1 SA lifetime value for 24 hours as per the requirement. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.16 FCS_IPSEC_EXT.1.8 TSS 1

Objective	The evaluator shall ensure the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 2 SA lifetime and/or the IKEv2 Child SA lifetime. The evaluator shall verify that the selection made here corresponds to the selection in FCS_IPSEC_EXT.1.5.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 2 SA lifetime and/or the IKEv2 Child SA lifetime and that the information corresponds to the selection in FCS_IPSEC_EXT.1.5. Upon investigation, the evaluator found that the TSS states that: **In the evaluated configuration, the TOE permits configuration of the:** • **IKEv1 Phase 2 SA and IKEv2 Child SA lifetimes in terms of (kilo)bytes (64 to 1GB) lifetime or length of time (180 to 28,800 seconds)** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.17 FCS_IPSEC_EXT.1.8 Guidance 1 **[TD0633]**

Objective	The evaluator shall verify that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before the maximum SA lifetime of 8 hours is exceeded (e.g. configure a time value of 7h 45min to ensure the actual rekey is performed no later than 8h). The evaluator shall verify that the guidance documentation allows the Administrator to configure the Phase 2 SA value of 8 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 8 hours. It is not permitted to configure a value of 8 hours if that leads to an actual rekey after more than 8hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.
Evaluator Findings	The evaluator examined the section titled **Configuring VPN on a Device Running Junos OS** in the AGD to verify that it includes instructions for configuring values for SA lifetimes. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure time as well as traffic-based SA lifetime for phase 2 in accordance with the ST selections. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.18 FCS_IPSEC_EXT.1.9 TSS 1

Objective	The evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating "x". The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of "x" meets the stipulations in the requirement.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the process for generating "x" for each DH group supported. Upon investigation, the evaluator found that the TSS states that: **The TOE supports Diffie-Hellman Groups 14, 19, and 20. In the IKEv1 phase 1 and phase 2 exchanges, the TOE and peer will agree on the best DH group both can support. When the TOE receives an IKE proposal, it will select the first DH group that matches the acceptable DH groups configured in the TOE (one or more of DH Groups 14, 19, or 20) and the negotiation will fail if there is no match. Similarly, when the peer initiates the IKE protocol, the TOE will select the first match from the IKE proposal sent by the peer and the negotiation fails is no acceptable match is found.** **The TOE uses HMAC DRBG with SHA-256 for the generation of DH exponents and nonces in the IKE key exchange protocol of length 224 bits (for DH Group 14).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.19 FCS_IPSEC_EXT.1.10 TSS 1

Objective	If the first selection is chosen, the evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating each nonce. The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.

	If the second selection is chosen, the evaluator shall check to ensure that, for each PRF hash supported, the TSS describes the process for generating each nonce. The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the process for generating each nonce for each DH group or PRF hash supported and indicates that the random number generated that meets the requirements in this PP is used, and indicates that the length of the nonces meet the stipulations in the requirement. Upon investigation, the evaluator found that the TSS states that: **The TOE supports Diffie-Hellman Groups 14, 19, and 20. In the IKEv1 phase 1 and phase 2 exchanges, the TOE and peer will agree on the best DH group both can support. When the TOE receives an IKE proposal, it will select the first DH group that matches the acceptable DH groups configured in the TOE (one or more of DH Groups 14, 19, or 20) and the negotiation will fail if there is no match. Similarly, when the peer initiates the IKE protocol, the TOE will select the first match from the IKE proposal sent by the peer and the negotiation fails is no acceptable match is found.** **The TOE uses HMAC DRBG with SHA-256 for the generation of DH exponents and nonces in the IKE key exchange protocol of length 224 bits (for DH Group 14).** **The TOE checks the strengths of the configured IKE algorithms prior to committing a tunnel configuration to ensures that the strength of the symmetric algorithm (128, 192 or 256 bits) negotiated to protect the IKEv1 Phase 1, IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm negotiated to protect the IKEv1 Phase 2, IKEv2 CHILD_SA connection. If the strength is not greater an error is displayed, and the configuration fails.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.20 FCS_IPSEC_EXT.1.11 TSS 1

Objective	The evaluator shall check to ensure that the DH groups specified in the requirement are listed as being supported in the TSS. If there is more than one DH group supported, the evaluator checks to ensure the TSS describes how a particular DH group is specified/negotiated with a peer.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS lists the DH groups specified in the requirement as being supported. Upon investigation, the evaluator found that the TSS states that: **The TOE supports Diffie-Hellman Groups 14, 19, and 20. In the IKEv1 phase 1 and phase 2 exchanges, the TOE and peer will agree on the best DH group both can support.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.21 FCS_IPSEC_EXT.1.11 Guidance 1

Objective	The evaluator ensures that the guidance documentation describes the configuration of all algorithms selected in the requirement.

Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it describes the configuration of all algorithms selected in the requirement. Upon investigation, the evaluator found that the AGD states the CLI command needed to configure the selected DH Groups. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.22 FCS_IPSEC_EXT.1.12 TSS 1

Objective	The evaluator shall check that the TSS describes the potential strengths (in terms of the number of bits in the symmetric key) of the algorithms that are allowed for the IKE and ESP exchanges. The TSS shall also describe the checks that are done when negotiating IKEv1 Phase 2 and/or IKEv2 CHILD_SA suites to ensure that the strength (in terms of the number of bits of key in the symmetric algorithm) of the negotiated algorithm is less than or equal to that of the IKE SA this is protecting the negotiation.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the potential strengths of the algorithms that are allowed for the IKE and ESP exchanges and the checks that are done when negotiating IKEv1 Phase 2 and/or IKEv2 CHILD_SA suites. Upon investigation, the evaluator found that the TSS states that: **The TOE checks the strengths of the configured IKE algorithms prior to committing a tunnel configuration to ensures that the strength of the symmetric algorithm (128, 192 or 256 bits) negotiated to protect the IKEv1 Phase 1, IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm negotiated to protect the IKEv1 Phase 2, IKEv2 CHILD_SA connection. If the strength is not greater an error is displayed, and the configuration fails.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.23 FCS_IPSEC_EXT.1.13 TSS 1

Objective	The evaluator ensures that the TSS identifies RSA and/or ECDSA as being used to perform peer authentication. The description must be consistent with the algorithms as specified in FCS_COP.1/SigGen Cryptographic Operations (for cryptographic signature).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies RSA and/or ECDSA as being used to perform peer authentication and that the algorithms are consistent with those specified in FCS_COP.1/SigGen Cryptographic Operations. Upon investigation, the evaluator found that the TSS states that: **The TOE supports both RSA and ECDSA for use with X.509v3 certificates that conform to RFC 4945 and pre-shared Keys for IPsec support.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.24 FCS_IPSEC_EXT.1.13 TSS 2

Objective	If pre-shared keys are chosen in the selection, the evaluator shall check to ensure that the TSS describes how pre-shared keys are established and used in authentication of IPsec

	connections. The description in the TSS shall also indicate how pre-shared key establishment is accomplished for TOEs that can generate a pre-shared key as well as TOEs that simply use a pre-shared key.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how pre-shared keys are established and used in authentication of IPsec connections. Upon investigation, the evaluator found that the TSS states that: **The TOE accepts Unicode characters to specify text-based pre-shared keys. Unicode characters are encoded as UTF-8 and treated as multiple bytes – up to 4 bytes depending on the character. The maximum length limit for text-based pre-shared keys enforced by the TOE is 255 bytes. When a pre-shared key is only composed of ASCII characters this limit is equivalent to 255 characters. The text-based pre-shared or bit-based keys may contain upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". The TOE accepts pre-shared text keys and converts the text string into an authentication value as per RFC 2409 for IKEv1 or RFC 4306 for IKEv2, using the PRF that is configured as the hash algorithm for the IKE exchanges** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.25 FCS_IPSEC_EXT.1.13 Guidance 1

Objective	The evaluator ensures the guidance documentation describes how to set up the TOE to use certificates with RSA and/or ECDSA signatures and public keys.
Evaluator Findings	The evaluator examined the sections titled **Configuring an IPsec VPN with an RSA Signature for IKE Authentication** and **Configuring an IPsec VPN with an ECDSA Signature for IKE Authentication** in the AGD to verify that it describes how to set up the TOE to use certificates with RSA and/or ECDSA signatures and public keys. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the TOE to use certificates with RSA or ECDSA signatures and public keys. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.26 FCS_IPSEC_EXT.1.13 Guidance 2

Objective	The evaluator shall check that the guidance documentation describes how preshared keys are to be generated and established. The description in the guidance documentation shall also indicate how pre-shared key establishment is accomplished for TOEs that can generate a pre-shared key as well as TOEs that simply use a pre-shared key.
Evaluator Findings	The evaluator examined the section titled **Configuring an IPsec VPN with a Preshared Key for IKE Authentication** in the AGD to verify that it describes how pre-shared keys are to be generated and established. Upon investigation, the evaluator found that the AGD states the CLI commands to configure pre-shared key establishment. The TOE simply uses pre-shared keys but does not generate them so no configuration steps for pre-shared key generation are required. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.27 FCS_IPSEC_EXT.1.13 Guidance 3

Objective	The evaluator will ensure that the guidance documentation describes how to configure the TOE to connect to a trusted CA and ensure a valid certificate for that CA is loaded into the TOE and marked "trusted".
Evaluator Findings	The evaluator examined the sections titled **Configuring an IPsec VPN with an RSA Signature for IKE Authentication** and **Configuring an IPsec VPN with an ECDSA Signature for IKE Authentication** in the AGD to verify that it describes how to configure the TOE to connect to a trusted CA and ensure a valid certificate for that CA is loaded into the TOE and marked "trusted". Upon investigation, the evaluator found that the AGD states the CLI commands needed to load CA certificates into the TOE and verify their validity. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3.1.28 FCS_IPSEC_EXT.1.14 TSS 1

Objective	The evaluator shall ensure that the TSS describes how the TOE compares the peer's presented identifier to the reference identifier. This description shall include which field(s) of the certificate are used as the presented identifier (DN, Common Name, or SAN). If the TOE simultaneously supports the same identifier type in the CN and SAN, the TSS shall describe how the TOE prioritizes the comparisons (e.g. the result of comparison if CN matches but SAN does not). If the location (e.g. CN or SAN) of non-DN identifier types must explicitly be configured as part of the reference identifier, the TSS shall state this. If the ST author assigned an additional identifier type, the TSS description shall also include a description of that type and the method by which that type is compared to the peer's presented certificate, including what field(s) are compared and which fields take precedence in the comparison.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE compares the peer's presented identifier to the reference identifier. Upon investigation, the evaluator found that the TSS states that: **The TOE permits the identity to be expressed as email address, fully qualified domain name or IP address. When configuring an IKE policy, the certificate name must be set so the TOE knows which certificate to use for authentication. If either the certificate does not validate, or the contents do not match the configured identity, then the SA will not be established. When configuring the IKE identity of the remote endpoint the administrator must specify an email address, fully qualified domain name, or IP address that will be matched against the SAN field, or a distinguished name, in the presented certificate.** **For public key-based authentication of IPsec connections, Junos OS validates the X.509 certificates by extracting the subject, issuer, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer CA is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. Junos OS verifies the validity of the signature** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.3.1.29 FCS_IPSEC_EXT.1.14 Guidance 1

Objective	The evaluator shall ensure that the operational guidance describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by the TOE does not guarantee unique identifiers, the evaluator shall ensure that the operational guidance provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not, and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). Upon investigation, the evaluator found that the AGD states **Configure the remote-identity statement at the set security ike gateway gateway-name hierarchy level to match the IKE ID that is received from the peer. The IKE ID values can be an IPv4 address or an IPv6 address, email id, FQDN, or a distinguished name.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.4 TSS and Guidance Activities (SSH)

5.4.1 FCS_SSHS_EXT.1

5.4.1.1 FCS_SSHS_EXT.1.2 TSS 1

Objective	The evaluator shall check to ensure that the TSS contains a list of supported public key algorithms that are accepted for client authentication and that this list is consistent with signature verification algorithms selected in FCS_COP.1/SigGen (e.g., accepting EC keys requires corresponding Elliptic Curve Digital Signature algorithm claims).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS contains a description of the public key algorithms that are acceptable for use for authentication, that this list conforms to FCS_SSHS_EXT.1.5, and that if password-based authentication methods have been selected in the ST then these are also described. Upon investigation, the evaluator found that the TSS states that **The TOE supports public key authentication for SSHv2 session authentication using the following algorithms: ecdsa-sha2-nistp256, ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.2 FCS_SSHS_EXT.1.2 TSS 2 **[TD0631]**

Objective	The evaluator shall confirm that the TSS includes the description of how the TOE establishes a user identity when an SSH client presents a public key or X.509v3 certificate. For example, the TOE could verify that the SSH client's presented public key matches one that is stored within the SSH server's authorized_keys file.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE establishes a user identity when an SSH client presents a public key or X.509v3 certificate. Upon investigation, the evaluator found that the TSS states that:

	Public Key Authentication Method: The TOE supports public key authentication for SSHv2 session authentication. Authentication succeeds if the correct private key is used. The TOE does not require multiple authentications (public key and password) for users. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.3 FCS_SSHS_EXT.1.2 TSS 3 **[TD0631]**

Objective	If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, then the evaluator shall confirm its role in the authentication process is described in the TSS.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS contains a description of the the role of password-based authentication in the authentication process. Upon investigation, the evaluator found that the TSS states that **The TOE supports password authentication. Expired passwords are not supported and cannot be used for authentication**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.4 FCS_SSHS_EXT.1.3 TSS 1

Objective	The evaluator shall check that the TSS describes how “large packets” in terms of RFC 4253 are detected and handled.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how “large packets” in terms of RFC 4253 are detected and handled. Upon investigation, the evaluator found that the TSS states that: **Maximum Packet length: The TOE reads the packet payload size in TCP packets to determine packet length. Packets greater than 256K bytes in an SSH transport connection are dropped and the connection is terminated by Junos OS.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.5 FCS_SSHS_EXT.1.4 TSS 1

Objective	The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that optional characteristics are specified, and the encryption algorithms supported are specified as well. The evaluator shall check the TSS to ensure that the encryption algorithms specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the optional characteristics and the encryption algorithms supported. Upon investigation, the evaluator found that the TSS states that: **Encryption: The TOE offers the following for encryption of SSH sessions: aes128-cbc and aes256-cbc, aes128-ctr, aes256-ctr. The TOE permits negotiation of encryption algorithms in each direction. The TOE does not allow the “none” algorithm for encryption** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.4.1.6 FCS_SSHS_EXT.1.4 Guidance 1

Objective	The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements).
Evaluator Findings	The evaluator examined the section titled **Configuring SSH and Console Connection** in the AGD to verify that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS. Upon investigation, the evaluator found that the AGD states that AGD describes the configuration of SSH on the TOE. Specifically, the evaluator found that AGD describes the following characteristics of SSH configuration, • System Login Messages and Announcements • Limiting User Login Attempts • Specifying Host-key Algorithms • Specifying key exchange Algorithms • Specifying ciphers allowed • Supported MAC algorithms The evaluator found that AGD describes the configuration of SSH from the CLI. Finally, the evaluator compared the configuration described in AGD to the TSS of ST. The evaluator found that the configuration options in AGD are consistent with the description of SSH in the TSS. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.7 FCS_SSHS_EXT.1.5 TSS 1 **[TD0631]**

Objective	The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that optional characteristics are specified, and the public key algorithms supported are specified as well. The evaluator shall check the TSS to ensure that the public key algorithms specified are identical to those listed for this component.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies the optional characteristics and the public key algorithms supported. Upon investigation, the evaluator found that the TSS states that **The TOE supports public key authentication for SSHv2 session authentication using the following algorithms: ecdsa-sha2-nistp256, ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.8 FCS_SSHS_EXT.1.5 Guidance 1

Objective	The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for

	instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements).
Evaluator Findings	The evaluator examined the section titled **Configuring SSH and Console Connection** in the AGD to verify that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS. Upon investigation, the evaluator found that the AGD states that AGD describes the configuration of SSH on the TOE. Specifically, the evaluator found that AGD describes the following characteristics of SSH configuration: • System Login Messages and Announcements • Limiting User Login Attempts • Specifying Host-key Algorithms • Specifying key exchange Algorithms • Specifying ciphers allowed • Supported MAC algorithms The evaluator found that AGD describes the configuration of SSH from the CLI. Finally, the evaluator compared the configuration described in AGD to the TSS of ST. The evaluator found that the configuration options in AGD are consistent with the description of SSH in the TSS. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.9 FCS_SSHS_EXT.1.6 TSS 1

Objective	The evaluator shall check the TSS to ensure that it lists the supported data integrity algorithms, and that the list corresponds to the list in this component.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** 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: **RFC 4253:** **Data Integrity: The TOE permits negotiation of HMAC-SHA1 in each direction for SSH transport.** **RFC 6668:** **Data Integrity Algorithms: Both the recommended and optional algorithms hmac-sha2-256 and hmac-sha2-512 (respectively) are implemented for SSH transport.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.10 FCS_SSHS_EXT.1.6 Guidance 1

Objective	The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed data integrity algorithms are used in SSH connections with the TOE (specifically, that the “none” MAC algorithm is not allowed).

Evaluator Findings	The evaluator examined the section titled **Configuring SSH on the Evaluated Configuration for NDcPP** in the AGD to verify that it contains instructions to the administrator on how to ensure that only the allowed data integrity algorithms are used in SSH connections with the TOE. Further, it was verified that 'none' is not a permissible MAC algorithm. **Specify all the permissible message authentication code algorithms for SSHv2** **[edit]** **user@host#set system services ssh macs hmac-sha1** **user@host#set system services ssh macs hmac-sha2-256** **user@host#set system services ssh macs hmac-sha2-512** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.11 FCS_SSHS_EXT.1.7 TSS 1

Objective	The evaluator shall check the TSS to ensure that it lists the supported key exchange algorithms, and that the list corresponds to the list in this component.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS lists the supported key exchange algorithms, and that that list corresponds to the list in this component. Upon investigation, the evaluator found that the TSS states that: **Ordering of Key Exchange Methods: Key exchange is performed only using one of the supported key exchange algorithms, which are ordered as follows: ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521 (specified in RFC 5656), diffie-hellman-group14-sha1 (specified in RFC 4253).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.12 FCS_SSHS_EXT.1.7 Guidance 1

Objective	The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed key exchange algorithms are used in SSH connections with the TOE.
Evaluator Findings	The evaluator examined the section titled **Configuring SSH on the Evaluated Configuration for NDcPP** in the AGD to verify that it contains instructions to the administrator on how to ensure that only the allowed key exchange algorithms are used in SSH connections with the TOE. Upon investigation, the evaluator found that the AGD states that: **Specify the SSH key-exchange for Diffie-Hellman keys for the system services.** **[edit] user@host#set system services ssh key-exchange dh-group14-sha1** **user@host#set system services ssh key-exchange ecdh-sha2-nistp256** **user@host#set system services ssh key-exchange ecdh-sha2-nistp384** **user@host#set system services ssh key-exchange ecdh-sha2-nistp521**

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

5.4.1.13 FCS_SSHS_EXT.1.8 TSS 1

Objective	The evaluator shall check that the TSS specifies the following: a) Both thresholds are checked by the TOE. b) Rekeying is performed upon reaching the threshold that is hit first.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS specifies that both thresholds are checked and that rekeying is performed upon reaching the threshold that is hit first. Upon investigation, the evaluator found that the TSS states that **For ciphers whose blocksize >= 16, the TOE rekeys every (2^32-1) bytes. The client may explicitly request a rekeying event as a valid SSHv2message at any time and the TOE will honor this request. Re-keying of SSH session keys can be configured using the sshd_config knob. The data-limit must be set between 51200 and 1Gbyte and the time-limit must be set within 1 and 60 minutes. The TOE will rekey based on whichever limit is reached first.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.4.1.14 FCS_SSHS_EXT.1.8 Guidance 1

Objective	If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, then the evaluator shall check that the guidance documentation describes how to configure those thresholds. Either the allowed values are specified in the guidance documentation and must not exceed the limits specified in the SFR (one hour of session time, one gigabyte of transmitted traffic) or the TOE must not accept values beyond the limits specified in the SFR. The evaluator shall check that the guidance documentation describes that the TOE reacts to the first threshold reached.
Evaluator Findings	The evaluator examined the section titled **Configuring SSH on the Evaluated Configuration for NDcPP** in the AGD to verify that it describes how to configure any thresholds that are configurable. Upon investigation, the evaluator found that the AGD states that the AGD specifically addresses the configuration of the rekey limits for TOE SSH connections. The AGD identifies the method of configuring either time-limit or data-limit values via the CLI. The evaluator found provided instructions include the specific configurations required to ensure only approved limits are used in SSH connection with the TOE. This was confirmed by comparing the instructions in AGD to the description of rekey limits found in the TSS of ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.5 TSS and Guidance Activities (User Data Protection)

5.5.1 FDP_RIP.2

5.5.1.1 FDP_RIP.2 TSS 1

Objective	"Resources" in the context of this requirement are network packets being sent through (as opposed to "to", as is the case when a security administrator connects to the TOE) the TOE. The concern is that once a network packet is sent, the buffer or memory area used by the packet still contains data from that packet, and that if that buffer is re-used, those data might remain and make their way into a new packet. The evaluator shall check to ensure that the TSS describes packet processing to the extent that they can determine that no data will be reused when processing network packets. The evaluator shall ensure that this description at a minimum describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes packet processing to the extent that they can determine that no data will be reused when processing network packets. The evaluator shall ensure that this description at a minimum describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs. Upon investigation, the evaluator found that the TSS states that: **The only resource made available to information flowing through a TOE is the temporary storage of packet information when access is requested and when information is being routed. User data is not persistent when resources are released by one user/process and allocated to another user/process. Temporary storage (memory) used to build network packets is overwritten with zeros (making the previous data unavailable or zeroized) when the resource is called into use by the next user/process. Junos knows, and keeps track of, the length of the packet. This means that when memory allocated from a previous user/process arrives to build the next network packet, Junos is aware of when the end of the packet is reached and pads a short packet with zeros accordingly. Therefore, no residual information from packets in a previous information stream can traverse through the TOE** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6 TSS and Guidance Activities (Firewall)

5.6.1 FFW_RUL_EXT.1

5.6.1.1 FFW_RUL_EXT.1 TSS

Objective	The evaluator shall verify that the TSS provides a description of the TOE's initialization/startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process. The evaluator shall verify that the TSS also include a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describe the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure. This could include the failure of a component, such as a process being terminated, or a failure within a component, such as

	memory buffers full and cannot process packets. The description shall also include a description how the TOE behaves in the situation where the traffic exceeds the amount of traffic the TOE can handle and how it is ensured that also in this condition stateful traffic filtering rules are still applied so that traffic does not pass that shouldn't pass according to the specified rules.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS provides a description of the TOE's initialization/startup process, provides a discussion that supports the assertion that packets cannot flow during this process, includes a narrative that identifies the components involved in processing the network packets, describes the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure and describes how the TOE behaves in the situation where the traffic exceeds the amount of traffic the TOE can handle. Upon investigation, the evaluator found that the TSS states that: **. The following steps list the boot sequence for the TOE:** • **BIOS hardware and memory checks** • **Loading and initialization of the FreeBSD Kernel OS** • **FIPS self-tests and firmware integrity tests are executed** • **The init utility is started (mounts file systems, sets up network cards to communicate on the network, and generally starts all the processes that usually are run on a FreeBSD system at startup)** • **Daemon programs such as Internet Service Daemon (INETD), Routing Protocol Daemon (RPD), Syslogd are started; Routing and forwarding tables are initialized** • **Management Daemon (or MGD) is loaded, allowing access to management interface** • **Physical interfaces are active** **Once the interfaces are brought up, they will start to receive and send packets based on the current configuration (or not receive or send any packets if they have not been previously configured).** **The Information Flow subsystem is responsible for processing the arriving packets from the network to the TOE's network interface. Based on Administrator-configured policy, interface and zone information, the packet flows through the various modules of the Information Flow subsystem. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces. By default, the TOE behavior is to deny packets when there is no rule match unless another required condition allows the network traffic If a security risk is found in the packet. e.g. denial-of-service attacks, the packet is dropped and an event is logged. The packet does not continue to the next module for processing. If the packet is not dropped by a given module, the interrupt handling routine calls the function for the next relevant module.** **The Information Flow subsystem consists of the following modules:** • **IP Classification Module** • **Attack Detection Module** • **Session Lookup Module** • **Security Policy Module** • **Session Setup Module**

	• **Inetd Module** • **Rdp Module** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.2 FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 TSS

Objective	The evaluator shall verify that the TSS describes a stateful packet filtering policy and the following attributes are identified as being configurable within stateful traffic filtering rules for the associated protocols:
	• ICMPv4
	  ○ Type
	  ○ Code
	• ICMPv6
	  ○ Type
	  ○ Code
	• IPv4
	  ○ Source address
	  ○ Destination Address
	  ○ Transport Layer Protocol
	• IPv6
	  ○ Source address
	  ○ Destination Address
	  ○ Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
	• TCP
	  ○ Source Port
	  ○ Destination Port
	• UDP
	  ○ Source Port
	  ○ Destination Port
	The evaluator shall verify that each rule can identify the following actions: permit or drop with the option to log the operation. The evaluator shall verify that the TSS identifies all interface types subject to the stateful packet filtering policy and explains how rules are associated with distinct network interfaces.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes a stateful packet filtering policy and the attributes listed above are identified as being configurable within stateful traffic filtering rules for the associated protocols. Upon investigation, the evaluator found that the TSS states that: **The TOE performs stateful network traffic filtering on network packets using the following network traffic protocols and network fields conforming to the described RFCs:** • **RFC 792 ICMPv4: Type, Code** • **RFC 4443 ICMPv6: Type, Code** • **RFC 791 (IPv4): Source address, Destination Address, Transport Layer Protocol** • **RFC 2460 (IPv6): Source address, Destination Address, Transport Layer Protocol** • **RFC 793 (TCP): Source port, Destination port**

	• **RFC 768 (UDP): Source port, Destination port** **Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.** **The TOE shall allow permit, deny, and log operations to be associated with rules and these rules can be assigned to distinct network interfaces.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.3 FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Guidance

Objective	The evaluators shall verify that the guidance documentation identifies the following attributes as being configurable within stateful traffic filtering rules for the associated protocols:
	• ICMPv4
	  ○ Type
	  ○ Code
	• ICMPv6
	  ○ Type
	  ○ Code
	• IPv4
	  ○ Source address
	  ○ Destination Address
	  ○ Transport Layer Protocol
	• IPv6
	  ○ Source address
	  ○ Destination Address
	  ○ Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
	• TCP
	  ○ Source Port
	  ○ Destination Port
	• UDP
	  ○ Source Port
	  ○ Destination Port
	The evaluator shall verify that the guidance documentation indicates that each rule can identify the following actions: permit, drop, and log.
	The evaluator shall verify that the guidance documentation explains how rules are associated with distinct network interfaces.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the AGD to verify that it identifies the attributes listed above as being configurable within stateful traffic filtering rules for the associated protocols. Upon investigation, the evaluator found that the AGD states that: **The TOE performs stateful network traffic filtering on network packets using the following network traffic protocols and network fields conforming to the described RFCs:** • **RFC 792 ICMPv4: Type, Code** • **RFC 4443 ICMPv6: Type, Code**

	• **RFC 791 (IPv4): Source address, Destination Address, Transport Layer Protocol** • **RFC 2460 (IPv6): Source address, Destination Address, Transport Layer Protocol** • **RFC 793 (TCP): Source port, Destination port** • **RFC 768 (UDP): Source port, Destination port** **Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.** **The TOE shall allow permit, deny, and log operations to be associated with rules and these rules can be assigned to distinct network interfaces.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.4 FFW_RUL_EXT.1.5 TSS

Objective	The evaluator shall verify that the TSS identifies the protocols that support stateful session handling. The TSS shall identify TCP, UDP, and, if selected by the ST author, ICMP. The evaluator shall verify that the TSS describes how stateful sessions are established (including handshake processing) and maintained. The evaluator shall verify that for TCP, the TSS identifies and describes the use of the following attributes in session determination: source and destination addresses, source and destination ports, sequence number, and individual flags. The evaluator shall verify that for UDP, the TSS identifies and describes the following attributes in session determination: source and destination addresses, source and destination ports. The evaluator shall verify that for ICMP (if selected), the TSS identifies and describes the following attributes in session determination: source and destination addresses, other attributes chosen in FFW_RUL_EXT.1.5. The evaluator shall verify that the TSS describes how established stateful sessions are removed. The TSS shall describe how connections are removed for each protocol based on normal completion and/or timeout conditions. The TSS shall also indicate when session removal becomes effective (e.g., before the next packet that might match the session is processed).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the protocols that support stateful session handling, describes how stateful sessions are established and maintained; identifies and describes the use of TCP, UDP and ICMP (if selected) attributes in session determination; and describes how established stateful sessions are removed. Upon investigation, the evaluator found that the TSS states that: **The TOE accepts network packets if it matches an established TCP, UDP or ICMP session using:** • **TCP: source and destination addresses, source and destination ports, sequence number, flags** • **UDP: source and destination addresses, source and destination ports** • **ICMP: source and destination addresses, type, code**

	The TOE will remove existing traffic flows due to session inactivity timeout, or completion of the session. **The TOE supports FTP (RFC 959) to dynamically establish sessions allowing network traffic according to Administrator rules. Session events will be logged in accordance with 'log' operations defined in the rules. Source and destination addresses, source and destination ports, transport layer protocol, and TOE Interface are recorded in each log record.** **Junos implements what is referred to as an Application Layer gateway (ALG) that inspects FTP traffic to determine the port number used for data sessions. The ALG permits data traffic for the duration of the session, closing the port when the session ends.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.5 FFW_RUL_EXT.1.5 Guidance

Objective	The evaluator shall verify that the guidance documentation describes stateful session behaviors. For example, a TOE might not log packets that are permitted as part of an existing session.
Evaluator Findings	The evaluator examined the section titled **Understanding Protocol Support** in the AGD to verify that it describes stateful session behaviors. Upon investigation, the evaluator found that the AGD states the network traffic protocols and network fields used to perform stateful network traffic filtering on network packets . Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.6 FFW_RUL_EXT.1.6 TSS

Objective	The evaluator shall verify that the TSS identifies the following as packets that will be automatically dropped and are counted or logged: a) Packets which are invalid fragments, including a description of what constitutes an invalid fragment b) Fragments that cannot be completely re-assembled c) Packets where the source address is defined as being on a broadcast network d) Packets where the source address is defined as being on a multicast network e) Packets where the source address is defined as being a loopback address f) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4; g) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6; h) Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified i) Other packets defined in FFW_RUL_EXT.1.6 (if any)
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the items listed above as packets that will be automatically

	dropped and are counted or logged. Upon investigation, the evaluator found that the TSS states that:
	The TOE enforces the following default reject rules with logging on all network traffic:
	• **invalid fragments;**
	• **fragmented IP packets which cannot be re-assembled completely;**
	• **where the source address is equal to the address of the network interface where the network packet was received;**
	• **where the source address does not belong to the networks associated with the network interface where the network packet was received;**
	• **where the source address is defined as being on a broadcast network;**
	• **where the source address is defined as being on a multicast network;**
	• **where the source address is defined as being a loopback address;**
	• **where the source address is a multicast;**
	• **packets where the source or destination address is a link-local address;**
	• **where the source or destination address is defined as being an address "reserved for future use" as specified in RFC 5735 for IPv4;**
	• **where the source or destination address is defined as an "unspecified address" or an address "reserved for future definition and use" as specified in RFC 3513 for IPv6;**
	• **with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified;**
	• **packets are checked for validity. "Invalid fragments" are those that violate these rules:**
	  ○ **No overlap**
	  ○ **The total fragments in one packet should not be more than 62 pieces**
	  ○ **The total length of merged fragments should not larger than 64k**
	  ○ **All fragments in one packet should arrive in 2 seconds**
	  ○ **The total queued fragments has limitation, depending on the platform**
	  ○ **The total number of concurrent fragment processing for different packet has limitations depending on platform**
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.6.1.7 FFW_RUL_EXT.1.6 Guidance

Objective	The evaluator shall verify that the guidance documentation describes packets that are discarded and potentially logged by default. If applicable protocols are identified, their descriptions need to be consistent with the TSS. If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.
Evaluator Findings	The evaluator examined the section titled **Configuring Default Deny-All and Reject Rules** in the AGD to verify that it describes packets that are discarded and potentially logged by default. Upon investigation, the evaluator found that the AGD states the default reject rules and the CLI commands to configure the same with logging.

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

5.6.1.8 FFW_RUL_EXT.1.7 TSS

Objective	The evaluator shall verify that the TSS explains how the following traffic can be dropped and counted or logged: a) Packets where the source address is equal to the address of the network interface where the network packet was received b) Packets where the source or destination address of the network packet is a link-local address c) Packets where the source address does not belong to the networks associated with the network interface where the network packet was received, including a description of how the TOE determines whether a source address belongs to a network associated with a given network interface

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS explains how the required traffic can be dropped and counted or logged. Upon investigation, the evaluator found that the TSS states that:
	The TOE enforces the following default reject rules with logging on all network traffic:
	• **invalid fragments;**
	• **fragmented IP packets which cannot be re-assembled completely;**
	• **where the source address is equal to the address of the network interface where the network packet was received;**
	• **where the source address does not belong to the networks associated with the network interface where the network packet was received;**
	• **where the source address is defined as being on a broadcast network;**
	• **where the source address is defined as being on a multicast network;**
	• **where the source address is defined as being a loopback address;**
	• **where the source address is a multicast;**
	• **packets where the source or destination address is a link-local address;**
	• **where the source or destination address is defined as being an address "reserved for future use" as specified in RFC 5735 for IPv4;**
	• **where the source or destination address is defined as an "unspecified address" or an address "reserved for future definition and use" as specified in RFC 3513 for IPv6;**
	• **with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified;**
	• **packets are checked for validity. "Invalid fragments" are those that violate these rules:**
	  ○ **No overlap**
	  ○ **The total fragments in one packet should not be more than 62 pieces**

	○ **The total length of merged fragments should not larger than 64k** ○ **All fragments in one packet should arrive in 2 seconds** ○ **The total queued fragments has limitation, depending on the platform** ○ **The total number of concurrent fragment processing for different packet has limitations depending on platform** The TSS also states that **The TOE is configured to associate network interfaces to IP subnets. Source IP addresses are then associated with the network interface.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.9 FFW_RUL_EXT.1.7 Guidance

Objective	The evaluator shall verify that the guidance documentation describes how the TOE can be configured to implement the required rules. If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.
Evaluator Findings	The evaluator examined the sections titled **Configuring Traffic Filter Rules** and **Logging the Dropped Packets Using Default Deny-all Option** in the AGD to verify that it describes how the TOE can be configured to implement the required rules and, if logging is configurable, provides instructions to configure auditing of automatically rejected packets. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the TOE to implement required traffic filtering rules and to configure logging for automatically rejected packets. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.10 FFW_RUL_EXT.1.8 TSS 1

Objective	The evaluator shall verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset. Upon investigation, the evaluator found that the TSS states the following: **The Session Lookup module performs lookups in the session table which is used for all interfaces based on the information in incoming packets.** **The Security Policy module examines traffic passing through the TOE (via Session Setup module) and determines if the traffic can pass based on administrator-configured access policies.** **The RPD (Routing Protocol Daemon) module provides the implementations and algorithms for the routing protocols and route calculations.** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.6.1.11 FFW_RUL_EXT.1.8 TSS 2 **[TD0545]**

Objective	If the TOE implements a mechanism that ensures that no conflicting rules can be configured, the TSS shall describe the underlying mechanism.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target and determined that the TOE does not implement a mechanism that ensures that no conflicting rules can be configured. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.12 FFW_RUL_EXT.1.8 Guidance

Objective	The evaluator shall verify that the guidance documentation describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.
Evaluator Findings	The evaluator examined the section titled **Configuring Traffic Filtering Rules** in the AGD to verify that it describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing. Upon investigation, the evaluator found that the AGD states that **The firewall filter terms are evaluated in the order in which they are configured** and also provides instructions for the administrator to configure the order of rule processing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.13 FFW_RUL_EXT.1.9 TSS

Objective	The evaluator shall verify that the TSS describes the process for applying stateful traffic filtering rules and also that the behavior (either by default, or as configured by the administrator) is to deny packets when there is no rule match unless another required conditions allows the network traffic (i.e., FFW_RUL_EXT.1.5 or FFW_RUL_EXT.2.1).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the process for applying stateful traffic filtering rules and states that the behavior is to deny packets when there is no rule match unless another required condition allows the network traffic. Upon investigation, the evaluator found that the TSS states that: **By default, the TOE behavior is to deny packets when there is no rule match unless another required condition allows the network traffic If a security risk is found in the packet. e.g. denial-of-service attacks, the packet is dropped and an event is logged.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.14 FFW_RUL_EXT.1.9 Guidance

Objective	The evaluator shall verify that the guidance documentation describes the behavior if no rules or special conditions apply to the network traffic. If the behavior is configurable, the

	evaluator shall verify that the guidance documentation provides the appropriate instructions to configure the behavior to deny packets with no matching rules.
Evaluator Findings	The evaluator examined the section titled **Configuring Default Deny-All and Reject Rules** in the AGD to verify that it describes the behavior if no rules or special conditions apply to the network traffic and, if the behavior is configurable, provides the appropriate instructions to configure the behavior to deny packets with no matching rules. Upon investigation, the evaluator found that the AGD states that **By default, security devices running Junos OS deny traffic unless rules are explicitly created to allow it using the following command** and then also states the CLI command needed to configure this befaviour. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.15 FFW_RUL_EXT.1.10 TSS

Objective	The evaluator shall verify that the TSS describes how the TOE tracks and maintains information relating to the number of half-open TCP connections. The TSS should identify how the TOE behaves when the administratively defined limit is reached and should describe under what circumstances stale half-open connections are removed (e.g. after a timer expires).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE tracks and maintains information relating to the number of half-open TCP connections. Upon investigation, the evaluator found that the TSS states that: **The TOE can be configured to drop connection attempts after a defined number of half-open TCP connections using the Junos screen 'tcp syn-flood', which provides both source and destination thresholds on the number of uncompleted TCP connections, as well as a timeout period. The source threshold option allows administrators to specify the number of SYN segments received per second from a single source IP address—regardless of the destination IP address—before Junos OS begins dropping connection requests from that source. Similarly, the destination threshold option allows administrators to specify the number of SYN segments received per second for a single destination IP address before Junos OS begins dropping connection requests to that destination.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.1.16 FFW_RUL_EXT.1.10 Guidance 1

Objective	The evaluator shall verify that the guidance documentation describes the behaviour of imposing TCP half-open connection limits and its default state if unconfigured. The evaluator shall verify that the guidance clearly indicates the conditions under which new connections will be dropped e.g. per-destination or per-client.
Evaluator Findings	The evaluator examined the section titled **Configuring TCP SYN Flood Attack Screen** in the AGD to verify that it describes the behaviour of imposing TCP half-open connection limits and its default state if unconfigured and indicates the conditions under which new connections will be dropped. Upon investigation, the evaluator found that the AGD states that the guidance describes the behavior of the TOE as it relates to TCP connections.
Verdict	Pass.

5.6.2 FFW_RUL_EXT.2

5.6.2.1 FFW_RUL_EXT.2.1 TSS

Objective	The evaluator shall verify that the TSS identifies the protocols that can cause the automatic creation of dynamic packet filtering rules. In some cases rather than creating dynamic rules, the TOE might establish stateful sessions to support some identified protocol behaviors. The evaluator shall verify that the TSS explains the dynamic nature of session establishment and removal. The TSS also shall explain any logging ramifications. The evaluator shall verify that for each of the protocols selected, the TSS explains the dynamic nature of session establishment and removal specific to the protocol.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies the protocols that can cause the automatic creation of dynamic packet filtering rules, explains the dynamic nature of session establishment and removal, and explains the dynamic nature of session establishment and removal specific to the selected protocols. Upon investigation, the evaluator found that the TSS states that: **The TOE supports FTP (RFC 959) to dynamically establish sessions allowing network traffic according to Administrator rules. Session events will be logged in accordance with 'log' operations defined in the rules. Source and destination addresses, source and destination ports, transport layer protocol, and TOE Interface are recorded in each log record.** **Junos implements what is referred to as an Application Layer gateway (ALG) that inspects FTP traffic to determine the port number used for data sessions. The ALG permits data traffic for the duration of the session, closing the port when the session ends.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.6.2.2 FFW_RUL_EXT.2.1 Guidance

Objective	The evaluator shall verify that the guidance documentation describes dynamic session establishment capabilities. The evaluator shall verify that the guidance documentation describes the logging of dynamic sessions consistent with the TSS.
Evaluator Findings	The evaluator examined the section titled **Configuring Traffic Filter Rules** in the AGD to verify that it describes dynamic session establishment capabilities. Upon investigation, the evaluator found that the AGD states that **Traffic filter rules can be configured on a device to enforce validation against protocols attributes and direct traffic accordingly to the configured attributes** and states the CLI commands needed to configure traffic filter rules for FTP, which is the protocol for which dynamic definition of rules is supported by the TOE as per the ST. The evaluator examined the section titled **Configuring Traffic Filter Rules** in the AGD to verify that it describes the logging of dynamic sessions consistently with the TSS. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure logging for traffic filter rules for FTP, which is the protocol for which dynamic definition of rules is supported by the TOE as per the ST. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.7 TSS and Guidance Activities (Identification and Authentication)

5.7.1 FIA_AFL.1

5.7.1.1 FIA_AFL.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked; the method by which the remote administrator is prevented from successfully logging on to the TOE; and the actions necessary to restore this ability. Upon investigation, the evaluator found that the TSS states that: **The retry-options can be configured to specify the action to be taken if the administrator fails to enter valid username/password credentials for password authentication when attempting to authenticate via remote access. The retry-options are applied following the first failed login attempt for a given username. The length of delay (5-10 seconds) after each failed attempt is specified by the backoff-factor, and the increase of the delay for each subsequent failed attempt is specified by the backoff-threshold (1-3). The tries-before-disconnect sets the maximum number of times (1-10) the administrator is allowed to enter a password to attempt to log in to the device through SSH before the connection is disconnected. Each failed attempt is tracked by the username. When the tries-before-disconnect number is reached for any particular user, that username is locked and cannot be used to authenticate remotely** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.1.2 FIA_AFL.1 TSS 2

Objective	The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available. Upon investigation, the evaluator found that the TSS states that: **Even when an account is blocked for remote access to the TOE, an administrator is always able to login locally through the serial console and the administrator can attempt authentication via remote access after the maximum timeout period of 24 hours.** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.7.1.3 FIA_AFL.1 Guidance 1

Objective	The evaluator shall examine the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described.
Evaluator Findings	The evaluator examined the section titled **Limiting the Number of User Login Attempts for SSH Sessions** in the AGD to verify that it provides instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented), and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). Upon investigation, the evaluator found that the AGD states that: **If the remote administrator presents a valid username and password, access to the Target of Evaluation (TOE) is granted. If the credentials are invalid, the TOE allows the authentication to be retried after an interval that starts after 1 second and increases exponentially. If the number of authentication attempts exceed the configured maximum, no authentication attempts are accepted for a configured time interval. When the interval expires, authentication attempts are again accepted.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.1.4 FIA_AFL.1 Guidance 2

Objective	The evaluator shall examine the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1.
Evaluator Findings	The evaluator examined the section titled **Limiting the Number of User Login Attempts for SSH Sessions** in the AGD to verify that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1. Upon investigation, the evaluator found that the AGD states that: **By configuring ssh root-login deny , you can ensure the root account remains active and continues to have local administrative privileges to the TOE even if other remote users are logged off.** Based on these findings, this assurance activity is considered satisfied.

5.7.2 FIA_PMG_EXT.1

5.7.2.1 FIA_PMG_EXT.1.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it contains the lists of the supported special character(s) and minimum and maximum number of charters supported for administrator passwords.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS contains the lists of the supported special character(s) and minimum and maximum number of charters supported for administrator passwords. Upon investigation, the evaluator found that the TSS states that: **Authentication data for fixed password authentication is a case-sensitive, alphanumeric value. The password has a:** • **Minimum length of 10 characters and maximum length of 20 characters** • **Must contain characters from at least two different character sets (upper, lower, numeric, punctuation)** • **Can be up to 20 ASCII characters in length (control characters are not recommended).** **Any standard ASCII, extended ASCII and Unicode characters can be selected when choosing a password.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.2.2 FIA_PMG_EXT.1.1 Guidance 1

Objective	The evaluator shall examine the guidance documentation to determine that it: a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported.
Evaluator Findings	The evaluator examined the section titled **Understanding the Associated Password Rules for an Authorized Administrator** in the AGD to verify that it identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and provides instructions on setting the minimum password length and describes the valid minimum password lengths supported. Upon investigation, the evaluator found that the AGD states that: **Use the following guidelines and configuration options for passwords and when selecting passwords for authorized administrator accounts. Passwords should be:** • **Changed periodically.** • **Private and not shared with anyone.** • **Contain a minimum of 10 characters. The minimum password length is 10 characters.** • **Include both alphanumeric and punctuation characters, composed of any combination of upper and lowercase letters, numbers, and special characters such as, "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". There should be at least a change in one case, one or more digits, and one or more punctuation marks.** • **Contain character sets. Valid character sets include uppercase letters, lowercase letters, numbers, punctuation, and other special characters**

	• **Contain the minimum number of character sets or character set changes. The minimum number of character sets required in plain-text passwords in Junos FIPS is 3.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.3 FIA_PSK_EXT.1

5.7.4 FIA_PSK_EXT.1/VPN

5.7.4.1 FIA_PSK_EXT.1/VPN TSS 1

Objective	The evaluator shall examine the TSS to ensure that it identifies all protocols that allow both text-based and bit-based pre-shared keys, and states that text-based pre-shared keys of 22 characters are supported. For each protocol identified by the requirement, the evaluator shall confirm that the TSS states the conditioning that takes place to transform the text-based pre-shared key from the key sequence entered by the user (e.g., ASCII representation) to the bit string used by the protocol, and that this conditioning is consistent with the last selection in the FIA_PSK_EXT.1.3 requirement.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies all protocols that allow both text-based and bit-based pre-shared keys, and states that text-based pre-shared keys of 22 characters are supported. Upon investigation, the evaluator found that the TSS states that **the TOE accepts Unicode characters to specify test-based pre-shared keys for IPsec only. Unicode characters are encoded as UTF-8 and treated as multiple bytes – up to 4 bytes depending on the character**. **The maximum length limit for text-based pre-shared keys enforced by TOE is 255 bytes. When a pre-shared key is only composed of ASCII characters this limit is equivalent to 255 characters. The text-based pre-shared or bit-based keys may contain upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". The TOE accepts pre-shared text keys and converts the text string into an authentication value as per RFC 2409 for IKEv1 or RFC 4306 for IKEv2, using the PRF that is configured as the hash algorithm for the IKE exchanges. There is no difference between how the TOE processes text-based or bit-based pre-shared keys.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.4.2 FIA_PSK_EXT.1/VPN Guidance 1

Objective	The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on the composition of strong text-based pre-shared keys, and (if the selection indicates keys of various lengths can be entered) that it provides information on the merits of shorter or longer pre-shared keys. The guidance must specify the allowable characters for pre-shared keys, and that list must be a super-set of the list contained in FIA_PSK_EXT.1.2.
Evaluator Findings	The evaluator examined the section titled **Configuring an IPsec VPN with a Preshared Key for IKE Authentication** in the AGD to verify that it provides guidance to administrators on the composition of strong text-based pre-shared keys, and (if the selection indicates keys of various lengths can be entered) that it provides information on the merits of shorter or longer

	pre-shared keys. Upon investigation, the evaluator found that the AGD states that **TOE accepts ASCII preshared or bit-based keys up to 255 characters (and their binary equivalents) that contain uppercase and lowercase letters, numbers, and special characters such as !, @, #, $, %, ^, &, *, (, and).** In relation to the strength of pre-shared keys, the evaluator found that the AGD states that **The Junos OS does not impose minimum complexity requirements for preshared keys. Hence, users are advised to carefully choose long preshared keys of sufficient complexity.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.4.3 FIA_PSK_EXT.1/VPN Guidance 2

Objective	The evaluator shall confirm the operational guidance contains instructions for either entering bit-based pre-shared keys for each protocol identified in the requirement, or generating a bit-based pre-shared key (or both). The evaluator shall also examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1 in the Base-PP.
Evaluator Findings	The evaluator examined the section titled **Configuring an IPsec VPN with a Preshared Key for IKE Authentication** in the AGD to verify that it contains instructions for either entering bit-based pre-shared keys for each protocol identified in the requirement, or generating a bit-based pre-shared key (or both), and it describes the process by which the bit-based pre-shared keys are generated. Upon investigation, the evaluator found that the AGD states the CLI commands needed to enter pre-shared keys used for IPsec connections. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.5 FIA_UIA_EXT.1

5.7.5.1 FIA_UIA_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it describes the logon process for each logon method (local, remote (HTTPS, SSH, etc.)) supported for the product. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a "successful logon".
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the logon process for each logon method supported for the product. Upon investigation, the evaluator found that the TSS states that: **Following TOE initialization, the login() process is listening for a connection at the local console. This 'login' process can be accessed through either direct connection to the local console or following successful establishment of a remote management connection over SSH, when a login prompt is displayed.** **This login process identifies and authenticates the user using PAM operations. The login process does two things; it first establishes that the requesting user is whom they claim to be and second provides them with an interactive Junos Command interactive command line interface (CLI).** **The SSH daemon supports public key authentication by looking up a public key in an authorized keys file located in the directory '.ssh' in the user's home directory (i.e.**

	'~/.ssh/') and this authentication method will be attempted before any other if the client has a key available. The SSH daemon will ignore the authorized keys file if it or the directory '.ssh' or the user's home directory are not owned by the user or are writeable by anyone else. **For password authentication, login() interacts with a user to request a username and password to establish and verify the user's identity. The username entered by the administrator at the username prompt is reflected to the screen, but no feedback to screen is provided while the entry made by the administrator at the password prompt until the Enter key is pressed. login() uses PAM Library calls for the actual verification of this data. The password is hashed and compared to the stored value, and success/failure is indicated to login(). PAM is used in the TOE to support authentication management, account management, session management and password management. Login primarily uses the session management and password management functionality offered by PAM.** **The TOE requires users to provide unique identification and authentication data (passwords/public key) before any access to the system is granted.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.5.2 FIA_UIA_EXT.1 TSS 2

Objective	The evaluator shall examine the TSS to determine that it describes which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes which actions are allowed before user identification and authentication. Upon investigation, the evaluator found that the TSS states that: **Prior to authentication, the only Junos OS managed responses provided to the administrator are:** • **Display of the access banner** • **ICMP echo responses.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.5.3 FIA_UIA_EXT.1 Guidance 1

Objective	The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator shall ensure the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator shall determine that the guidance documentation provides sufficient instruction on limiting the allowed services.
Evaluator Findings	The evaluator examined the section titled **Configuring Administrative Credentials and Privileges** in the AGD to verify that it describes any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in.

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

5.7.6 FIA_UAU.7

5.7.6.1 FIA_UAU.7 Guidance 1

Objective	The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed.
Evaluator Findings	Upon investigation, the evaluator found that the TOE gives no visual feedback while entering authentication data for each local login allowed. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.7 FIA_X509_EXT.1/Rev

5.7.7.1 FIA_X509_EXT.1/Rev TSS 1

Objective	The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). It is expected that revocation checking is performed when a certificate is used in an authentication step and when performing trusted updates (if selected). It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). Upon investigation, the evaluator found that the TSS states that: **To validate certificates, the TOE extracts the subject, issuer, subjects public key, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. The TOE verifies the validity of the signature. If the signature is not valid, the validation fails. It then confirms that the current date and time is within the valid time period specified in the certificate.** **If the TOE has been configured to perform a revocation check using CRL (as specified in RFC 5280 Section 6.3). If the CRL fails to download, the certificate is considered to have failed validation, unless the option to skip CRL checking on download failure has been enabled.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.7.2 FIA_X509_EXT.1/Rev TSS 2

Objective	The TSS shall describe when revocation checking is performed and on what certificates. If the revocation checking during authentication is handled differently depending on whether a full certificate chain or only a leaf certificate is being presented, any differences must be summarized in the TSS section and explained in the Guidance.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes when revocation checking is performed and on what certificates. Upon investigation, the evaluator found that the TSS states that: **If the TOE has been configured to perform a revocation check using CRL (as specified in RFC 5280 Section 6.3). If the CRL fails to download, the certificate is considered to have failed validation, unless the option to skip CRL checking on download failure has been enabled.** **The TOE validates a certificate path by building a chain of (at least 3) certificates based upon issuer and subject linkage, validating each according the certificate validation procedure described above. If any certificate in the chain fails validation, the validation fails as a whole. A self-signed certificate is not required to be at the root of the certificate chain.** **The TOE determines if a certificate is a CA certificate by requiring the CA:true flag to be present in the basicConstraints section.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.7.3 FIA_X509_EXT.1/Rev Guidance 1

Objective	The evaluator shall also ensure that the guidance documentation describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE and describes how certificate revocation checking is performed and on which certificate. Upon investigation, the evaluator found that the AGD states that **To validate certificates, the TOE extracts the subject, issuer, subjects public key, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. The TOE verifies the validity of the signature. If the signature is not valid, the validation fails. It then confirms that the current date and time is within the valid time period specified in the certificate. If the TOE has been configured to perform a revocation check using CRL (as specified in RFC 5280 Section 6.3). If the CRL fails to download, the certificate is considered to have failed validation, unless the option to skip CRL checking on download failure has been enabled**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.8 FIA_X509_EXT.2

5.7.8.1 FIA_X509_EXT.2 TSS & Guidance 1

Objective	The evaluator shall check the TSS to ensure that it describes how the TOE chooses which certificates to use and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE chooses which certificates to use. Upon investigation, the evaluator found that the TSS states that: **When configuring an IKE policy, the certificate name must be set so the TOE knows which certificate to use for authentication.** The evaluator examined the section titled **Configuring VPNs** in the AGD to ensure that it includes any necessary instructions for configuring the operating environment so that the TOE can use the certificates. Upon investigation, the evaluator found that the AGD states the necessary commands and instructions needed to configure the operating enironment so that the TOE can use the certificates. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.8.2 FIA_X509_EXT.2 TSS & Guidance 2

Objective	The evaluator shall examine the TSS to confirm that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The evaluator shall verify that any distinctions between trusted channels are described. If the requirement that the administrator is able to specify the default action, then the evaluator shall ensure that the guidance documentation contains instructions on how this configuration action is performed.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. Upon investigation, the evaluator found that the TSS states that: **If the TSF cannot establish a connection to determine the validity of a certificate, the TOE takes the action configured by the administrator. In the NDcPP deployment, "disable on-download-failure" may be set for a CA to allow connections to be established when CRLs could not be retrieved . Otherwise, connections involving a CA for which the specified CRL could not be retrieved are rejected.** **For public key-based authentication of IPsec connections, Junos OS validates the X.509 certificates by extracting the subject, issuer, signature, basicConstraints and validity period fields. If any of those fields is not present, the validation fails. The issuer is looked up in the PKI database. If the issuer CA is not present, or if the issuer certificate does not have the CA:true flag in the basicConstraints section, the validation fails. Junos OS verifies the validity of the signature. If the signature is not valid, the validation fails. It then confirms that the current date and time is within the valid time period specified in the certificate.** The evaluator examined the sections titled **Configuring VPNs** in the AGD to verify that, if the requirement that the administrator is able to specify the default action, the guidance documentation contains instructions on how this configuration action is performed. Upon investigation, the evaluator found that the AGD suggests that the 'disable-on-download-

	failure' option is used to override the default behavior and permit certificate verification even if the CRL fails to download. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.7.8.3 FIA_X509_EXT.2 Guidance 1

Objective	The evaluator shall also ensure that the guidance documentation describes the configuration required in the operating environment so the TOE can use the certificates. The guidance documentation shall also include any required configuration on the TOE to use the certificates. The guidance document shall also describe the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel.
Evaluator Findings	The evaluator examined the sections titled **Configuring VPNs** in the AGD. Upon investigation, the evaluator found that the AGD states the necessary CLI commands to configure the TOE to use the certificates. The authentication method in the IKE proposal is to be mentioned as rsa or ecdsa signature along with the certificate name mentioned in the ike policy. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.9 FIA_X509_EXT.3

5.7.9.1 FIA_X509_EXT.3 TSS 1

Objective	If the ST author selects "device-specific information", the evaluator shall verify that the TSS contains a description of the device-specific fields used in certificate requests.
Evaluator Findings	The evaluator examined the section titled **FIA_X509_EXT.3 X.509 Certificate Requests** in the Security Target to verify that does not select "device-specific" information. This work unit is not applicable.
Verdict	NA

5.7.9.2 FIA_X509_EXT.3 Guidance 1

Objective	The evaluator shall check to ensure that the guidance documentation contains instructions on requesting certificates from a CA, including generation of a Certificate Request. If the ST author selects "Common Name", "Organization", "Organizational Unit", or "Country", the evaluator shall ensure that this guidance includes instructions for establishing these fields before creating the Certification Request.
Evaluator Findings	The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that it contains instructions on requesting certificates from a CA, including generation of a Certification Request. Upon investigation, the evaluator found that the AGD states the CLI commands needed to request certificates from a CA and to generate a certification request. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8 TSS and Guidance Activities (Security Management)

5.8.1 FMT_MOF.1/ManualUpdate

5.8.1.1 FMT_MOF.1/ManualUpdate Guidance 1

Objective	The evaluator shall examine the guidance documentation to determine that any necessary steps to perform manual update are described. The guidance documentation shall also provide warnings regarding functions that may cease to operate during the update (if applicable).
Evaluator Findings	The evaluator examined the section titled **Configuring Roles and Authentication** Methods in the AGD to verify that it describes any necessary steps to perform manual update. Upon investigation, the evaluator found that the AGD states that the 'request vmhost software add' command is used to perfor, a manual update. The evaluator examined the section titled **Configuring Roles and Authentication Methods** in the AGD to verify that it provides warnings regarding functions that may cease to operate during the update (if applicable). Upon investigation, the evaluator found that the AGD states that **Some functionalities might be impacted during the reboot following the software upgrade and not during the upgrade.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.2 FMT_FMT_MOF.1/Functions

5.8.2.1 FMT_MOF.1/Functions TSS 1

Objective	For non-distributed TOEs, the evaluator shall ensure the TSS for each administrative function identified the TSS details how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies each administrative function identified the TSS details how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE). Upon investigation, the evaluator found that the TSS states that: **Syslog can be configured to store the audit logs locally, and optionally to send them to one or more syslog log servers in real time via Netconf over SSH.** **Only a Security Administrator can read log files or delete log and archive files through the CLI interface or through direct access to the filesystem having first authenticated as a Security Administrator. The syslogs are automatically deleted locally according to configurable limits on storage volume. The default maximum size is 1Gb. The default maximum size can be modified by the user, using the "`size`" argument for the "`set system syslog`" CLI command** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.2.2 FMT_MOF.1/Functions Guidance 2

Objective	For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE) are performed to include required configuration settings.
Evaluator Findings	The evaluator examined the section titled **Configuring Roles and Authentication Methods , Configuring the Remote Syslog Server** and **Configuring Audit Log Options in the Evaluated Configuration** in the AGD to verify that it describes how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE) are performed to include required configuration settings. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.3 FMT_MOF.1/Services

5.8.3.1 FMT_MOF.1/Services TSS 2

Objective	For non-distributed TOEs, the evaluator shall ensure the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed. Upon investigation, the evaluator found that the TSS states that: **Managed Functions:** • **Transmission of audit data to an external IT entity, including Start/stop and modify the behaviour of the trusted communication channel to external syslog server (netconf over SSH) and the trusted path for remote Administrative sessions (SSH)** • **Handling of audit data, including setting limits of log file size** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.3.2 FMT_MOF.1/Services Guidance 2

Objective	For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed.
Evaluator Findings	The evaluator examined the section titled **Configuring Event Logging to a Remote Server** and **Configuring Audit Log Options in the Evaluated Configuration** in the AGD to verify that it describes how the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed. Upon investigation, the evaluator found that the AGD states the steps to start event logging to a remote server along with the steps to set audit log file options. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.8.4 FMT_MTD.1/CoreData

5.8.4.1 FMT_MTD.1/CoreData TSS 1

Objective	The evaluator shall examine the TSS to determine that, for each administrative function identified in the guidance documentation; those that are accessible through an interface prior to administrator log-in are identified. For each of these functions, the evaluator shall also confirm that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies administrative functions that are accessible through an interface prior to administrator log-in. Upon investigation, the evaluator found that the TSS states that: **No administrative functions are accessible prior to logging in.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users. Upon investigation, the evaluator found that the TSS states that: **The Administrator is the only role with the ability to manage the TSF data. The Administrator can perform management functions via a specialized interface.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.4.2 FMT_MTD.1/CoreData TSS 2

Objective	If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator shall examine the TSS to determine that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that, if the TOE supports handling of X.509v3 certificates and implements a trust store, the TSS contains sufficient information to describe how the ability to manage the TOE's trust store is restricted. Upon investigation, the evaluator found that the TSS states that: **The Administrator is the only role with the ability to manage the TSF data. The Administrator can perform management functions via a specialized interface** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.4.3 FMT_MTD.1/CoreData Guidance 1

Objective	The evaluator shall review the guidance documentation to determine that each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.
Evaluator Findings	The evaluator examined the section titled **Configuring Roles and Authentication**

Methods in the AGD to verify that it identifies each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP. Upon investigation, the evaluator found that the AGD describes how administrative users can configure the TSF-data manipulating functions for the TOE. The evaluator found that the configuration of the following functionality is described within AGD,

Administer the TOE locally and remotely	**Configuring Roles and Authentication Methods**
Configure the access banner	**Configuring a System Login Message and Announcement**
Configure the session inactivity time before session termination or locking	**Configuring the User Session Idle Timeout**
Update the TOE, and to verify the updates use digital signature prior to installing those updates	**Installing Software on Network Services Platform with a Single Routing Engine**
Configure the authentication failure parameters	**Limiting the Number of User Login Attempts for SSH Sessions**
Configure firewall rules	**Configuring Traffic Filtering Rules**
Configure the cryptographic functionality	**Configuring SSH on the Evaluated Configuration for NDcPP, Configuring VPN on a Device Running Junos OS**
Configure the lifetime for IPsec SAs	**Configuring VPN on a Device Running Junos OS**
Import X.509v3 certificates	**Configuring VPN on a Device Running Junos OS**
Enable, disable, determine and modify the behavior of all the security functions of the TOE identified [VPNGW_MOD] to the Administrator	**Configuring VPNs**
Configure all security management functions identified in [VPNGW_MOD]	**Configuring VPNs**
Ability to configure audit behavior	**Configuring Audit Log Options in the Evaluated Configuration**
Ability to configure thresholds for SSH rekeying	**Configuring SSH on the Evaluated Configuration for NDcPP**
Ability to re-enable an Administrator account	**Understanding the Associated Password Rules for an Authorized Administrator**
Ability to set the time which is used for time-stamps	**Configuring the time and date**
Ability to configure the reference identifier for the peer	**Configuring VPN on a Device Running Junos OS**

	The evaluator found that this encompasses all of the TSF-data manipulating functionality required by the NDcPP. Based on these findings, this assurance activity is considered satisfied Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.4.4 FMT_MTD.1/CoreData Guidance 2

Objective	If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator shall also review the guidance documentation to determine that it explains how to designate a CA certificate a trust anchor.
Evaluator Findings	The evaluator examined the section titled **Configuring Roles and Authentication Methods** in the AGD to verify that, if the TOE supports handling of X.509v3 certificates and provides a trust store, it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. Upon investigation, the evaluator found that the AGD states that **Administrative users (Security Administrator) must provide unique identification and authentication data before any administrative access to the system is granted.** The evaluator examined the section titled **Configuring VPNs** in the AGD to verify that, if the TOE supports loading of CA certificates, it provides sufficient information for the administrator to securely load CA certificates into the trust store and that it explains how to designate a CA certificate a trust anchor. Upon investigation, the evaluator found that the AGD states the CLI commands needed to securely load CA certificates into the TOE's trust store and to designate a CA certificate as a trust anchor. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.5 FMT_MTD.1/CryptoKeys

5.8.5.1 FMT_MTD.1/CryptoKeys TSS 1

Objective	For distributed TOEs it is required to verify the TSS to ensure that it describes how every function related to security management is realized for every TOE component and shared between different TOE components. The evaluator shall confirm that all relevant aspects of each TOE component are covered by the FMT SFRs.
Verdict	NA. TOE is not distributed.

5.8.5.2 FMT_MTD.1/CryptoKeys TSS 2

Objective	For non-distributed TOEs, the evaluator shall ensure the TSS lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS lists the keys the Security Administrator is able to manage to include the

	options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed. Upon investigation, the evaluator found that the TSS states that: **The Security Administrator has the capability to:** • **Manage crypto keys:** ○ **SSH key generation (ecdsa, ssh-rsa)** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.5.3 FMT_MTD.1/CryptoKeys Guidance 1

Objective	For distributed TOEs it is required to verify the Guidance Documentation to describe management of each TOE component. The evaluator shall confirm that all relevant aspects of each TOE component are covered by the FMT SFRs.
Evaluator Findings	NA. TOE is not distributed.
Verdict	Pass

5.8.5.4 FMT_MTD.1/CryptoKeys Guidance 2

Objective	For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how those operations are performed.
Evaluator Findings	The evaluator examined the section titled **Configuring Roles and Authentication Methods** in the AGD to verify that it lists the keys the Security Administrator is able to manage to include the options available and how those operations are performed. Upon investigation, the evaluator found that the AGD states that the Security Administrator is **Responsible for the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the evaluated product. The SSH keys are managed by the security administrator.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.6 FMT_SMF.1

5.8.6.1 FMT_SMF.1 TSS 1

Objective	The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface). The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the TSS to verify that it details which security management functions are available through which interface(s). Upon investigation, the evaluator found that the TSS states that:

	The Security Administrator is associated with the defined login class "security-admin", which has the necessary permission set to permit the administrator to perform all tasks necessary to manage Junos OS in accordance with the requirements of [ND_cPP], using both the local as well as the remote administrative interface.
	The Security Administrator has the capability to:
	• **Administer the TOE locally via the serial ports on the physical device or remotely over an SSH connection.**
	• **Initiate a manual update of TOE software:**
	  ○ **Query currently executing version of TOE software (both Junos OS and underlying Wind River Linux Host OS)**
	  ○ **Verify update using published digital signature**
	• **Manage Functions:**
	  ○ **Transmission of audit data to an external IT entity, including Start/stop and modify the behaviour of the trusted communication channel to external syslog server (netconf over SSH) and the trusted path for remote Administrative sessions (SSH)**
	  ○ **Handling of audit data, including setting limits of log file size**
	• **Manage TSF data:**
	  ○ **Create, modify, delete administrator accounts, including configuration of authentication failure parameters**
	  ○ **Reset administrator passwords**
	• **Manage crypto keys:**
	  ○ **SSH key generation (ecdsa, ssh-rsa)**
	• **Perform management functions:**
	  ○ **Configure the access banner**
	  ○ **Configure the session inactivity time before session termination or locking, including termination of session when serial console cable is disconnected**
	  ○ **Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;**
	  ○ **Ability to import X.509v3 certificates**
	  ○ **Manage cryptographic functionality, including:**
	  ▪ **ssh ciphers**
	  ▪ **hostkey algorithm**
	  ▪ **key exchange algorithm**
	  ▪ **hashed message authentication code**
	  ▪ **thresholds for SSH rekeying**
	  ○ **Set the system time**
	  ○ **Ability to configure Firewall rules;**
	  ○ **Ability to configure the VPN-associated cryptographic functionality;**
	  ○ **Definition of packet filtering rules;**
	  ○ **Association of packet filtering rules to network interfaces;**
	  ○ **Ordering of packet filtering rules by priority;**

	○ **Ability to configure the IPsec functionality, including configuration of IKE lifetime-seconds (within range 180 to 86400 , with default value of 180 seconds), IPsec lifetime-seconds (within range 180 to 86400, with default value of 28800 seconds), and Lifetime-kilobytes (within range 64 to 4294967294 kilobytes) and ability to configure the reference identifier for the peer;** ○ **Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality** ○ **Modify these parameters that define the network traffic to be collected and analysed:** ▪ **Source IP addresses (host address and network address);** ▪ **Destination IP addresses (host address and network address);** ▪ **Source port (TCP and UDP);** ▪ **Destination port (TCP and UDP);** ▪ **Protocol (IPv4 and IPv6)** ▪ **ICMP type and code** ○ **Update (import) IPS signatures;** ○ **Create custom IPS signatures;** ○ **Configure anomaly detection;** ○ **Enable and disable actions to be taken when signature or anomaly matches are detected;** ○ **Modify thresholds that trigger IPS reactions;** ○ **Modify the duration of traffic blocking actions;** ○ **Modify the known-good and known-bad lists (of IP addresses or address ranges);** ○ **Configure the known-good and known-bad lists to override signature-based IPS policies.** **Security Administrators are able to initiate an update of the TOE firmware if a new version of the TOE firmware is available. Updates are downloaded and applied manually (there is no automatic updating of the Junos OS).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.6.2 FMT_SMF.1 Guidance 1

Objective	The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface. The evaluator shall ensure the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.
Evaluator Findings	The evaluator examined the section titled **Understanding Management Interfaces** in the AGD to verify that it describes the local administrative interface. Upon investigation, the evaluator found that the AGD states that:

	Local Management Interfaces—The RJ-45 console port on the front panel of a device is configured as RS-232 data terminal equipment (DTE). You can use the command-line interface (CLI) over this port to configure the device from a terminal. No configuration or warnings are necessary for the administrator to ensure the interface is local. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.7 FMT_SMF.1/IPS

5.8.7.1 FMT_SMF.1/IPS TSS

Objective	The evaluator shall verify that the TSS describes how the IPS data analysis and reactions can be configured. Note that this activity should have been addressed with the TSS assurance activities for IPS_ABD_EXT.1, IPS_IPB_EXT.1 and IPS_ABD_EXT.1.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the IPS data analysis and reactions can be configured. Upon investigation, the evaluator found that this activity has been addressed with the TSS assurance activities for IPS_ABD_EXT.1, IPS_IPB_EXT.1 and IPS_ABD_EXT.1. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.7.2 FMT_SMF.1/IPS Guidance

Objective	The evaluator shall verify that the operational guidance describes the instructions for each function defined in the SFR, describes how to configure the IPS data analysis and reactions, including how to set any configurable defaults and how to configure each of the applicable analysis pattern matching methods and reaction modes.
Evaluator Findings	The evaluator examined the section titled **IDP Extended Package Configuration Overview** in the AGD to verify that it describes the instructions for each function defined in the SFR, describes how to configure the IPS data analysis and reactions, including how to set any configurable defaults and how to configure each of the applicable analysis pattern matching methods and reaction modes. Upon investigation, the evaluator found that the AGD describes the CLI commands to configure each of the function defined in the SFR as follows:

Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality	**IDP Extended Package Configuration Overview**
Modify these parameters that define the network traffic to be collected and analyzed: • Source IP addresses (host address and network address) • Destination IP addresses (host address and network address) • Source port (TCP and UDP)	**IDP Extended Package Configuration Overview**

	• Destination port (TCP and UDP) • Protocol (IPv4 and IPv6) • ICMP type and code	
	Update (import) signatures	**IDP Extended Package Configuration Overview**
	Create custom signatures	**IDP Extended Package Configuration Overview**
	Configure anomaly detection	**IDP Extended Package Configuration Overview**
	Enable and disable actions to be taken when signature or anomaly matches are detected	**IDP Extended Package Configuration Overview**
	Modify thresholds that trigger IPS reactions	**IDP Extended Package Configuration Overview**
	Modify the duration of traffic blocking actions	**IDP Extended Package Configuration Overview**
	Modify the known-good and known-bad lists (of IP addresses or address ranges)	**Configuring Security Flow Policies**
	Configure the known-good and known-bad lists to override signature-based IPS policies]	**Configuring Security Flow Policies**
	Based on these findings, this assurance activity is considered satisfied.	
Verdict	Pass.	

5.8.8 FMT_SMF.1/VPN

5.8.8.1 FMT_SMF.1/VPN TSS

Objective	The evaluator shall examine the TSS to confirm that all management functions specified in FMT_SMF.1/VPN are provided by the TOE. As with FMT_SMF.1 in the Base-PP, the evaluator shall ensure that the TSS identifies what logical interfaces are used to perform these functions and that this includes a description of the local administrative interface.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS states that all management functions specified in FMT_SMF.1/VPN are provided by the TOE. Upon investigation, the evaluator found that the TSS states that all management functions specified in FMT_SMF.1/VPN are provided by the TOE. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.8.2 FMT_SMF.1/VPN Guidance

Objective	The evaluator shall examine the operational guidance to confirm that all management functions specified in FMT_SMF.1/VPN are provided by the TOE. As with FMT_SMF.1 in the Base-PP, the evaluator shall ensure that the operational guidance identifies what logical

	interfaces are used to perform these functions and that this includes a description of the local administrative interface.
Evaluator Findings	The evaluator examined the section titled **Configuring VPN on a Device Running Junos OS** in the AGD to confirm that all management functions specified in FMT_SMF.1/VPN are provided by the TOE. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure the management functions specified in the SFR. The guidance document also describes the local administrative interface in the section titled **Understanding Management Interfaces**. Upon investigation, the evaluator found that the AGD states **Local Management Interfaces—The RJ-45 console port on the front panel of a device is configured as RS-232 data terminal equipment (DTE). You can use the command-line interface (CLI) over this port to configure the device from a terminal.** The evaluator examined the section titled **Configuring VPNs** and found that the operational guidance identifies what logical interfaces are used to perform the VPN functions and states that **ge interfaces are the logical interfaces used for VPN functions**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.9 FMT_SMR.2

5.8.9.1 FMT_SMR.2 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the TSS and the section titled **Understanding Roles and Services for Junos OS in Common Criteria and FIPS Mode** in the AGD to verify that the TOE supported roles and any restrictions of the roles involving administration of the TOE. Upon investigation, the evaluator found that the AGD states that: **the FIPS 140-2 standard defines two user roles: Crypto Officer and FIPS user. These roles are defined in terms of Junos OS user capabilities. All other user types defined for Junos OS in FIPS mode (operator, administrative user, and so on) must fall into one of the two categories: Crypto Officer or FIPS user. For this reason, user authentication in FIPS mode is role-based rather than identity-based.** **In addition to their FIPS roles, both Crypto Officer and user can perform normal configuration tasks on the NFX350 device as individual user configuration allows.** **Crypto Officers and FIPS users perform all FIPS-mode-related configuration tasks and issue all statements and commands for Junos OS in FIPS mode. Crypto Officer and FIPS user configurations must follow the guidelines for Junos OS in FIPS mode.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.8.9.2 FMT_SMR.2 Guidance 1

Objective	The evaluator shall review the guidance documentation to ensure that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration.

Evaluator Findings	The evaluator examined the section titled **Understanding Management Interfaces** in the AGD to verify that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration. Upon investigation, the evaluator found that the AGD states that: **The following management interfaces can be used in the evaluated configuration:** **• Local Management Interfaces—The RJ-45 console port on the front panel of a device is configured as RS-232 data terminal equipment (DTE). You can use the command-line interface (CLI) over this port to configure the device from a terminal.** **• Remote Management Protocols—The device can be remotely managed over any Ethernet interface. SSHv2 is the only permitted remote management protocol that can be used in the evaluated configuration. The remote management protocols J-Web and Telnet are not available for use on the device.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.9 TSS and Guidance Activities (Packet Filtering)

5.9.1 FPF_RUL_EXT.1

5.9.1.1 FPF_RUL_EXT.1.1 TSS 1

Objective	The evaluator shall verify that the TSS provide a description of the TOE's initialization/startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process. The evaluator shall verify that the TSS also includes a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describes the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure. This could include the failure of a component, such as a process being terminated, or a failure within a component, such as memory buffers full and cannot process packets.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS provides a description of the TOE's initialization/startup process and a discussion that supports the assertion that packets cannot flow during this process. The evaluator verified that the TSS includes a narrative that identifies the components involved in processing the network packets and describes the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure Upon investigation, the evaluator found that the TSS states that: **The boot sequence of the TOE appliances also aids in establishing the securing domain and preventing tamping or bypass of security functionality. This includes ensuring the packet filtering rules cannot be bypassed during the boot sequence of the TOE. The following steps list the boot sequence for the TOE:** • **BIOS hardware and memory checks** • **Loading and initialization of the FreeBSD Kernel OS** • **FIPS self-tests and firmware integrity tests are executed**

	• **The init utility is started (mounts file systems, sets up network cards to communicate on the network, and generally starts all the processes that usually are run on a FreeBSD system at startup)** • **Daemon programs such as Internet Service Daemon (INETD), Routing Protocol Daemon (RPD), Syslogd are started; Routing and forwarding tables are initialized** • **Management Daemon (or MGD) is loaded, allowing access to management interface** • **Physical interfaces are active** **Once the interfaces are brought up, they will start to receive and send packets based on the current configuration (or not receive or send any packets if they have not been previously configured).** **Interfaces are brought up only after successful loading of kernel and Information Flow subsystems, and these interfaces cannot send or receive packets unless previously configured by an Administrator.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.9.1.2 FPF_RUL_EXT.1.1 Guidance 1

Objective	The operational guidance associated with this requirement is assessed in the subsequent test assurance activities.
Evaluator Findings	The operational guidance associated with this requirement is assessed in the subsequent test assurance activities.
Verdict	Pass

5.9.1.3 FPF_RUL_EXT.1.4 TSS 1

Objective	The evaluator shall verify that the TSS describes a Packet Filtering policy that can use the following fields for each identified protocol, and that the RFCs identified for each protocol are supported:
	• IPv4 (RFC 791)
	  ○ Source address
	  ○ Destination Address
	  ○ Protocol
	• IPv6 (RFC 2460)
	  ○ Source address
	  ○ Destination Address
	  ○ Next Header (Protocol)
	• TCP (RFC 793)
	  ○ Source Port
	  ○ Destination Port
	• UDP (RFC768)
	  ○ Source Port
	  ○ Destination Port

	The evaluator shall verify that the TSS describes how conformance with the identified RFCs has been determined by the TOE developer (e.g., third party interoperability testing, protocol compliance testing). The evaluator shall verify that each rule can identify the following actions: permit, discard, and log. The evaluator shall verify that the TSS identifies all interface types subject to the Packet Filtering policy and explains how rules are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes a Packet Filtering policy, describes how conformance with the identified RFCs has been determined, each rule can identify the required actions, identifies all interface types subject to the Packet Filtering policy and explains how rules are associated with distinct network interfaces. Upon investigation, the evaluator found that the TSS states that: **The TOE performs stateful network traffic filtering on network packets using the following network traffic protocols and network fields conforming to the described RFCs:** • **RFC 792 ICMPv4: Type, Code** • **RFC 4443 ICMPv6: Type, Code** • **RFC 791 (IPv4): Source address, Destination Address, Transport Layer Protocol** • **RFC 2460 (IPv6): Source address, Destination Address, Transport Layer Protocol** • **RFC 793 (TCP): Source port, Destination port** • **RFC 768 (UDP): Source port, Destination port** **Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.** **The TOE shall allow permit, deny, and log operations to be associated with rules and these rules can be assigned to distinct network interfaces.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.9.1.4 FPF_RUL_EXT.1.4 Guidance 1

Objective	The evaluators shall verify that the operational guidance identifies the following protocols as being supported and the following attributes as being configurable within Packet filtering rules for the associated protocols:
	• IPv4 (RFC 791)
	  ○ Source address
	  ○ Destination Address
	  ○ Protocol
	• IPv6 (RFC 2460)
	  ○ Source address
	  ○ Destination Address
	  ○ Next Header (Protocol)
	• TCP (RFC 793)
	  ○ Source Port

	○ Destination Port • UDP (RFC768) ○ Source Port ○ Destination Port The evaluator shall verify that the operational guidance indicates that each rule can identify the following actions: permit, discard, and log. The evaluator shall verify that the operational guidance explains how rules are associated with distinct network interfaces. The guidance may describe the other protocols contained within the ST (e.g., IPsec, IKE, potentially HTTPS, SSH, and TLS) that are processed by the TOE. The evaluator shall ensure that it is made clear what protocols were not considered as part of the TOE evaluation.
Evaluator Findings	The evaluator examined the section titled **Understanding Protocol Support** in the AGD to verify that it identifies the required protocols as being supported and the required attributes as being configurable within Packet filtering rules, indicates that each rule can identify the required actions, explains how rules are associated with distinct network interfaces, and makes clear what protocols were not considered as part of the TOE evaluation. Upon investigation, the evaluator found that the AGD states the network traffic protocols and network fields used to perform stateful network traffic filtering on network packets. The section titled **Understanding a Security Flow Policy on a Device Running Junos OS** explains the possible actions taken by packet filtering rules and also explains how rules are associated with distinct network interfaces. Upon investigation, the evaluator found that the AGD states that **Each of these policies are associated to zones on which distinct network interfaces are bound**. It also states the following: **The following modes can be defined for a security flow policy to determine how a device directs traffic:** • **Bypass—The Permit option directs the traffic traversing the device through the stateful firewall inspection, but not through the IPsec VPN tunnel.** • **Discard—The Deny option inspects and drops all packets that do not match any Permit policies.** • **Protect—The traffic is routed through an IPsec tunnel based on the combination of route lookup and Permit policy inspection.** • **Log—This option logs traffic and session information for all the modes mentioned above.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.1.5 FPF_RUL_EXT.1.5 TSS 1

Objective	The evaluator shall verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established

	session, and application of administrator defined and ordered ruleset. Upon investigation, the evaluator found that the TSS states that: **The Information Flow subsystem is responsible for processing the arriving packets from the network to the TOE's network interface. Based on Administrator-configured policy, interface and zone information, the packet flows through the various modules of the Information Flow subsystem. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces. By default, the TOE behavior is to deny packets when there is no rule match unless another required condition allows the network traffic If a security risk is found in the packet. e.g. denial-of-service attacks, the packet is dropped and an event is logged. The packet does not continue to the next module for processing. If the packet is not dropped by a given module, the interrupt handling routine calls the function for the next relevant module.** **The IP Classification module retrieves information from packets received on the network interface device, classifies packets into several categories, saves classification information in packet processing context, and provides other modules with that information for assisting further processing.** **The Attack Detection module provides inline attack detection such as IP Spoofing for the security appliance. This module monitors arriving traffic, performs predefined attack detection services (prevents attacks), and issues actions when an attack is found.** **The Session Lookup module performs lookups in the session table which is used for all interfaces based on the information in incoming packets. Specifically, the lookup is based on the exact match of source IP address and port, destination IP address and port, protocol attributes (e.g., SYN, ACK, RST, and FIN), and egress/ingress zone. The input is passed to the module as a set of parameters from the Attack Detection module via a function call. The module returns matching wing if a match is found and 0 otherwise. Sessions are removed when terminated.** **The Session Setup module is only available for packets that do not match current established sessions. It is activated after the Session Lookup module. If packet has a matched session, it will skip the session setup module and proceed to the Security Policy module, and other modules. Eventually if the packet is not destined for the TOE, the Network interface will pass the traffic out of the appliance.** **The Security Policy module examines traffic passing through the TOE (via Session Setup module) and determines if the traffic can pass based on administrator-configured access policies. The Security Policy module is the core of the firewall and IPS functionalities in the TOE: It is the policy enforcement engine that fulfills the security requirements for the user. The Security Policy module will deny packets when there is no policy match unless another policy allows the traffic.** **The Session Setup module performs the auditing of denied packets. If there is a policy to specifically deny traffic, traffic matching this deny policy is dropped and logged in traffic log. If there is no policy to deny traffic, traffic that does not match any policy is dropped and not logged. In either case, Session Setup module does not create any sessions for denied traffic. Sessions are created for allowed traffic.** **The INETD module provides internet services for the TOE. The module listens on designated ports used by internet services such as FTP. When a TCP or UDP packet arrives with a particular destination port number, INETD launches the appropriate server program (e.g., SSHD) to handle the connection.** **The RPD (Routing Protocol Daemon) module provides the implementations and algorithms for the routing protocols and route calculations. The primary goal of the RPD is to create**

	and maintain the Routing Information Base (RIB), which is a database of routing entries. Each routing entry consists of a destination address and some form of next hop information. RPD module maintains the internal routing table and properly distributes routes from the routing table to Kernel subsystem used for traffic forwarding at the Network interface. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.9.1.6 FPF_RUL_EXT.1.5 Guidance 1

Objective	The evaluator shall verify that the operational guidance describes how the order of Packet filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.
Evaluator Findings	The evaluator examined the section titled **Configuring Traffic Filtering Rules** in the AGD to verify that it describes how the order of Packet filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing. Upon investigation, the evaluator found that the AGD states that **The firewall filter terms are evaluated in the order in which they are configured** and also provides instructions so that an administrator can configure the order of rule processing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.1.7 FPF_RUL_EXT.1.6 TSS 1

Objective	The evaluator shall verify that the TSS describes the process for applying Packet filtering rules and also that the behavior (either by default, or as configured by the administrator) is to discard packets when there is no rule match.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the process for applying Packet filtering rules and that the behavior is to deny packets when there is no rule match. Upon investigation, the evaluator found that the TSS states that: **Based on Administrator-configured policy, interface and zone information, the packet flows through the various modules of the Information Flow subsystem. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces. By default, the TOE behavior is to deny packets when there is no rule match unless another required condition allows the network traffic If a security risk is found in the packet. e.g. denial-of-service attacks, the packet is dropped and an event is logged.** **The Information Flow subsystem consists of the following modules:** • **IP Classification Module** • **Attack Detection Module** • **Session Lookup Module** • **Security Policy Module** • **Session Setup Module** • **Inetd Module** • **Rdp Module**

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

5.9.1.8 FPF_RUL_EXT.1.6 TSS 2 **[TD0597]**

Objective	The evaluator shall verify the TSS describes when the IPv4/IPv6 protocols supported by the TOE differ from the full list provided in the RFC Values for IPv4 and IPv6 table.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes when the IPv4/IPv6 protocols supported by the TOE differ from the full list provided in the RFC Values for IPv4 and IPv6 table. Upon investigation, the evaluator found that the TSS states that: **The TOE performs stateful network traffic filtering on network packets using the following network traffic protocols and network fields conforming to the described RFCs:** • **RFC 792 ICMPv4: Type, Code** • **RFC 4443 ICMPv6: Type, Code** • **RFC 791 (IPv4): Source address, Destination Address, Transport Layer Protocol** • **RFC 2460 (IPv6): Source address, Destination Address, Transport Layer Protocol** • **RFC 793 (TCP): Source port, Destination port** • **RFC 768 (UDP): Source port, Destination port** **Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.1.9 FPF_RUL_EXT.1.6 Guidance 1 **[TD0597]**

Objective	The evaluator shall verify that the operational guidance describes the behavior if no rules or special conditions apply to the network traffic. If the behavior is configurable, the evaluator shall verify that the operational guidance provides the appropriate instructions to configure the behavior to discard packets with no matching rules. The evaluator shall verify that the operational guidance describes the range of IPv4/IPv6 protocols supported by the TOE.
Evaluator Findings	The evaluator examined the section titled **Configuring Default Deny-All and Reject Rules** in the AGD to verify that it describes the behavior if no rules or special conditions apply to the network traffic. Upon investigation, the evaluator found that the AGD states that **By default, security devices running Junos OS deny traffic unless rules are explicitly created to allow it using the following command** and then states the CLI command needed to configure this behavior. The evaluator examined the section titled **Understanding Protocol Support** in the AGD to verify that it describes the range of IPv4/IPv6 protocols supported by the TOE. Upon investigation, the evaluator found that the AGD states the range of IPv4/IPv6 protocols and fields supported by the TOE. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.10 TSS and Guidance Activities (Protection of the TSF)

5.10.1 FPT_APW_EXT.1

5.10.1.1 FPT_APW_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS shall also detail passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details all authentication data that are subject to this requirement and the method used to obscure the plaintext password data when stored. Evaluator verified that the TSS details that passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose. Upon investigation, the evaluator found that the TSS states that: **Locally stored authentication credentials are protected:** • **The passwords are stored in obfuscated form using sha-256 or sha-512.** • **Authentication data for public key-based authentication methods are stored in a directory owned by the user (and typically with the same name as the user). This directory contains the files '.ssh/authorized_keys' and '.ssh/authorized_keys2' which are used for SSH public key authentication.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.2 FPT_FLS.1/SelfTest

5.10.2.1 FPT_FLS.1/SelfTest TSS

Objective	The evaluator shall ensure the TSS describes how the TOE ensures a shutdown upon a self-test failure, a failed integrity check of the TSF executable image, or a failed health test of the noise source. If there are instances when a shut-down does not occur, (e.g., a failure is deemed non-security relevant), the evaluator shall ensure that those cases are identified and a rationale is provided that supports the classification and justificaties why the TOE's ability to enforce its security policies is not affected in any such instance.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the TOE ensures a shutdown upon a self-test failure, a failed integrity check of the TSF executable image, or a failed health test of the noise source. Upon investigation, the evaluator found that the TSS states that: **The TOE will run the following set of self-tests during power on to check the correct operation of the TOE:** • **Power on test – determines the boot-device responds and performs a memory size check to confirm the amount of available memory.** • **File integrity test –verifies integrity of all mounted signed packages, to assert that system files have not been tampered with. To test the integrity of the firmware,**

	the fingerprints of the executables and other immutable files are regenerated and validated against the SHA1 fingerprints contains in the manifest file. • **Crypto integrity test – checks integrity of major CSPs, such as SSH hostkeys and iked credentials, such as Cas, CERTS, and various keys.** • **Authentication error – verifies that veriexec is enabled and operates as expected using /opt/sbin/kats/cannot-exec.real.** • **Kernel, libmd, OpenSSL, QuickSec, SSH IPsec – verifies correct output from known answer tests for appropriate algorithms.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.2.2 FPT_FLS.1/SelfTest Guidance

Objective	The evaluator shall verify that the operational guidance provides information on the self-test failures that can cause the TOE to shut down and how to diagnose the specific failure that has occurred, including possible remediation steps if available.
Evaluator Findings	The evaluator examined the section titled **Performing Self-Tests on a Device** in the AGD to verify that it provides information on the self-test failures that can cause the TOE to shut down and how to diagnose the specific failure that has occurred. Upon investigation, the evaluator found that the AGD states that: **When a KAT self-test fails, a log message is written to the system log messages file with details of the test failure. Then the system panics and reboots**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.3 FPT_SKP_EXT.1

5.10.3.1 FPT_SKP_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details how any preshared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS shall describe how they are protected/obscured.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** and **Table 19** in the Security Target to verify that the TSS details how any pre-shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose. Upon investigation, the evaluator found that the TSS states that: **Junos OS does not provide a CLI interface to permit the viewing of keys. Cryptographic keys are protected through the enforcement of kernel-level file access rights, limiting access to the contents of cryptographic key containers to processes with cryptographic rights or shell users with root permission.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.4 FPT_STM_EXT.1

5.10.4.1 FPT_STM_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to ensure that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS lists each security function that makes use of time and provides a description of how the time is maintained and considered reliable in the context of each of the time related functions. Upon investigation, the evaluator found that the TSS states that: **All events recorded by syslog are timestamped. The clock function of Junos OS provides a source of date and time information for the appliance, used in audit timestamps. The clock is also used to determine certificate expiration, administrator session timeouts, and IPsec/SSH rekey thresholds. Wind River Linux kernel provides the current time when it bootstraps the Junos OS VM. Once the Junos OS VM is started it maintains its own time using the hardware Time Stamp Counter as the clock source.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.4.2 FPT_STM_EXT.1 Guidance 1

Objective	The evaluator examines the guidance documentation to ensure it instructs the administrator how to set the time. If the TOE supports the use of an NTP server, the guidance documentation instructs how a communication path is established between the TOE and the NTP server, and any configuration of the NTP client on the TOE to support this communication.
Evaluator Findings	The evaluator examined the section titled **Configuring the time and date** in the AGD to verify that it instructs the administrator how to set the time. Upon investigation, the evaluator found that the AGD states that: **To configure a system date and time, use the following command:** **[edit]** **user@host> set date YYYYMMDDHHMM.ss** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.5 FPT_TST_EXT.1.1

5.10.5.1 FPT_TST_EXT.1.1 TSS 1

Objective	The evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" shall be used). The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.

Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details the self-tests that are run by the TSF on start-up. Upon investigation, the evaluator found that the TSS states that: **The TOE will run the following set of self-tests during power on to check the correct operation of the TOE:** • **Power on test – determines the boot-device responds and performs a memory size check to confirm the amount of available memory.** • **File integrity test –verifies integrity of all mounted signed packages, to assert that system files have not been tampered with. To test the integrity of the firmware, the fingerprints of the executables and other immutable files are regenerated and validated against the SHA1 fingerprints contains in the manifest file.** • **Crypto integrity test – checks integrity of major CSPs, such as SSH hostkeys and iked credentials, such as Cas, CERTS, and various keys.** • **Authentication error – verifies that veriexec is enabled and operates as expected using /opt/sbin/kats/cannot-exec.real.** • **Kernel, libmd, OpenSSL, QuickSec, SSH IPsec – verifies correct output from known answer tests for appropriate algorithms.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly. Upon investigation, the evaluator found that the TSS states that: **Within the package, each Junos OS firmware image includes fingerprints of executables and other immutable files. Junos firmware will not execute any binary without validating a registered fingerprint. This feature protects the system against unauthorized software and activity that might compromise the integrity of the device. These self-tests ensure that only authorized executables are allowed to run thus ensuring the correct operation of the TOE**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.5.2 FPT_TST_EXT.1.1 Guidance 1

Objective	The evaluator shall also ensure that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.
Evaluator Findings	The evaluator examined the section titled **Performing Self-Tests on a Device** in the AGD to verify that it describes the possible errors that may result from such tests, and actions the administrator should take in response. Upon investigation, the evaluator found that the AGD states that: **The KAT self-tests are performed automatically at startup and reboot, regardless of whether FIPS mode is enabled on the NFX350 device. Conditional self-tests are also performed automatically to verify digitally signed software packages, generated random numbers, RSA and DSA key pairs, and manually entered keys.** **If the KATs are completed successfully, the system log (syslog) file is updated to display the tests that were executed.**

	If one of the KATs fail, the device panics and reboot continuously. The device can be recovered using USB install. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.6 FPT_TST_EXT.3

5.10.6.1 FPT_TST_EXT.3 TSS

Objective	The evaluator verifies that the TSS describes the method used to perform self-testing on the TSF executable code, and that this method is consistent with what is described in the SFR.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the method used to perform self-testing on the TSF executable code, and that this method is consistent with what is described in the SFR. Upon investigation, the evaluator found that the TSS states that: **The TOE will run the following set of self-tests during power on to check the correct operation of the TOE:** • **File integrity test –verifies integrity of all mounted signed packages, to assert that system files have not been tampered with. To test the integrity of the firmware, the fingerprints of the executables and other immutable files are regenerated and validated against the SHA1 fingerprints contains in the manifest file**. • **Crypto integrity test – checks integrity of major CSPs, such as SSH hostkeys and iked credentials, such as Cas, CERTS, and various keys.** **These self-tests ensure that only authorized executables are allowed to run thus ensuring the correct operation of the TOE.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7 FPT_TUD_EXT.1

5.10.7.1 FPT_TUD_EXT.1 TSS 1

Objective	The evaluator shall verify that the TSS describe how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS needs to describe how and when the inactive version becomes active. The evaluator shall verify this description.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how to query the currently active version. Upon investigation, the evaluator found that the TSS states that: **Security Administrators are able to query the current version of the TOE firmware using the CLI command "show version" and, if a new version of the TOE firmware is available, initiate an update of the TOE firmware.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS, if a trusted update can be installed on the TOE with a delayed activation, describes how and when the inactive version becomes active. Upon investigation, the evaluator found that the TSS states that:

	Junos OS does not provide partial updates for the TOE, customers requiring updates must migrate to a subsequent release. Updates are downloaded and applied manually (there is no automatic updating of the Junos OS). Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7.2 FPT_TUD_EXT.1 TSS 2

Objective	The evaluator shall verify that the TSS describes all TSF software update mechanisms for updating the system firmware and software (for simplicity the term 'software' will be used in the following although the requirements apply to firmware and software). The evaluator shall verify that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS shall detail this mechanism instead of the digital signature verification mechanism. The evaluator shall verify that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes all TSF software update mechanisms for updating the system software, includes a digital signature verification of the software before installation and that installation fails if the verification fails. Upon investigation, the evaluator found that the TSS states that: **The installable software package has a digital signature that is checked when the Security Administrator attempts to install the package.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification. Upon investigation, the evaluator found that the TSS states that: **The firmware is digitally signed. The signature of the complete package is verified at the beginning of the installation process before the package is expanded. If signature verification fails, an error message is displayed and the package is not installed.** **The Junos OS kernel maintains a set of fingerprints (SHA1 digests) for executable files and other files which should be immutable, as described in Section 7.3. The manifest file is signed using the Juniper package signing key and is verified by the TOE using the public key (stored on the TOE filesystem in clear, protected by filesystem access rights). ECDSA (P-256) with SHA-256 is used for digital signature package verification.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7.3 FPT_TUD_EXT.1 TSS 3

Objective	If the options 'support automatic checking for updates' or 'support automatic updates' are chosen from the selection in FPT_TUD_EXT.1.2, the evaluator shall verify that the TSS explains

	what actions are involved in automatic checking or automatic updating by the TOE, respectively.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS, if the options 'support automatic checking for updates' or 'support automatic updates' are chosen, explains what actions are involved in automatic checking or automatic updating by the TOE. Upon investigation, the evaluator found that the TSS states that: **Updates are downloaded and applied manually (there is no automatic updating of the Junos OS).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7.4 FPT_TUD_EXT.1 TSS 5

Objective	If a published hash is used to protect the trusted update mechanism, then the evaluator shall verify that the trusted update mechanism does involve an active authorization step of the Security Administrator, and that download of the published hash value, hash comparison and update is not a fully automated process involving no active authorization by the Security Administrator. In particular, authentication as Security Administration according to FMT_MOF.1/ManualUpdate needs to be part of the update process when using published hashes.
Evaluator Findings	Published hash for trusted update is not applicable to the TOE.
Verdict	Pass

5.10.7.5 FPT_TUD_EXT.1 Guidance 1

Objective	The evaluator shall verify that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation needs to describe how to query the loaded but inactive version.
Evaluator Findings	The evaluator examined the section titled **Installing software on network services platform with a single routing engine** in the AGD to verify that it describes how to query the currently active version and, if a trusted update can be installed on the TOE with a delayed activation, the loaded but inactive version. Upon investigation, the evaluator found that the AGD states that: **After the reboot has completed, log in and use the show version command to verify.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7.6 FPT_TUD_EXT.1 Guidance 2

Objective	The evaluator shall verify that the guidance documentation describes how the verification of the authenticity of the update is performed (digital signature verification or verification of published hash). The description shall include the procedures for successful and unsuccessful verification. The description shall correspond to the description in the TSS.

Evaluator Findings	The evaluator examined the section titled **Installing software on network services platform with a single routing engine** in the AGD to verify that it describes how the verification of the authenticity of the update is performed. Upon investigation, the evaluator found that the AGD states that: **Junos OS is delivered in signed packages that contain digital signatures to ensure the Juniper Networks software is running. When installing the software packages, Junos OS validates the signatures and the public key certificates used to digitally sign the software packages. If the signature or certificate is found to be invalid (for example, when the certificate validity period has expired or cannot be verified against the root CA stored in the Junos OS internal store), the installation process fails.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.10.7.7 FPT_TUD_EXT.1 Guidance 3

Objective	If a published hash is used to protect the trusted update mechanism, the evaluator shall verify that the guidance documentation describes how the Security Administrator can obtain authentic published hash values for the updates.
Evaluator Findings	AGD does not describe updates based on published hash.
Verdict	Pass

5.10.7.8 FPT_TUD_EXT.1 Guidance 6

Objective	If this was information was not provided in the TSS: If the ST author indicates that a certificate-based mechanism is used for software update digital signature verification, the evaluator shall verify that the Guidance Documentation contains a description of how the certificates are contained on the device. The evaluator also ensures that the Guidance Documentation describes how the certificates are installed/updated/selected, if necessary.
Evaluator Findings	AGD does not describe updates based on certificates.
Verdict	Pass

5.11 TSS and Guidance Activities (TOE Access)

5.11.1 FTA_SSL_EXT.1

5.11.1.1 FTA_SSL_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies whether local administrative session locking or termination is supported and the related inactivity time period settings. Upon investigation, the evaluator found that the TSS states that: **The Security Administrator can set the TOE so that a user session is terminated after a period of inactivity. For each user session Junos OS maintains a count of clock cycles (provided by the system clock) since last activity. The count is reset each time there is**

	activity related to the user session. When the counter reaches the number of clock cycles equating to the configured period of inactivity the user session is locked out. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.1.2 FTA_SSL_EXT.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period.
Evaluator Findings	The evaluator examined the section titled **Configuring the User Session Idle Timeout** in the AGD to verify that it states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period. Upon investigation, the evaluator found that the AGD states that: **To configure the idle timeout for a user session, use the following command:** **[edit]** **user@host# set system login idle-timeout minutes** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.2 FTA_SSL.3

5.11.2.1 FTA_SSL.3 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details the administrative remote session termination and the related inactivity time period.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies administrative remote session termination and the related inactivity time period. Upon investigation, the evaluator found that the TSS states that: **Junos OS overwrites the display device and makes the current contents unreadable after the local interactive session is terminated due to inactivity, thus disabling any further interaction with the TOE. This mechanism is the inactivity timer for administrative sessions.** **The Security Administrator can configure this inactivity timer on administrative sessions after which the session will be logged out.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.2.2 FTA_SSL.3 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination.
Evaluator Findings	The evaluator examined the section titled **Configuring the User Session Idle Timeout** in the AGD to verify that it includes instructions for configuring the inactivity time period for remote administrative session termination. Upon investigation, the evaluator found that the AGD states that:

	To configure the idle timeout for a user session, use the following command: **[edit]** **user@host# set system login idle-timeout minutes** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.3 FTA_SSL.4

5.11.3.1 FTA_SSL.4 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details how the local and remote administrative sessions are terminated.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies details how the local and remote administrative sessions are terminated. Upon investigation, the evaluator found that the TSS states that: **User sessions (local and remote) can be terminated by users. The administrative user can logout of existing session by typing logout to exit the CLI admin session and the Junos OS makes the current contents unreadable after the admin initiates the termination. No user activity can take place until the user re-identifies and authenticates**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.3.2 FTA_SSL.4 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation states how to terminate a local or remote interactive session.
Evaluator Findings	The evaluator examined the section titled **Login and Logout Events Using SSH** in the AGD to verify that it states how to terminate a local or remote interactive session. Upon investigation, the evaluator found that the AGD states that the 'quit' command is used to terminate local and remote interactive sessions. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.11.4 FTA_TAB.1

5.11.4.1 FTA_TAB.1 TSS 1

Objective	The evaluator shall check the TSS to ensure that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS). The evaluator shall check the TSS to ensure that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS details each administrative method of access available to the Security Administrator and states that the TOE is displaying an advisory notice and consent warning

	message for each administrative method of access. Upon investigation, the evaluator found that the TSS states that: **Junos enables Security Administrators to configure an access banner provided with the authentication prompt. The banner can provide warnings against unauthorized access to the secure switch as well as any other information that the Security Administrator wishes to communicate. The banner is shown at the local console and during remote access sessions using SSH.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.11.4.2 FTA_TAB.1 Guidance 1

Objective	The evaluator shall check the guidance documentation to ensure that it describes how to configure the banner message.
Evaluator Findings	The evaluator examined the section titled **Configuring a System Login Message and Announcement** in the AGD to verify that it describes how to configure the banner message. Upon investigation, the evaluator found that the AGD states that: **A login message appears before the user logs in and a system login announcement appears after the user logs in. By default, no login message or announcement is displayed on the device.** **To configure a system login message through console or management interface, use the following command:** **[edit]** **user@host# set system login message login-message-banner-text** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12 TSS and Guidance Activities (Trusted Path/Channels)

5.12.1 FTP_ITC.1

5.12.1.1 FTP_ITC.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. The evaluator shall also confirm that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of

	assured identification of the non-TSF endpoint. Upon investigation, the evaluator found that the TSS states that: **Junos OS provides an SSH server to support Trusted Channels using SSHv2 protocol which ensures the confidentiality and integrity of communication with the remote audit server.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes all secure communication mechanisms in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST. Upon investigation, the evaluator found that the TSS states that: **Export of audit information to a secure, remote server is achieved by setting up an event trace monitor that sends event log messages by using NETCONF over SSH to the remote system event logging server. The remote audit server initiates the connection. The SSHv2 protocol ensures that the data transmitted over an SSH session cannot be disclosed or altered by using the encryption and integrity mechanisms of the protocol with the FIPS cryptographic module.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12.1.2 FTP_ITC.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken.
Evaluator Findings	The evaluator examined the section titled **Configuring the Remote Syslog Server** in the AGD to verify that it contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken. Upon investigation, the evaluator found that the AGD states that AGD provide configuration instruction for configuring connections with each authorized IT entity. Specifically, the evaluator found that AGD provides guidance for configuring connections with a syslog server. Next, the evaluator reviewed AGD and found that for each connection a description of how to recover from unintentional disconnections. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12.2 FTP_ITC.1/VPN

5.12.2.1 FTP_ITC.1/VPN TSS 1

Objective	The evaluation activities specified for FTP_ITC.1 in the Supporting Document for the Base-PP shall be applied for IPsec VPN communications. From FTP_ITC.1: The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. The evaluator shall also confirm that all secure communication mechanisms are described in

	sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. Upon investigation, the evaluator found that the TSS states that: **The TOE identifies as a multi-site VPN gateway. It has a dedicated IPsec VPN interface that only supports tunnel mode.** **NFX350 supports numerous routing standards as well as IPSec protocols** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes all secure communication mechanisms in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST. Upon investigation, the evaluator found that the TSS states that: **The TOE identifies as a multi-site VPN gateway. It has a dedicated IPsec VPN interface that only supports tunnel mode.** **NFX350 supports numerous routing standards as well as IPSec protocols. These functions can all be managed through the Junos OS software, either from a connected console on the management interface or via a network connection. Network management can be secured using IPsec (SSH is covered in FTP_ITC.1).** **Secure communication mechanism includes inbound and outbound traffic. For inbound traffic, the TOE looks up the SA by using the destination IP address, security protocol, and security parameter index (SPI) value. For outbound VPN traffic, the policy invokes the SA associated with the VPN tunnel.** **For allowed protocols the TOE supports AES-GCM-128, AES-GCM-192 and AES-GCM-256, and AES-CBC-128, AES-CBC-192 or AES-CBC-256 using HMAC SHA-256 for ESP protection.** **The TOE supports AES-CBC-128, AES-CBC-192, and AES-CBC-256 for payload protection in IKEv1 and IKEv2. The TOE also supports AES-GCM-128 and AES-GCM-256 for the payload protection in IKEv2.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12.2.2 FTP_ITC.1/VPN Guidance 1

Objective	The evaluation activities specified for FTP_ITC.1 in the Supporting Document for the Base-PP shall be applied for IPsec VPN communications. From FTP_ITC.1: The evaluator shall confirm that the guidance documentation contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken.
Evaluator Findings	The evaluator examined the section titled **Configuring the Remote Syslog Server** in the AGD to verify that it contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be

	unintentionally broken. Upon investigation, the evaluator found that the AGD states that AGD provide configuration instruction for configuring connections with each authorized IT entity. Specifically, the evaluator found that AGD provides guidance for configuring connections with a syslog server. Next, the evaluator reviewed AGD and found that for each connection a description of how to recover from unintentional disconnections. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12.3 FTP_TRP.1/Admin

5.12.3.1 FTP_TRP.1/Admin TSS 1

Objective	The evaluator shall examine the TSS to determine that the methods of remote TOE administration are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of TOE 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 titled **TOE Summary Specification** in the Security Target to verify that the TSS indicates the methods of remote TOE administration and how those communications are protected. Upon investigation, the evaluator found that the TSS states that: **The Junos OS SSH Server supports Trusted Paths using SSHv2 protocol which ensures the confidentiality and integrity of user sessions. The encrypted communication path between Junos OS SSH Server and a remote administrator is provided by the use of an SSH session. Remote administrators of Junos OS initiate communication to the Junos CLI through the SSH tunnel created by the SSH session.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS protocols are consistent with those specified in the requirement. Upon investigation, the evaluator found that the TSS states that: **Assured identification of Junos OS is guaranteed by using public key based authentication for SSH. The SSHv2 protocol ensures that the data transmitted over an SSH session cannot be disclosed or altered by using the encryption and integrity mechanisms of the protocol with the FIPS cryptographic module.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.12.3.2 FTP_TRP.1/Admin Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.
Evaluator Findings	The evaluator examined the section titled **Configuring SSH and Console Connection** in the AGD to verify that it contains instructions for establishing the remote administrative sessions for each supported method. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13 TSS and Guidance Activities (Intrusion Prevention)

5.13.1 IPS_ABD_EXT.1

5.13.1.1 IPS_ABD_EXT.1 TSS

Objective	The evaluator shall verify that the TSS describes the composition, construction, and application of baselines or anomaly-based attributes specified in IPS_ABD_EXT.1.1. The evaluator shall verify that the TSS provides a description of how baselines are defined and implemented by the TOE, or a description of how anomaly-based rules are defined and configured by the administrator. The evaluator shall verify that each baseline or anomaly-based rule can be associated with a reaction specified in IPS_ABD_EXT.1.3. The evaluator shall verify that the TSS identifies all interface types capable of applying baseline or anomaly-based rules and explains how they are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., Page 18 of 47 Activity Assurance Activity where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes the composition, construction, and application of baselines or anomaly-based attributes, and provides a description of how baselines are defined and implemented by the TOE, or a description of how anomaly-based rules are defined and configured by the administrator. Upon investigation, the evaluator found that the TSS states that: **The TOE allows administrators to define signatures for anomalous traffic in terms of throughput (bits per second), time of the day for defined source/destination address and source/destination port, frequency of traffic patterns and thresholds of traffic patterns.** **Anomaly signatures based on time of day characteristics are implemented by configuring schedulers using the Junos command 'set schedulers' and attaching them to firewall policies, which in turn specify the target traffic in terms of IP addresses and port numbers as well as the action to be perform on signature triggering (allow or block/drop traffic).** **Anomaly signatures based on throughput characteristics are implemented by configuring policers with a bandwidth limit and the desired signature action (discard or forward), using the Junos command 'set firewall policer', and attaching it to any interface with the Junos command 'set interfaces'. Traffic exceeding the specified throughput limit is dropped when the policer is configured to discard traffic. A policer can be applied to specific inbound or outbound IP packets in a Layer 3 traffic flow at a logical interface by using a stateless firewall filter. If an input firewall filter is configured on the same logical interface as a policer, the policer is executed first. If an output firewall filter is configured on the same logical interface as a policer, the firewall filter is executed first.** The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies all interface types capable of applying baseline or anomaly-based rules and explains how they are associated with distinct network interfaces. Upon investigation, the evaluator found that the TSS states that: **A policer can be applied to specific inbound or outbound IP packets in a Layer 3 traffic flow at a logical interface by using a stateless firewall filter. If an input firewall filter is configured on the same logical interface as a policer, the policer is executed first. If an output firewall**

	filter is configured on the same logical interface as a policer, the firewall filter is executed first. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.1.2 IPS_ABD_EXT.1 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions to manually create baselines or anomaly-based rules according to the selections made in IPS_ABD_EXT.1.1. Note that dynamic "profiling" of a network to establish a baseline is outside the scope of this PP. The evaluator shall verify that the operational guidance provides instructions to associate reactions specified in IPS_ABD_EXT.1.3 with baselines or anomaly-based rules. The evaluator shall verify that the operational guidance provides instructions to associate the different policies with distinct network interfaces.
Evaluator Findings	The evaluator examined the section titled **IDP Extended Package Configuration Overview** in the AGD to verify that it provides instructions to manually create baselines or anomaly-based rules according to the selections made in IPS_ABD_EXT.1.1, associate reactions specified in IPS_ABD_EXT.1.3 with baselines or anomaly-based rules, and associate the different policies with distinct network interfaces. Upon investigation, the evaluator found that the AGD states the CLI commands to create baseline or anomaly-based rules in accordance with the selections made in the SFR. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.2 IPS_IPB_EXT.1

5.13.2.1 IPS_IPB_EXT.1 TSS

Objective	The evaluator shall verify how good/bad lists affect the way in which traffic is analyzed with respect to processing packets. The evaluator shall also very that the TSS provides details for the attributes that create a known good list, a known bad list, and their associated rules, including how to define the source or destination IP address (e.g. a single IP address or a range of IP addresses). If the TSF uses address types other than a single IP or a range of IP addresses (e.g. MAC addresses), the evaluator shall check that the TSS explains what configurations would cause non-IP lists of known-good and known-bad addresses to take precedence over IP-based address lists. The evaluator shall also verify that the TSS identifies all the roles and level of access for each of those roles that have been specified in the requirement.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how good/bad lists affect the way in which traffic is analyzed with respect to processing packets, provides details for the attributes that create a known good list, a known bad list, and their associated rules, including how to define the source or destination IP address, explains what configurations would cause non-IP lists of known-good and known-bad addresses to take precedence over IP-based address lists and identifies all the roles and level of access for each of those roles that have been specified in the requirement. Upon investigation, the evaluator found that the TSS states that:

	The TOE supports the definition of known-good and known-bad lists of source and/or destination addresses at the firewall rule level. Address ranges can be defined by creating address book entries and attaching them to firewall policies. Non-IP lists of known-good and known-bad addresses aren't supported by the TOE. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.2.2 IPS_IPB_EXT.1 Guidance

Objective	The evaluator shall verify that the administrative guidance provides instructions with how each role specified in the requirement can create, modify and delete the attributes of a known good and known bad lists.
Evaluator Findings	The evaluator examined the section titled **Configuring Traffic Filtering Rules** in the AGD to verify that it provides instructions with how each role specified in the requirement can create, modify and delete the attributes of a known good and known bad lists. Upon investigation, the evaluator found that the AGD states the commands used to create, modify and delete the attributes of a known good and known bad lists. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.3 IPS_NTA_EXT.1

5.13.3.1 IPS_NTA_EXT.1.1 TSS

Objective	The evaluator shall verify that the TSS explains the TOE's capability of analyzing IP traffic in terms of the TOE's policy hierarchy (precedence). The TSS should identify if the TOE's policy hierarchy order is configurable by the administrator for IPS policy elements (known-good lists, known-bad lists, signature-based rules, and anomaly-based rules). Regardless of whether the precedence is configurable, the evaluator shall verify that the TSS describes the default precedence as well as the IP analyzing functions supported by the TOE. The TSS associated with this requirement is assessed in the subsequent assurance activities.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS explains the TOE's capability of analyzing IP traffic in terms of the TOE's policy hierarchy, identifies if the TOE's policy hierarchy order is configurable, and describes the default precedence. Upon investigation, the evaluator found that the TSS states that: **The Junos OS Intrusion Detection and Prevention (IDP) policy enables selectively enforcing various attack detection and prevention techniques on network traffic passing through an IDP-enabled device. Policy rules can be defined to match a section of traffic based on a zone, network, and application, and then take active or passive preventive actions on that traffic.** Further TSS states that: **Once stateful firewall processing of packets has been performed by the Information Flow subsystem, if a firewall policy that has been marked for IDP processing is triggered, the packets are processed by the IPS subsystem as follows:** • **Fragmentation Processing – IP Fragments are reordered and reassembled. Duplicate, over/undersized, overlapping, incomplete and other invalid fragments are discarded.**

	• **Flow Module SSL Decryption – sessions are checked for existing IP Actions, if none exist, new sessions are created. If a destination is marked for SSL decryption, a copy of the SSL traffic will be sent to the decryption engine. The original packet will be queue until inspection is complete.**
	• **Packet Serialization and TCP Reassembly – packets are ordered and all TCP packets are reassembled into complete application messages.**
	• **Application ID – pattern matching is performed on the traffic to determine what application the traffic is. The traffic is still inspected for Attacks, even if application cannot be determined.**
	• **Protocol Decoding – protocol parsing and decoding is performed. Messages are deconstructed into application "contexts" which identify components of messages. Protocol Anomaly Detection is performed, along with AppDoS (if configured) by thresholds of these contexts.**
	• **Attack Signature Matching – signatures are detected via deterministic finite automaton (DFA) pattern matching.**
	• **IDP Attack Actions – when an attack is detected the corresponding policy configured action is executed. Possible actions include:**
	  ○ **No Action**
	  ○ **Drop packet**
	  ○ **Drop connection**
	  ○ **Close client (send an RST packet to the client)**
	  ○ **Close server (sends an RST packet to the server)**
	  ○ **Close client and server (sends an RST packet to both client and server)**
	The TOE supports stateful signature-based attack detection defined as Attack Objects. Attack Objects use context-based matching to match regular expressions in specific locations where they occur. Attack Objects can be composed of multiple signatures and protocol anomalies, including logical expressions between signatures for compound matching.
	The TOE is capable of inspecting IPv4, IPv6, ICMPv4, TCP and UPD traffic. Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.
	The TOE is capable of inspecting all traffic passing through the TOE's Ethernet interfaces (inline mode). Each of these interfaces types can be assigned to Zones on which firewall and IDP policies are predicated.
	The TOE supports the definition of known-good and known-bad lists of source and/or destination addresses at the firewall rule level. Address ranges can be defined by creating address book entries and attaching them to firewall policies.
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass

5.13.3.2 IPS_NTA_EXT.1.1 Guidance

Objective	The evaluator shall verify that the guidance describes the default precedence. If the precedence is configurable. The evaluator shall verify that the guidance explains how to configure the precedence.

Evaluator Findings	The evaluator examined the section titled **Configuring Traffic Filtering Rules** in the AGD to verify that it describes the default precedence and, if the precedence is configurable, explains how to configure the precedence. Upon investigation, the evaluator found that the AGD states that **The firewall filter terms are evaluated in the order in which they are configured**. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.3.3 IPS_NTA_EXT.1.2 TSS

Objective	The evaluator shall verify that the TSS indicates that the following protocols are supported: • IPv4 • IPv6 • ICMPv4 • ICMPv6 • TCP • UDP The evaluator shall verify that the TSS describes how conformance with the identified protocols has been determined by the TOE developer. (e.g., third party interoperability testing, protocol compliance testing).
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS indicates that the required protocols are supported and describes how conformance with the identified protocols has been determined by the TOE developer. Upon investigation, the evaluator found that the TSS states that: **The TOE is capable of inspecting IPv4, IPv6, ICMPv4, ICMPv6, TCP and UDP traffic. Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.3.4 IPS_NTA_EXT.1.2 Guidance

Objective	The Guidance associated with this requirement is assessed in the subsequent assurance activities.
Evaluator Findings	The Guidance associated with this requirement is assessed in the subsequent assurance activities.
Verdict	Pass

5.13.3.5 IPS_NTA_EXT.1.3 TSS

Objective	The evaluator shall verify that the TSS identifies all interface types capable of being deployed in the modes of promiscuous, and or inline mode as well as the interfaces necessary to facilitate each deployment mode (at a minimum, the interfaces need to support inline mode). The TSS should also provide descriptions how the management interface is distinct from sensor interfaces.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS identifies all interface types capable of being deployed in the modes of promiscuous and or inline mode as well as the interfaces necessary to facilitate each

	deployment mode and describes how the management interface is distinct from sensor interfaces. Upon investigation, the evaluator found that the TSS states that: **The TOE is capable of inspecting all traffic passing through the TOE's Ethernet interfaces (inline mode). Each of these interfaces types can be assigned to Zones on which firewall and IDP policies are predicated.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.3.6 IPS_NTA_EXT.1.3 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions on how to deploy each of the deployment methods outlined in the TSS. The evaluator shall also verify that the operational guidance provides instructions of applying IPS policies to interfaces for each deployment mode. If the management interface is configurable the evaluator shall verify operational guidance explains how to configure the interface into a management interface. The evaluator shall verify that the operational guidance explains how the TOE sends commands to remote traffic filtering devices. Note: the secure channel configurations between the TOE and the remote device would be discussed as per FTP_ITC.1 (if the ST author selects other interface types) and/or FTP_TRP.1 (for interfaces in management mode) in the base PP.
Evaluator Findings	The evaluator examined the section titled **IDP Extended Package Configuration Overview** in the AGD to verify that it provides instructions on how to deploy each of the deployment methods outlined in the TSS; applying IPS policies to interfaces for each deployment mode; and, if the management interface is configurable, explains how to configure the interface into a management interface. Upon investigation, the evaluator found that the AGD states the steps for each deployment method along with the steps for applying IPS policies to interfaces for each deployment mode and explains how to configure the interface into a management interface. The TOE does not support sending traffic to remote traffic filtering devices. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.4 IPS_SBD_EXT.1

5.13.4.1 IPS_SBD_EXT.1.1 TSS

Objective	The evaluator shall verify that the TSS describes what is comprised within a signature rule. The evaluator shall verify that each signature can be associated with a reaction specified in IPS_SBD_EXT.1.5. The evaluator shall verify that the TSS identifies all interface types capable of applying signatures and explains how rules are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes what is comprised within a signature rule, states that each

	signature can be associated with a reaction specified in IPS_SBD_EXT.1.5, and identifies all interface types capable of applying signatures and explains how rules are associated with distinct network interfaces. Upon investigation, the evaluator found that the TSS states that: **The TOE supports stateful signature-based attack detection defined as Attack Objects. Attack Objects use context-based matching to match regular expressions in specific locations where they occur.** **The TOE supports inspection of the following packet header information:** • **IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options.** • **IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options.** • **ICMP: Type; Code; Header Checksum; and Rest of Header (varies based on the ICMP type and code).** • **ICMPv6: Type; Code; and Header Checksum.** • **TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.** • **UDP: Source port; destination port; length; and UDP checksum.** **Signatures can be defined to match the any of above header-field values, using the command "set security idp custom-attack", along with the actions (allow/block), using the command "set security idp idp-policy", that the TOE will perform when a match is found in the processed packets. The matching criteria can be "equal", "greater-than", "less-than" or "not-equal".** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.4.2 IPS_SBD_EXT.1.1 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions with how to create and/or configure rules using the following protocols and header inspection fields: • IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options. • IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options. • ICMP: Type; Code; Header Checksum; and Rest of Header(varies based on the ICMP type and code). • ICMPv6: Type; Code; and Header Checksum. • TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options. • UDP: Source port; destination port; length; and UDP checksum. The evaluator shall verify that the operational guidance provides instructions with how to select and/or configure reactions specified in IPS_SBD_EXT.1.5 in the signature rules.
Evaluator Findings	The evaluator examined the section titled **IDP Extended Package Configuration Overview** in the AGD to verify that it provides instructions with how to create and/or configure rules using the required protocols and header inspection fields and how to select and/or configure

	reactions specified in IPS_SBD_EXT.1.5 in the signature rules. Upon investigation, the evaluator found that the AGD states the CLI commands used to configure rules using the mentioned protocols and header inspection fields. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.4.3 IPS_SBD_EXT.1.2 TSS

Objective	The evaluator shall verify that the TSS describes what is comprised within a string-based detection signature. The evaluator shall verify that each packet payload string-based detection signature can be associated with a reaction specified in IPS_SBD_EXT.1.5.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes what is comprised within a string-based detection signature and how each packet payload string-based detection signature can be associated with a reaction specified in IPS_SBD_EXT.1.5. Upon investigation, the evaluator found that the TSS states that: **The TOE also supports string-based pattern-matching inspection of packet payload data for the above listed protocols. For TCP payload inspection, Junos OS provides pre-defined attack signatures to detect FTP commands, HTTP commands and content, and STMP states. Alternative, administrators can define custom-attack signatures for these application layer protocols using the command "set security idp custom-attack".** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.4.4 IPS_SBD_EXT.1.2 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions with how to configure rules using the packet payload string-based detection fields defined in IPS_SBD_EXT.1.2. The evaluator shall verify that the operational guidance provides instructions with how to configure reactions specified in IPS_SBD_EXT.1.5 for each string-based detection signature. The evaluator shall verify that the operational guidance provides instructions with how rules are associated with distinct network interfaces that are capable of being associated with signatures.
Evaluator Findings	The evaluator examined the section titled **Configuring the IDP Extended Package** in the AGD to verify that it provides instructions on how to configure rules using the packet payload string-based detection fields defined in IPS_SBD_EXT.1.2, provides instructions with how to configure reactions specified in IPS_SBD_EXT.1.5 for each string-based detection signature, and provides instructions with how rules are associated with distinct network interfaces that are capable of being associated with signatures. Upon investigation, the evaluator found that the AGD states that the rules can be configured to perform string-based pattern-matching when creating a custom filter. The 'pattern' option is configurable to search for a specified pattern within a packet or across multiple packets. Furthermore, the evaluator found that that the operational guidance provides instruction with how to configure reactions specified in IPS_SBD_EXT.1.5 for each string-based detection

	signature. If a string-based signature has been detected, the TOE responds based on the action specified in the 'action' setting. The evaluator found that the rules are applied to security zones which are defined by the specific interface. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.4.5 IPS_SBD_EXT.1.3 TSS

Objective	The evaluator shall verify that the TSS describes how the attacks defined in IPS_SBD_EXT.1.3 are processed by the TOE and what reaction is triggered when these attacks are identified.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the attacks defined in IPS_SBD_EXT.1.3 are processed by the TOE and what reaction is triggered when these attacks are identified. Upon investigation, the evaluator found that the TSS states that: **An IDP policy is made up of rule bases, and each rule base contains a set of rules that specify rule parameters, such as traffic match conditions, action, and logging requirements. IDP policies can then be associated to firewall policies. IDP can be invoked on a firewall rule by rule basis for maximum granularity. Only firewall policies marked for IDP will be processed by IDP engine, all other rules will only be processed by the firewall .** **Firewall Policies match Source Zone, Destination Zone, Source IP, Destination IP, Source Port, Destination Port, and Protocol. Interface and VLAN matching can be achieved through the use of zones. Rules are organized into a firewall policy rule base. Within IPS Policies, further matching for specific attacks is done on Source Zone, Destination Zone, Source IP, Destination IP, Source Port, Destination Port, and Protocol. Interface matching can be achieved through the use of zones. Attack Actions are configurable on a rule by rule basis. Rules within policies are processed in an Administrator-defined order when network traffic flows through the TOE network interfaces.** **The TOE supports stateful signature-based attack detection defined as Attack Objects. Attack Objects use context-based matching to match regular expressions in specific locations where they occur. Attack Objects can be composed of multiple signatures and protocol anomalies, including logical expressions between signatures for compound matching.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.4.6 IPS_SBD_EXT.1.3 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions with configuring rules to identify the attacks defined in IPS_SBD_EXT.1.3 as well as the reactions to these attacks as specified in IPS_SBD_EXT.1.5.
Evaluator Findings	The evaluator examined the section titled **IDP Extended Package Configuration Overview** in the AGD to verify that it provides instructions with configuring rules to identify the attacks defined in IPS_SBD_EXT.1.3 as well as the reactions to these attacks as specified in IPS_SBD_EXT.1.5. Upon investigation, the evaluator found that the AGD describes how to apply the Junos predefined screen options to the interfaces using the command "set security screen ids-option" or through custom-attack signatures using 'set security idp custom-attack'. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.13.4.7 IPS_SBD_EXT.1.4 TSS

Objective	The evaluator shall verify that the TSS describes how the attacks defined in IPS_SBD_EXT.1.4 are processed by the TOE and what reaction is triggered when these attacks are identified.
Evaluator Findings	The evaluator examined the section titled **TOE Summary Specification** in the Security Target to verify that the TSS describes how the attacks defined in IPS_SBD_EXT.1.4 are processed by the TOE and what reaction is triggered when these attacks are identified. Upon investigation, the evaluator found that the TSS states that: **The TOE is capable of detecting the following signatures using Junos predefined screen options:** **MOD_IPS signature name** / **Junos screen name** **IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)** / **ip tear-drop** **IP source address equal to the IP destination (Land attack)** / **tcp land** **Fragmented ICMP Traffic (e.g. Nuke attack)** / **icmp fragment** **Large ICMP Traffic (Ping of Death attack)** / **icmp ping-death** **TCP NULL flags** / **tcp tcp-no-flag** **TCP SYN+FIN flags** / **tcp syn-fin** **TCP FIN only flags** / **tcp fin-no-ack** **UDP Bomb Attack** / **udp length-error** **ICMP flooding (Smurf attack, and ping flood)** / **icmp flood** **TCP flooding (e.g. SYN flood)** / **tcp syn-flood** **IP protocol scanning** / **ip unknown-protocol** **TCP port scanning** / **tcp port-scan** **UDP port scanning** / **udp port-scan** **ICMP scanning** / **icmp ip-sweep** **The default action for the above screens is to drop the packets. To allow the packets through, the "alarm-without-drop" action can be defined using the command "set security screen ids-option".** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.13.4.8 IPS_SBD_EXT.1.4 Guidance

Objective	The evaluator shall verify that the operational guidance provides instructions with configuring rules to identify the attacks defined in IPS_SBD_EXT.1.4 as well as the reactions to these attacks as specified in IPS_SBD_EXT.1.5.
Evaluator Findings	The evaluator examined the section titled **Configuring Network Attacks** in the AGD to verify that it provides instructions with configuring rules to identify the attacks defined in

	IPS_SBD_EXT.1.4 as well as the reactions to these attacks as specified in IPS_SBD_EXT.1.5. Upon investigation, the evaluator found that the AGD states the CLI commands needed to configure rules to identify attacks defined in IPS_SBD_EXT.1.4 along with the reactions to these attacks. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.13.4.9 IPS_SBD_EXT.1.6 Guidance

Objective	The evaluator shall verify that the operational guidance provides configuration instructions, if needed, to detect payload across multiple packets.
Evaluator Findings	No separate configuration is needed for detection of payloads across multiple packets since it is covered by the custom signature or custom attack. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6 Detailed Test Cases (Test Activities)

Testing Time and Location:

All testing was carried at the Acumen Security offices located in 2400 Research Blvd Suite #395, Rockville, MD 20850. Testing occurred November 2020 through March 2022. Regression testing and sample re-runs were performed in January – March 2022. Testing was performed on Junos OS 20.3R1.8 and regression testing was performed on Junos OS 20.3R3. The firmware release for CC evaluation was upgraded from 20.3R1.8 to 20.3R3-S3.1 to ensure that there are no open vulnerabilities and to ensure that all bug fixes are addressed in the evaluated version of the TOE firmware. Following tests were performed as part of regression testing:

- FAU_STG_EXT.1 Test #1
- FPT_STM_EXT.1.1 Test #1
- FIA_PMG_EXT.1.1 Test#2
- FCS_IPSEC_EXT.1.14 Test#5
- FCS_IPSEC_EXT.1.14 Test#6a
- FCS_SSHS_EXT.1.3 Test #1
- FCS_SSHS_EXT.1.4 Test #1
- FPT_TUD_EXT.1 Test #1
- FPT_TUD_EXT.1 Test #2a
- FIA_X509_EXT.1.1/Rev Test #8a
- FIA_X509_EXT.1.1/Rev Test #8b
- FIA_X509_EXT.1.1/Rev Test #8c
- FPF_RUL_EXT.1.1 Test #1
- FPF_RUL_EXT.1.1 Test #2
- FIA_PSK_EXT.1 Test #1
- FIA_PSK_EXT.1 Test #3
- FFW_RUL_EXT.1.7 Test #1
- FFW_RUL_EXT.1.7 Test #2

Testing was performed within their Common Criteria lab in a controlled, isolated environment and completed in March 2022 by the Acumen Security Evaluation Team following the CCTL's NVLAP-accredited test procedures.

6.1 FAU_GEN.1 Test #1

Item	Data / Description
Test ID	FAU_GEN.1 Test #1
Objective	The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the table of audit events and administrative actions listed above. This should include all instances of an event: for instance, if there are several different I&A mechanisms for a system, the FIA_UIA_EXT.1 events must be generated for each mechanism. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. If HTTPS is implemented, the test demonstrating the establishment and termination of a TLS session can be combined with the test for an HTTPS session. When verifying the test results, the evaluator shall ensure the audit records generated during

	testing match the format specified in the guidance documentation, and that the fields in each audit record have the proper entries. Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.
Test Execution Steps	**Covered by audit records in each test case.**
Pass/Fail with explanation	Pass. Covered by audit records in each test case.

6.2 FAU_STG_EXT.1 Test #1

Item	Data / Description
Test ID	FAU_STG_EXT.1 Test #1
Objective	Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE during several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing. The evaluator shall verify that the TOE is capable of transferring audit data to an external audit server automatically without administrator intervention.
Test Execution Steps	• Record the audit server name and version information. • Configure the TOE to communicate with a syslog server by generating an ECDSA public key on the remote syslog server. • On the TOE, create a class named monitor that has permission to trace events. • On the TOE, create a user named syslog-mon with the class monitor and with ECDSA public key authentication. • On the TOE, configure NETCONF with SSH. • The TOE logs that NETCONF client was used. • Verify the traffic between the TOE and syslog is not sent in plaintext.
Pass/Fail with explanation	Pass. The TOE is capable of transferring audit data to an external audit server automatically without administrator intervention.

6.3 FAU_STG_EXT.1 Test #2 (b)

Item	**Data / Description**
Test ID	FAU_STG_EXT.1 Test #2 (b)
Objective	Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that:

	1) The audit data remains unchanged with every new auditable event that should be tracked but that the audit data is recorded again after the local storage for audit data is cleared (for the option 'drop new audit data' in FAU_STG_EXT.1.3) 2) The existing audit data is overwritten with every new auditable event that should be tracked according to the specified rule (for the option 'overwrite previous audit records' in FAU_STG_EXT.1.3).
Test Execution Steps	• Configure smallest possible logging space. • Find the timestamp of the oldest message in the local audit log. • Generate lots of audit records. • Find the timestamp of the oldest message in the local audit log.
Pass/Fail with explanation	Pass. When audit data is filled to the max, the existing audit data is overwritten.

6.4 FPT_STM_EXT.1.1 Test #1

Item	Data / Description
Test ID	FPT_STM_EXT.1.1 Test #1
Objective	Test 1: If the TOE supports direct setting of the time by the Security Administrator then the evaluator uses the guidance documentation to set the time. The evaluator shall then use an available interface to observe that the time was set correctly.
Test Execution Steps	• Verify current time via remote console over SSH • Set new time via remote console over SSH • Verify the time on the TOE was updated • Verify logs were generated for time change
Pass/Fail with explanation	Pass. The TOE allows the administrative user to configure the time on the TOE. This meets the testing requirements.

6.5 FTP_ITC.1 Test #1

Item	Data / Description
Test ID	FTP_ITC.1 Test #1
Objective	The evaluators shall ensure that communications using each protocol with each authorized IT entity is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.
Pass/Fail with explanation	Pass. This testing is covered by the requirements in FAU_STG_EXT.1 for SSH to the syslog server and in FCS_IPSEC_EXT.1.1 for VPN communications.

6.6 FTP_ITC.1 Test #2

Item	Data / Description
Test ID	FTP_ITC.1 Test #2

Objective	Test 2: For each protocol that the TOE can initiate as defined in the requirement, the evaluator shall follow the guidance documentation to ensure that in fact the communication channel can be initiated from the TOE.
Pass/Fail with explanation	Pass. This test is performed in conjunction with the tests associated with FTP_ITC.1 Test# 1, FAU_STG_EXT.1 and FCS_IPSEC_EXT.1. The TOE initiates the session to the external entity. This meets the testing requirements.

6.7 FTP_ITC.1 Test #3

Item	Data / Description
Test ID	FTP_ITC.1 Test #3
Objective	Test 3: The evaluator shall ensure, for each communication channel with an authorized IT entity, the channel data is not sent in plaintext.
Pass/Fail with explanation	Pass. This test is performed in conjunction with the tests associated with FTP_ITC.1 Test# 1, FAU_STG_EXT.1 and FCS_IPSEC_EXT.1. External connections from the TOE are sent via an encrypted channel. This meets the testing requirements.

6.8 FTP_ITC.1 Test #4

Item	Data / Description
Test ID	FTP_ITC.1 Test #4
Objective	The evaluator shall, for each instance where the TOE acts as a client utilizing a secure communication mechanism with a distinct IT entity, physically interrupt the connection of that IT entity for the following durations: 1. A duration that exceeds the TOE's application layer timeout setting, 2. A duration shorter than the application layer timeout but of sufficient length to interrupt the network link layer. The evaluator shall ensure that, when the physical connectivity is restored, communications are appropriately protected and no TSF data is sent in plaintext. In the case where the TOE is able to detect when the cable is removed from the device, another physical network device (e.g. a core switch) shall be used to interrupt the connection between the TOE and the distinct IT entity. The interruption shall not be performed at the virtual node (e.g. virtual switch) and must be physical in nature.
Test Execution Steps	• Connect to the TOE via SSH and unplug the network cable for 5 seconds. Ensure that the connection is interrupted. • Reconnect to the TOE via SSH and ensure that no data is sent in plain text. • Connect to the TOE via SSH and unplug the network cable for 1 minute. Ensure that the connection is interrupted. • Reconnect to the TOE via SSH and ensure that no data is sent in plain text. • Begin an IPSEC connection by pinging the peer device and disconnect the bridge network cable for 5 seconds. • Ensure that no data is sent in plain text during the disconnection period. • Begin an IPSEC connection by pinging the peer device and disconnect the bridge network cable for 1 minute. • Ensure that no data is sent in plain text during the disconnection period.

Pass/Fail with explanation	Pass. The TOE responds accordingly when a physical disconnection occurs. This meets the testing requirements.

6.9 FCS_CKM.2 RSA

Item	Data / Description
Test ID	FCS_CKM.2 RSA
Objective	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_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses RSAES-PKCS1-v1_5. ***TD0580 has been applied.***
Pass/Fail with explanation	NA. As per TD0580 this test has been removed.

6.10 FCS_CKM.2 DH14

Item	Data / Description
Test ID	FCS_CKM.2 DH14
Objective	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_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses Diffie-Hellman group 14. ***TD0580 has been applied.***
Pass/Fail with explanation	NA. As per TD0580 this test has been removed.

6.11 FIA_AFL.1.1

Item	Data / Description
Test ID	FIA_AFL.1.1
Objective	NDcPP: Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE (and, if the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall also use the operational guidance to configure the time period after which access is re-enabled). The evaluator shall test that once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful.

Test Execution Steps	• Set user login time out after successive unsuccessful authentication attempts. • Start a SSH session with the TOE and attempt to login 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.
Pass/Fail with explanation	Pass. The TOE did not allow authentication once the authentication attempt limit has been reached. This meets the testing requirements.

6.12 FIA_AFL.1.2

Item	Data / Description
Test ID	FIA_AFL.1.2
Objective	NDcPP: Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows. If the administrator action selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall confirm by testing that following the operational guidance and performing each action specified in the ST to re-enable the remote administrator's access results in successful access (when using valid credentials for that administrator). If the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall wait for just less than the time period configured in Test 1 and show that an authorisation attempt using valid credentials does not result in successful access. The evaluator shall then wait until just after the time period configured in Test 1 and show that an authorisation attempt using valid credentials results in successful access.
Test Execution Steps	• Set user login time out after successive unsuccessful authentication attempts. • Start a SSH session with the TOE and attempt to login with wrong password and lock the user. • Verify with logs. • Verify the user is lockout for configured time. • Wait for 300 seconds and login into the device. This should pass. • Verify that user got unlocked. • Enter valid username and password when prompted. This should succeed.
Pass/Fail Explanation	Pass. "Tester" was unlocked after lockout time of "300".

6.13 FIA_PMG_EXT.1.1 Test#1

Item	Data / Description
Test ID	FIA_PMG_EXT.1.1 Test#1
Objective	The evaluator shall compose passwords that meet the requirements in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator

	shall ensure that all characters, and a minimum length listed in the requirement are supported and justify the subset of those characters chosen for testing
Test Execution Steps	• Set the minimum password length to 10 characters. • Attempt to create 3 users (good11, good22, good33) that meet the password requirements. • Username: "good11" and Password: "Good@12345" . • Username: "good22" and Password: "gOOd!@#$%67890". • Username: "good33" and Password: "12345^&*()gooD". • Try to establish a TOE connection using all above 3 users that meet the password requirements. • Verify with the TOE logs.
Pass/Fail Explanation	Pass. The TOE was able to create users with good passwords. This meets the requirement.

6.14 FIA_PMG_EXT.1.1 Test#2

Item	Data / Description
Test ID	FIA_PMG_EXT.1.1 Test#2
Objective	b) Test 2: The evaluator shall compose passwords that do not meet the requirements in some way. For each password, the evaluator shall verify that the TOE does not support the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that the TOE enforces the allowed characters and the minimum length listed in the requirement and justify the subset of those characters chosen for testing.
Test Execution Steps	• Attempt to create 3 users (bad4, bad5, bad6) that do not meet the password requirements. The TOE did not allow the creation of these accounts as they did not meet the password length or complexity set by the TOE. • Username: "bad11" and Password: "BAD12345^&*()". • Username: "bad22" and Password: "123$%^Bad". • Username: "bad33" and Password: "1234567890bad".
Pass/Fail Explanation	Pass. The TOE was able to reject users with bad passwords. This meets the requirement.

6.15 FIA_UIA_EXT.1.1 Test #1

Item	Data / Description
Test ID	FIA_UIA_EXT.1.1 Test #1
Objective	The evaluator shall use the guidance documentation to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct I&A information results in the ability to access the system, while providing incorrect information results in denial of access.
Test Execution Steps	• Configure the user for testing login credential. ○ Attempt to log into the TOE via console with bad credentials; this should fail with TOE logs.

	○ Attempt to log into the TOE via console with good credentials; this should succeed with TOE logs. ○ Log into the TOE via SSH with bad credentials; this should fail. ○ Log into the TOE via SSH with good credentials; this should succeed. ○ Verify audit logs reflect both attempts. • Remote with public-key based authentication ○ Login to the device remotely with bad credentials. This will fail. ○ Verify the connection failed via logs. ○ Login to the device remotely with good credentials. This will succeed. ○ Verify the connection is successful via logs.
Pass/Fail with explanation	Pass. Presenting incorrect authentication credentials results in denied access to the TOE. Presenting correct authentication credentials results in access being allowed to the TOE. This meets the testing requirements.

6.16 FIA_UIA_EXT.1.1 Test #2

Item	Data / Description
Test ID	FIA_UIA_EXT.1.1 Test #2
Objective	The evaluator shall configure the services allowed (if any) according to the guidance documentation, and then determine the services available to an external remote entity. The evaluator shall determine that the list of services available is limited to those specified in the requirement.
Test Execution Steps	• Show that the Ping is allowed prior to authentication. • Show that commands are not available prior to login. • Verify authentication logs reflect failure. • Verify that only the login banner was displayed. • Login into the TOE with correct credentials. • Show that the previously disabled commands are now available. • Verify authentication logs reflect success.
Pass/Fail with explanation	Pass. No system services are available to an unauthenticated user connecting remotely. This meets the testing requirements.

6.17 FIA_UIA_EXT.1.1 Test #3

Item	Data / Description
Test ID	FIA_UIA_EXT.1.1 Test #3
Objective	Test 3: For local access, the evaluator shall determine what services are available to a local administrator prior to logging in, and make sure this list is consistent with the requirement.
Test Execution Steps	• Show that commands are not available prior to login. • Verify authentication logs reflect failure. • Verify that only the login banner was displayed. • Login into the TOE. • Show that the previously enabled commands are now available. • Verify authentication logs reflect success.

Pass/Fail with explanation	Pass. No system services are available to an unauthenticated user via the directly connected console. This meets the testing requirements.

6.18 FIA_UAU_EXT.7.1 Test #1

Item	Data / Description
Test ID	FIA_UAU_EXT.7.1 Test #1
Objective	The evaluator shall locally authenticate to the TOE. While making this attempt, the evaluator shall verify that at most obscured feedback is provided while entering the authentication information.
Test Execution Steps	• Connect to the TOE via console with correct authentication credentials. • Verifying the logs reflects for local login. • Connect to the TOE via local console with incorrect authentication credentials. • Verifying the logs reflects for local login.
Pass/Fail Explanation	Pass. At both the directly connected and remote login prompt, the TOE does not provide any feedback.

6.19 FMT_MOF.1/ManualUpdate Test #1

Item	Data / Description
Test ID	FMT_MOF.1/ManualUpdate Test #1
Objective	The evaluator shall try to perform the update using a legitimate update image without prior authentication as security administrator (either by authentication as a user with no administrator privileges or without user authentication at all – depending on the configuration of the TOE). The attempt to update the TOE shall fail.
Test Execution Steps	• Create a read only user to attempt to perform an update. • Verify that the user “test” Fail to update the TOE, as he has no options to change the settings. • Verify via logs.
Pass/Fail with explanation	Pass. Unprivileged user are not able to perform a software update on the TOE. This meets the testing requirements.

6.20 FMT_MOF.1/ManualUpdate Test #2

Item	Data / Description
Test ID	FMT_MOF.1/ManualUpdate Test #2
Objective	The evaluator shall try to perform the update with prior authentication as Security Administrator using a legitimate update image. This attempt should be successful. This test case should be covered by the tests for FPT_TUD_EXT.1 already.
Pass/Fail with explanation	Pass. This testing is covered by the requirements in FPT_TUD_EXT.1.

6.21 FMT_MOF.1/Functions (1) Test#1

Item	Data / Description
Test ID	FMT_MOF.1/Functions (1) Test#1
Objective	Test 1 (if 'transmission of audit data to external IT entity' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify all security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). Attempts to modify parameters without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to modify the security related parameters can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Execution Steps	• Log into the TOE as a lower privileged user. • Make sure that "tester" is lower privileged user. • Attempt to modify the parameters involved with the syslog server and verify the command is rejected.
Pass/Fail with explanation	Pass. The TOE does not allow an unauthenticated user to modify and delete the audit records.

6.22 FMT_MOF.1_Functions(1) Test #2

Item	Data / Description
Test ID	FMT_MOF.1_Functions(1) Test #2
Objective	Test 2 (if 'transmission of audit data to external IT entity' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify all security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity with prior authentication as Security Administrator. The effects of the modifications should be confirmed. The evaluator does not have to test all possible values of the security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity but at least one allowed value per parameter.
Pass/Fail with explanation	Pass. This testing is covered by the requirements in FAU_STG_EXT.1.

6.23 FMT_MOF.1/Functions (2) Test#1

Item	Data / Description
Test ID	FMT_MOF.1/Functions (2) Test#1
Objective	Test 1 (if 'handling of audit data' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify all security related parameters for configuration of the handling of audit data without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or

	without user authentication at all). Attempts to modify parameters without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.2, FAU_STG_EXT.1.3 and FAU_STG_EXT.2/LocSpace.
Test Execution Steps	• Log into the TOE as a lower privileged user. • Make sure that "test" is lower privileged user. • Attempt to modify the parameters involved with the syslog server and verify the command is rejected.
Pass/Fail with explanation	Pass. The TOE does not allow an unauthenticated user to modify and delete the audit records.

6.24 FMT_MOF.1_Functions(2) Test#2

Item	Data / Description
Test ID	FMT_MOF.1_Functions(2) Test#2
Objective	Test 2 (if 'handling of audit data' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify all security related parameters for configuration of the handling of audit data with prior authentication as Security Administrator. The effects of the modifications should be confirmed. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.2, FAU_STG_EXT.1.3 and FAU_STG_EXT.2/LocSpace. The evaluator does not necessarily have to test all possible values of the security related parameters for configuration of the handling of audit data but at least one allowed value per parameter.
Test Execution Steps	• Login to the TOE as an admin. • Make sure that "Tester1" is privileged user. • Attempt to modify the parameters involved with the syslog server. • Verify the logs.
Pass/Fail with explanation	Pass. The TOE allows an authenticated user to modify and delete the audit records.

6.25 FMT_MOF.1/Services Test #1

Item	Data / Description
Test ID	FMT_MOF.1/Services Test #1
Objective	The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) without prior authentication as security administrator (either by authenticating as a user with no administrator privileges, if possible, or without prior authentication at all). The attempt to enable/disable this service/these services should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to

	enable/disable this service/these services can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Execution Steps	• Start a SSH session onto the TOE with lower privilege user. • Attempt to enable and disable the services.
Pass/Fail with explanation	Pass. User without prior authentication/privilege was unable to perform actions on the TOE.

6.26 FMT_MOF.1/Services Test #2

Item	Data / Description
Test ID	FMT_MOF.1/Services Test #2
Objective	The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) with prior authentication as security administrator. The attempt to enable/disable this service/these services should be successful.
Test Execution Steps	• Start a SSH session onto the TOE with admin user. • Attempt to enable and disable the services. • Verify with audit logs.
Pass/Fail with explanation	Pass. User with prior authentication/privilege was able to perform actions on the TOE.

6.27 FMT_MTD.1/CryptoKeys Test #1

Item	Data / Description
Test ID	FMT_MTD.1/CryptoKeys Test #1
Objective	The evaluator shall try to perform at least one of the related actions (modify, delete, generate/import) without prior authentication as security administrator (either by authentication as a non-administrative user, if supported, or without authentication at all). Attempts to perform related actions without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to manage cryptographic keys can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Execution Steps	• Start a SSH session onto the TOE with non-administrator user. • Attempt to modify SSH ciphers by non-administrative user. This will fail.
Pass/Fail Explanation	Pass. Unprivileged user cannot perform security related configurations on the TOE. This meets the testing requirements.

6.28 FMT_MTD.1/CryptoKeys Test #2

Item	Data / Description

Test ID	FMT_MTD.1/CryptoKeys Test #2
Objective	The evaluator shall try to perform at least one of the related actions with prior authentication as security administrator. This attempt should be successful.
Test Execution Steps	• Start a SSH session onto the TOE with security administrator user. • Log into the TOE, attempt to modify SSH ciphers. This will succeed. • Verify via logs.
Pass/Fail with explanation	Pass. Authenticated user can perform security related configurations on the TOE. This meets the testing requirements.

6.29 FMT_SMF.1 Test #1

Item	Data / Description
Test ID	FMT_SMF.1 Test #1
Objective	The evaluator tests management functions as part of testing the SFRs identified in section 2.4.4. No separate testing for FMT_SMF.1 is required unless one of the management functions in FMT_SMF.1.1 has not already been exercised under any other SFR. **TD0631 has been applied.**
Test Execution Steps	The TSF shall be capable of performing the following management functions: • Ability to administer the TOE locally and remotely; • *Ability to configure the access banner;* • *Ability to configure the session inactivity time before session termination or locking;* • *Ability to update the TOE, and to verify the updates using digital signature and [no other] capability prior to installing those updates;* • *Ability to configure the authentication failure parameters for FIA_AFL.1;* • ***Ability to manage the cryptographic keys;*** • ***Ability to configure the cryptographic functionality;*** • ***Ability to configure the lifetime for IPsec SAs;*** • ***Ability to import X.509v3 certificates to the TOE's trust store;*** *[* • *Ability to start and stop services;* • *Ability to configure audit behavior(e.g. changes to storage locations for audit; changes to behavior when local audit storage space is full);* • *Ability to configure thresholds for SSH rekeying;* • *Ability to re-enable an Administrator account;* • *Ability to set the time which is used for time-stamps;* • *Ability to configure the reference identifier for the peer;* • *Ability to manage the TOE's trust store and designate X.509v3 certificates as trust anchors;* • *Ability to manage the trusted public keys database;*
Pass/Fail with explanation	Pass. All management functions identified have been tested throughout the evaluation. Thus, this requirement has been met.

6.30 FMT_SMR.2 Test #1

Item	Data / Description
Test ID	FMT_SMR.2 Test #1
Objective	In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this cPP be tested; for instance, if the TOE can be administered through a local hardware interface; SSH; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.
Pass/Fail with explanation	Pass. There are two interfaces where these can be tested (over the CLI and remote SSH). It is covered in FIA_UIA_EXT.1.1 Test #2, FIA_UIA_EXT.1.1 Test #3, FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3.1 Test #1, FTA_SSL.4.1 Test #1 and FTA_TAB.1 Test #1. This meets the testing requirements.

6.31 FTA_SSL.3.1 Test #1

Item	Data / Description
Test ID	FTA_SSL.3.1 Test #1
Objective	The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a remote interactive session with the TOE. The evaluator then observes that the session is terminated after the configured time period.
Test Execution Steps	• Log into the TOE via SSH. • Configure a new idle time for one minute (60 seconds). • Log out of the TOE. • Log into the TOE via SSH. • Session will be closed in 60 seconds if there is no activity. • Configure a new idle timeout for two minutes (120 seconds). • Log out of the TOE. • Log into the TOE and verify session will be closed in 120 seconds if there is no activity. • Verify that a log was created for the configured timeout period.
Pass/Fail with explanation	Pass. The remote administrative time out periods can be set by the administrative user. The TOE enforces the configured inactivity period in each instance. This meets the testing requirements.

6.32 FTA_SSL.4.1 Test #1

Item	Data / Description
Test ID	FTA_SSL.4.1 Test #1
Objective	The evaluator initiates an interactive local session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Execution Steps	• Log onto the TOE through a local administrative interface. • Verify the logs reflect log in. • Using the instructions provided by the user guide log off. • Verify the logs reflect the log off.

Pass/Fail with explanation	Pass. The TOE allows user to terminate the directly connected administrative sessions. This meets the testing requirements.

6.33 FTA_SSL.4.1 Test #2

Item	Data / Description
Test ID	FTA_SSL.4.1 Test #2
Objective	The evaluator initiates an interactive remote session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Execution Steps	• Log onto the TOE remotely via SSH. • Verify the logs reflect login. • Log out the TOE. • Verify the logs reflect that a session has been created and terminated.
Pass/Fail with explanation	Pass. The TOE allows user to terminate the remote administrative sessions. This meets the testing requirements.

6.34 FTA_SSL_EXT.1.1 Test #1

Item	Data / Description
Test ID	FTA_SSL_EXT.1.1 Test #1
Objective	The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a local interactive session with the TOE. The evaluator then observes that the session is either locked or terminated after the configured time period. If locking was selected from the component, the evaluator then ensures that reauthentication is needed when trying to unlock the session.
Test Execution Steps	• Configure a time-out period for 60 seconds. • Log out of TOE. • After logging into the TOE, wait 58 seconds and perform a command. • Verify session will be closed in 1 minute if there is no activity. • Verify that a log was created for the configured timeout. • Configure a new idle time (120 seconds). • Log out of TOE. • After logging into the TOE, wait 118 seconds and perform a command. • Verify session will be closed in 2 minutes if there is no activity. • Verify that a log was created for the configured timeout period.
Pass/Fail with explanation	Pass. The local administrative inactivity was able to be set to multiple values. In each instance, the TOE logged the user out after the configured time. This meets the testing requirements.

6.35 FTA_TAB.1.1 Test #1

Item	Data / Description
Test ID	FTA_TAB.1.1 Test #1
Objective	The evaluator follows the guidance documentation to configure a notice and consent warning message. The evaluator shall then, for each method of access specified in the TSS, establish a session with the TOE. The evaluator shall verify that the notice and consent warning message is displayed in each instance.
Test Execution Steps	• Configure access banners on TOE. • Verify that the audit records reflected the configuration steps. • Log into the TOE via SSH. • Log into the TOE via console.
Pass/Fail with explanation	Pass. An access banner can be set for all the methods that can be used to access the device. This meets the testing requirements.

6.36 FTP_TRP.1_Admin Test #1

Item	Data / Description
Test ID	FTP_TRP.1_Admin
Objective	Test 1: The evaluators shall ensure that communications using each specified (in the guidance documentation) remote administration method is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.
Test Execution Steps	• Attempt to connect to the TOE via SSH. • Verify that Wireshark shows a successful connection. • Verify that the TOE shows a successful connection.
Pass/Fail with explanation	Pass. Remote administrative access to the TOE is over secure protected channels and the data was not sent in plaintext. This meets the testing requirements.

6.37 FTP_TRP.1/Admin Test #2

Item	Data / Description
Test ID	FTP_TRP.1/Admin Test #2
Objective	The evaluator shall ensure, for each communication channel, the channel data is not sent in plaintext.
Pass/Fail with explanation	Pass. This is covered by FTP_TRP.1/Admin_T1, FCS_SSH_EXT.1 and FCS_IPSEC_EXT.1. In that test, the data was not sent in plaintext.

6.38 FCS_IPSEC_EXT.1.1 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.1 Test#1
Objective	The evaluator shall configure the SPD such that there is a rule for dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext. The selectors used in the

	construction of the rule shall be different such that the evaluator can generate a packet and send packets to the gateway with the appropriate fields (fields that are used by the rule - e.g., the IP addresses, TCP/UDP ports) in the packet header. The evaluator performs both positive and negative test cases for each type of rule (e.g. a packet that matches the rule and another that does not match the rule). The evaluator observes via the audit trail, and packet captures that the TOE exhibited the expected behavior: appropriate packets were dropped, allowed to flow without modification, encrypted by the IPsec implementation.

Test Execution Steps	• Configure three IKE/IPsec rules on the switch for connecting to an IKE/IPsec Peer.
	  ○ Allow (PROTECT) a specific type of traffic.
	  ○ Deny (DISCARD) a specific type of traffic.
	  ○ Send plaintext (BYPASS) a specific type of traffic.
	• Send traffic that will trigger each rule.
	• Capture the traffic flows to and from the device.
	• Verify that the traffic is processed as required for the configured IKE/IPsec rules.

Pass/Fail with explanation	Pass. The TOE dropped packets when configured, encrypted packets when configured, and sent packets in plaintext when configured. This meets the testing requirements.

6.39 FCS_IPSEC_EXT.1.1 Test#2

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.1 Test#2
Objective	The evaluator shall devise several tests that cover a variety of scenarios for packet processing. As with Test 1, the evaluator ensures both positive and negative test cases are constructed. These scenarios must exercise the range of possibilities for SPD entries and processing modes as outlined in the TSS and guidance documentation. Potential areas to cover include rules with overlapping ranges and conflicting entries, inbound and outbound packets, and packets that establish SAs as well as packets that belong to established SAs. The evaluator shall verify, via the audit trail and packet captures, for each scenario that the expected behavior is exhibited, and is consistent with both the TSS and the guidance documentation.

Test Execution Steps	• Configure IKE/IPsec policies meeting the following:
	  ○ Allow (PROTECT) a large set of traffic (e.g., TCP/IP, subnet).
	  ○ Deny (DISCARD) a subset of the traffic (e.g., specific protocol, specific address).
	• Send traffic that should be denied.
	• Verify that the connection cannot be established.
	
	• Configure IKE/IPsec policies meeting the following:
	  ○ Send plaintext (BYPASS) a large set of traffic (e.g., TCP/UDP, subnet).
	  ○ Allow (PROTECT) a subset of the traffic (e.g., specific protocol, specific address).
	• Send traffic that should be encrypted.
	• Verify that the connection is not in plaintext.
	• Configure IKE/IPsec policies meeting the following:
	  ○ Deny (DISCARD) a small set of the traffic (e.g., specific protocol, specific address).
	  ○ Send plaintext (BYPASS) a larger superset of traffic (e.g., TCP/UDP, subnet).
	• Send traffic that should be sent in clear text.

	• Verify that the connection is not dropped.
	
	• Configure IKE/IPsec policies meeting the following:
	  ○ Allow (PROTECT) all traffic.
	• Send various types of traffic.
	• Verify that all traffic is encrypted.
	
	• Configure IKE/IPsec policies meeting the following:
	  ○ Send plaintext (BYPASS) all traffic.
	• Send various types of traffic.
	• Verify that all traffic is sent plaintext.
	
	• Configure IKE/IPsec policies meeting the following:
	  ○ Deny (DISCARD) all traffic.
	• Send various types of traffic.
	• Verify that all traffic is dropped.

Pass/Fail with explanation	Pass. The TOE dropped packets when configured, encrypted packets when configured, and sent packets in plaintext when configured. This meets the testing requirements.

6.40 FCS_IPSEC_EXT.1.2 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.2 Test#1
Objective	The evaluator shall configure the SPD such that there is a rule for dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext. The evaluator may use the SPD that was created for verification of FCS_IPSEC_EXT.1.1. The evaluator shall construct a network packet that matches the rule to allow the packet to flow in plaintext and send that packet. The evaluator should observe that the network packet is passed to the proper destination interface with no modification. The evaluator shall then modify a field in the packet header; such that it no longer matches the evaluator-created entries (there may be a "TOE created" final entry that discards packets that do not match any previous entries). The evaluator sends the packet and observes that the packet was dropped.
Test Execution Steps	• Configure policy on TOE to allow the packet to flow in plaintext. • Attempt a connection. • Verify Connection is successful. • Verify the packet capture. • Attempt a connection with modified header. • Verify connection is unsuccessful.
Pass/Fail with explanation	Pass. When the modified packet is sent, the TOE rejects the connection. This meets the testing requirements.

6.41 FCS_IPSEC_EXT.1.3 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.3 Test#1
Objective	Test 1: If tunnel mode is selected, the evaluator uses the guidance documentation to configure the TOE to operate in tunnel mode and also configures a VPN peer to operate in tunnel mode. The evaluator configures the TOE and the VPN peer to use any of the allowable cryptographic algorithms, authentication methods, etc. to ensure an allowable SA can be negotiated. The evaluator shall then initiate a connection from the TOE to connect to the VPN peer. The evaluator observes (for example, in the audit trail and the captured packets) that a successful connection was established using the tunnel mode.
Test Execution Steps	• Configure an IKE/IPsec connection (ensure that tunnel mode is configured). • Initiate traffic through IPsec Tunnel. • Verify Tunnel mode was used within logs. • Verify packet capture.
Pass/Fail with explanation	Pass. The TOE is able to be configured to support tunnel mode of operation. This meets the testing requirements.

6.42 FCS_IPSEC_EXT.1.4 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.4 Test#1
Objective	The evaluator shall configure the TOE as indicated in the guidance documentation configuring the TOE to use each of the supported algorithms, attempt to establish a connection using ESP, and verify that the attempt succeeds.
Test Execution Steps	<u>IKEv1</u> • Configure the TOE for IKEv1 AES-CBC-128 & SHA-256 configuration in ESP. • Configure the PEER for IKEv1 AES-CBC-128 & SHA-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-128 & SHA-256. • Verify via packet capture that the connection was established using AES-CBC-128 & SHA-256. • Configure the TOE for IKEv1 AES-CBC-192 & sha-256 configuration in ESP. • Configure the PEER for IKEv1 AES-CBC-192 & sha-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-192 & SHA256 in ESP. • Verify via packet capture that the connection was established using AES-CBC-192 & SHA256 in ESP. . • Configure the TOE for IKEv1 AES-CBC-256 & sha-256 configuration in the ESP. • Configure the PEER for IKEv1 AES-CBC-256 & sha-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-256 & SHA256 in ESP.

	• Verify via packet capture that the connection was established using AES-CBC-256 & SHA256 in ESP. • Configure the TOE for IKEv1 AES-GCM-128 in ESP. • Configure the PEER for IKEv1 AES-GCM-128 in ESP . • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-GCM-128. • Verify via packet capture that the connection was established using AES-GCM-128. • Configure the TOE for IKEv1 AES-GCM-256 in ESP. • Configure the PEER for IKEv1 AES-GCM-256 in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-GCM-256. • Verify via packet capture that the connection was established using AES-GCM-256. <u>IKEv2</u> • Configure the TOE for IKEv2 AES-CBC-128 & SHA-256 configuration in ESP. • Configure the PEER for IKEv2 AES-CBC-128 & SHA-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-128 & SHA-256. • Verify via packet capture that the connection was established using AES-CBC-128 & SHA-256. • Configure the TOE for IKEv2 AES-CBC-192 & sha-256 configuration in ESP. • Configure the PEER for IKEv2 AES-CBC-192 & sha-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-192 & sha-256. • Verify via packet capture that the connection was established using AES-CBC-192 & sha-256. • Configure the TOE for IKEv2 AES-CBC-256 & sha-256 configuration in the ESP. • Configure the PEER for IKEv2 AES-CBC-256 & sha-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-CBC-256 & SHA256. • Verify via packet capture that the connection was established using AES-CBC-256 & SHA256. • Configure the TOE for IKEv2 AES-GCM-128 in ESP. • Configure the PEER for IKEv2 AES-GCM-128 in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-GCM-128. • Verify via packet capture that the connection was established using AES-GCM-128. • Configure the TOE for IKEv2 AES-GCM-192 in ESP. • Configure the PEER for IKEv2 AES-GCM-192 in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-GCM-192. • Verify via packet capture that the connection was established using AES-GCM-192.

	• Configure the TOE for IKEv2 AES-GCM-256 configuration in the ESP. • Configure the PEER for IKEv2 AES-GCM-256 configuration in ESP. • Start an IPsec connection (using Ping). • Verify via logs that the connection was established using AES-GCM-256. • Verify via packet capture that the connection was established using AES-GCM-256.
Pass/Fail with explanation	Pass. IPsec SAs can be configured with each claimed algorithm. IPsec SAs can be configured with each claimed hash algorithm. This meets the testing requirements.

6.43 FCS_IPSEC_EXT.1.5 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.5 Test#1
Objective	Test 1: If IKEv1 is selected, the evaluator shall configure the TOE as indicated in the guidance documentation and attempt to establish a connection using an IKEv1 Phase 1 connection in aggressive mode. This attempt should fail. The evaluator should then show that main mode exchanges are supported.
Test Execution Steps	• Configure the TOE to support IKEv1 using main mode only. • Configure peer for aggressive mode. • Attempt to establish an IPsec session. • Verify that Aggressive mode connections are not possible via packet capture. • Verify that Aggressive mode connections are not possible via log. • Configure the PEER to support IKEv1 using main mode only • Attempt to establish an IPsec session via ping. This will pass • Verify that main mode is established in the ipsec connection via log • Verify that main mode is established in the ipsec connection via packet capture
Pass/Fail with explanation	Pass. The TOE rejected a connection attempt with Aggressive mode and then accepted a connection attempt with main mode. This meets the testing requirements.

6.44 FCS_IPSEC_EXT.1.6 Test#1

Item	**Data / Description**
Test ID	FCS_IPSEC_EXT.1.6 Test#1
Objective	The evaluator shall configure the TOE to use the ciphersuite under test to encrypt the IKEv1 and/or IKEv2 payload and establish a connection with a peer device, which is configured to only accept the payload encrypted using the indicated ciphersuite. The evaluator will confirm the algorithm was that used in the negotiation.
Test Execution Steps	• Configure the TOE with an IKE1 policy using AES-CBC-128. • Configure the Peer with an IKE1 policy using AES-CBC-128. • Attempt a connection between the two devices. • Verify that the negotiation uses AES-CBC-128 as specified in the policy. • Redo the same test for AES-CBC-192 & AES-CBC-256. • Redo for AES-128-GCM & AES-256-GCM for IKEv2.

Pass/Fail with explanation	Pass. IKE SAs can be configured with each claimed algorithm. This meets the testing requirements.

6.45 FCS_IPSEC_EXT.1.7 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.7 Test#1
Objective	If 'number of bytes' is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime in terms of the number of bytes allowed following the guidance documentation. The evaluator shall configure a test peer with a byte lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer, and determine that once the allowed number of bytes through this SA is exceeded, a new SA is negotiated. The evaluator shall verify that the TOE initiates a Phase 1 negotiation.
Pass/Fail with explanation	N/A, as ST does not claim 'number of bytes' to be checked for Phase1.

6.46 FCS_IPSEC_EXT.1.7 Test#2

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.7 Test#2
Objective	If 'length of time' is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime of 24 hours for the Phase 1 SA following the guidance documentation. The evaluator shall configure a test peer with a lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer, maintain the Phase 1 SA for 24 hours, and determine that a new Phase 1 SA is negotiated on or before 24 hours has elapsed. The evaluator shall verify that the TOE initiates a Phase 1 negotiation.
Test Execution Steps	• Configure the IKA SA Lifetime as 24 hours (86400 seconds) on the TOE • Configure the IKE SA for more than 24 hours (86500 seconds) on the peer • Capture the traffic between the TOE and peer • Establish and IPsec connection between the TOE and peer • Maintain the connection for 24 hours • Verify that a rekey was initiated before 24 hours via log review and packet capture
Pass/Fail with explanation	Pass. The TOE renegotiates phase 1 after the lifetime exceeds the lifetime of the TOE. This meets the testing requirements.

6.47 FCS_IPSEC_EXT.1.8 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.8 Test#1
Objective	If '**number of bytes**' is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime in terms of the number of bytes allowed following the guidance documentation. The evaluator shall configure a test peer with a byte lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer,

	and determine that once the allowed number of bytes through this SA is exceeded, a new SA is negotiated. The evaluator shall verify that the TOE initiates a Phase 2 negotiation.
Test Execution Steps	• Configure the bytes per lifetime. • Establish an IPsec session. • Transmit packets across the connections repeatedly. • Verify that when the bytes threshold is crossed a rekey is initiated
Pass/Fail with explanation	Pass. The TOE initiates a new SA when the allowed number of bytes through the existing SA is exceeded. This meets the testing requirements.

6.48 FCS_IPSEC_EXT.1.8 Test#2

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.8 Test#2
Objective	If 'length of time' is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime of 8 hours for the Phase 2 SA following the guidance documentation. The evaluator shall configure a test peer with a lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer, maintain the Phase 1 SA for 8 hours, and determine that once a new Phase 2 SA is negotiated when or before 8 hours has lapsed. The evaluator shall verify that the TOE initiates a Phase 2 negotiation.
Test Execution Steps	• Configure the IKA SA Lifetime as 8 hours (28800 seconds) on the TOE • Configure the IKE SA for more than 8 hours (30000 seconds) on the peer • Capture the traffic between the TOE and peer • Establish and IPsec connection between the TOE and peer • Maintain the connection for 8 hours • Verify that a rekey was initiated before 8 hours via log review and packet capture
Pass/Fail with explanation	Pass. The TOE initiated a rekey after the configured time (8 hours in this case). This meets the testing requirements.

6.49 FCS_IPSEC_EXT.1.11 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.11 Test#1
Objective	For each supported DH group, the evaluator shall test to ensure that all supported IKE protocols can be successfully completed using that particular DH group.
Test Execution Steps	IKEv1 • Configure DH group 14 for IKEv1 on TOE. • Configure DH group 14 for IKEv1 on PEER. • Start an IPsec connection (using Ping). • Verify that DH Group 14 was used via log. • Verify that Group 14 is used via capture. • Configure the TOE for Group 19. • Configure the Peer for Group 19. • Generate traffic to trigger the IPsec session.

	• Verify that DH group 19 was used via log . • Verify that DH Group 19 was used via packet capture. • Configure the TOE for Group 20. • Configure the Peer for Group 20. • Generate traffic to trigger the IPsec session. • Verify that DH group 20 was used via log. • Verify that DH Group 20 was used via packet capture. IKEv2 • Configure DH group 14 for IKEv2 on TOE. • Configure DH group 14 for IKEv2 on PEER. • Start an IPsec connection (using Ping). • Verify that DH Group 14 was used via log. • Verify that Group 14 is used via capture. • Configure the TOE for Group 19. • Configure the Peer for Group 19. • Generate traffic to trigger the IPsec session. • Verify that DH group 19 was used via log. • Verify that DH Group 19 was used via packet capture. • Configure the TOE for Group 20. • Configure the Peer for Group 20. • Generate traffic to trigger the IPsec session. • Verify that DH group 20 was used via log. • Verify that DH Group 20 was used via packet capture.
Pass/Fail with explanation	Pass. This test showed that each of the DH group supported by the TOE was configurable in IPsec connection. This meets the testing requirements.

6.50 FCS_IPSEC_EXT.1.12 Test #1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.12 Test#1
Objective	This test shall be performed for each version of IKE supported. The evaluator shall successfully negotiate an IPsec connection using each of the supported algorithms and hash functions identified in the requirements
Pass/Fail with explanation	Pass. This testing is covered by the requirements in FCS_IPSEC_EXT.1.4 Test#1 and FCS_IPSEC_EXT.1.6 Test#1.

6.51 FCS_IPSEC_EXT.1.12 Test#2

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.12 Test#2

Objective	This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an SA for ESP that selects an encryption algorithm with more strength than that being used for the IKE SA (i.e., symmetric algorithm with a key size larger than that being used for the IKE SA). Such attempts should fail.
Test Execution Steps	• Configure TOE to use AES-CBC-128 in P1 and AES-CBC-128 in P2 IKEv1. • Configure peer to use AES-CBC-128 in P1 and AES-CBC-256 in P2 IKEv1. • Attempt to establish a connection. • Verify the connection is rejected using logs. • Verify the connection is rejected using Packet Capture. • Configure TOE to use AES-CBC-128 in P1 and AES-CBC-128 in P2 IKEv2. • Configure peer to use AES-CBC-128 in P1 and AES-CBC-256 in P2 IKEv2. • Attempt to establish a connection. • Verify the connection is rejected using logs. • Verify the connection is rejected using Packet Capture.
Pass/Fail with explanation	Pass. When attempting to connect to a peer with the IPsec SA strength larger than the IKE SA strength, the TOE can reject the connection. This meets the testing requirements.

6.52 FCS_IPSEC_EXT.1.12 Test#3

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.12 Test#3
Objective	This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an IKE SA using an algorithm that is not one of the supported algorithms and hash functions identified in the requirements. Such an attempt should fail.
Test Execution Steps	**IKEv1** • Configure the TOE to use AES and SHA-256. • Configure the Peer to use 3DES and SHA-256. • Attempt a secure IPsec connection from peer. • Verify the connection is rejected via packet capture. • Verify the logs reflected on the TOE. **IKEv2** • Configure the TOE to use AES and SHA-256. • Configure the Peer to use 3DES and SHA-256. • Attempt a secure IPsec connection from peer. • Verify the connection is rejected via packet capture. • Verify the logs reflected on the TOE.
Pass/Fail with explanation	Pass. The TOE will only support and propose the configured algorithm. If the TOE peer does not have matching algorithms this session will not be established. This meets the testing requirements.

6.53 FCS_IPSEC_EXT.1.12 Test#4

Item	Data / Description

Test ID	FCS_IPSEC_EXT.1.12 Test#4
Objective	This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an SA for ESP (assumes the proper parameters where used to establish the IKE SA) that selects an encryption algorithm that is not identified in FCS_IPSEC_EXT.1.4. Such an attempt should fail.

Test Execution Steps	IKEv1:
	• Configure the TOE to support the following algorithms:
	  • IKE SA (Phase 1): AES-CBC-128, SHA-256
	  • IPsec SA (Phase 2): AES-CBC-128, SHA-256
	• Configure a peer to support the following algorithms:
	  • IKE SA (Phase 1): AES-CBC-128, SHA-256
	  • IPsec SA (Phase 2): Triple-DES, SHA-256
	• Attempt to make a connection.
	• Verify that the connection cannot be established via logs.
	• Verify that the connection cannot be established via packet Capture.
	
	IKEv2:
	
	• Configure the TOE to support the following algorithms:
	  • IKE SA (Phase 1): AES-CBC-128, SHA-256
	  • IPsec SA (Phase 2): AES-CBC-128, SHA-256
	• Configure a peer to support the following algorithms:
	  • IKE SA (Phase 1): AES-CBC-128, SHA-256
	  • IPsec SA (Phase 2): Triple-DES, SHA-256
	• Attempt to make a connection.
	• Verify that the connection cannot be established via logs.
	• Verify that the connection cannot be established via packet Capture.

Pass/Fail with explanation	Pass. Since the IPsec SA parameters did not match the IPsec SA parameters of the TOE peer, an IPsec connection could not be established. An IKE SA, however, could be established because the peer parameters matched. This meets the testing requirements.

6.54 FCS_IPSEC_EXT.1.14 Test#1

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#1
Objective	Test 1: (conditional) For each CN/identifier type combination selected, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the field in the peer's presented certificate and shall verify that the IKE authentication succeeds. If the TOE prioritizes CN checking over SAN (through explicit configuration of the field when specifying the reference identifier or prioritization rules), the evaluator shall also configure the SAN so it contains an incorrect identifier of the correct type (e.g. the reference identifier on the TOE is example.com, the CN=example.com, and the SAN:FQDN=otherdomain.com) and verify that IKE authentication succeeds.
Pass/Fail with explanation	N/A. The TOE does not prioritize CN checking over SAN.

6.55 FCS_IPSEC_EXT.1.14 Test#2

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#2
Objective	Test 2: (conditional) For each SAN/identifier type combination selected, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the field in the peer's presented certificate and shall verify that the IKE authentication succeeds. If the TOE prioritizes SAN checking over CN (through explicit specification of the field when specifying the reference identifier or prioritization rules), the evaluator shall also configure the CN so it contains an incorrect identifier formatted to be the same type (e.g. the reference identifier on the TOE is DNS-ID; identify certificate has an identifier in SAN with correct DNS-ID, CN with incorrect DNS-ID (and not a different type of identifier)) and verify that IKE authentication succeeds.
Test Execution Steps	• Create and load a peer certificate with a SAN IP that matches the TOE's reference identifier. • Verify through logs and a packet capture that the connection succeeds. • Create and load a peer certificate with a FQDN in the SAN that matches the TOE's reference identifier. • Verify through logs and a packet capture that the connection succeeds. • Create and load a peer certificate with a User FQDN in the SAN that matches the TOE's reference identifier. • Verify through logs and a packet capture that the connection succeeds. • Create and load a peer certificate with a SAN IP that matches the TOE's reference identifier but with an incorrect IP in the CN field. • Verify through logs and a packet capture that the connection succeeds. • Create and load a peer certificate with a FQDN in the SAN that matches the TOE's reference identifier but with an incorrect FQDN in the CN field. • Verify through logs and a packet capture that the connection succeeds. • Create and load a peer certificate with a User FQDN in the SAN that matches the TOE's reference identifier but with an incorrect User FQDN in the CN field. • Verify through logs and a packet capture that the connection succeeds.
Pass/Fail with explanation	Pass. The TOE accepts connections when the SAN matches with the PEER. This meets the testing requirements.

6.56 FCS_IPSEC_EXT.1.14 Test#3b

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#3b
Objective	Test 3: (conditional) For each CN/identifier type combination selected, the evaluator shall: a) Create a valid certificate with the CN so it contains the valid identifier followed by '\0'. If the TOE prioritizes CN checking over SAN (through explicit specification of the field when specifying the reference identifier or prioritization rules) for the same identifier type, the evaluator shall configure the SAN so it matches the reference identifier. b) Configure the peer's reference identifier on the TOE (per the administrative guidance) to match the CN without the '\0' and verify that IKE authentication fails.

Pass/Fail with explanation	N/A. The TOE does not prioritize CN checking over SAN.

6.57 FCS_IPSEC_EXT.1.14 Test#4b

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#4b
Objective	Test 4: (conditional) For each SAN/identifier type combination selected, the evaluator shall: a) Create a valid certificate with an incorrect identifier in the SAN. The evaluator shall configure a string representation of the correct identifier in the DN. If the TOE prioritizes CN checking over SAN (through explicit specification of the field when specifying the reference identifier or prioritization rules) for the same identifier type, the addition/modification shall be to any non-CN field of the DN. Otherwise, the addition/modification shall be to the CN. b) Configure the peer's reference identifier on the TOE (per the administrative guidance) to match the correct identifier (expected in the SAN) and verify that IKE authentication fails.
Test Execution Steps	• Create and load a peer certificate with an incorrect SAN IP but a correct IP in the CN field. • Verify through logs and a packet capture that the connection fails. • Create and load a peer certificate with an incorrect FQDN in the SAN but a correct FQDN in the CN field. • Configure the correct FQDN on the TOE's peer reference identifier. • Verify through logs and a packet capture that the connection fails. • Create and load a peer certificate with an incorrect User FQDN in the SAN but a correct User FQDN in the CN field. • Configure the correct User FQDN on the TOE's peer reference identifier. • Verify through logs and a packet capture that the connection fails.
Pass/Fail with explanation	Pass. The TOE rejects connection when the SAN mismatches with the PEER. This meets the testing requirements.

6.58 FCS_IPSEC_EXT.1.14 Test#5

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#5
Objective	Test 5: (conditional) If the TOE supports DN identifier types, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the subject DN in the peer's presented certificate and shall verify that the IKE authentication succeeds.
Test Execution Steps	• Configure a peer certificate with DN identifier types commonName, organizationalName, organizationalUnitName, and countryName. • Configure the TOE to use DN as the peer identity. • Verify that the connection succeeds.
Pass/Fail with explanation	Pass. The TOE establishes connection when the presented and the reference identifier of the DN match. This meets the testing requirements.

6.59 FCS_IPSEC_EXT.1.14 Test#6a

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#6a
Objective	Test 6: (conditional) If the TOE supports DN identifier types, to demonstrate a bit-wise comparison of the DN, the evaluator shall create the following valid certificates and verify that the IKE authentication fails when each certificate is presented to the TOE: a) Duplicate the CN field, so the otherwise authorized DN contains two identical CNs.
Test Execution Steps	• Create a peer certificate with a single CN field. • Use the Acumen x509-mod tool to duplicate CN on the DN of the certificate. • Present this certificate to the TOE and verify that the IKE authentication fails. • Verify the failure via Packet Capture.
Pass/Fail with explanation	Pass. When presented with a certificate that contains two identical CNs in the DN field, the TOE rejects the connection. This meets the testing requirements.

6.60 FCS_IPSEC_EXT.1.14 Test#6b

Item	Data / Description
Test ID	FCS_IPSEC_EXT.1.14 Test#6b
Objective	Test 6: (conditional) If the TOE supports DN identifier types, to demonstrate a bit-wise comparison of the DN, the evaluator shall create the following valid certificates and verify that the IKE authentication fails when each certificate is presented to the TOE: b) Append '\0' to a non-CN field of an otherwise authorized DN.
Test Execution Steps	• Create a peer certificate with '\0' appended to non-CN field. • Configure Peer with new certificate. • Initiate the traffic over the tunnel. • Present this certificate to the TOE and verify that the IKE authentication fails.
Pass/Fail with explanation	Pass. When presented with a certificate that has a Null Character appended to a non-CN field of an otherwise authorized DN, the TOE rejects the connection. This meets the testing requirements.

6.61 FCS_SSHS_EXT.1.2 Test #1

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.2 Test#1
Objective	Test 1: For each supported client public-key authentication algorithm, the evaluator shall configure a remote client to present a public key corresponding to that authentication method (e.g., 2048-bit RSA key when using ssh-rsa public key). The evaluator shall establish sufficient separate SSH connections with an appropriately configured remote non-TOE SSH client to demonstrate the use of all applicable public key algorithms. It is sufficient to observe the successful completion of the SSH Authentication Protocol to satisfy the intent of this test.

	TD0631 has been applied.
Test Execution Steps	• Generate an ecdsa-sha2-nistp256 public key on the VM. • Copy the public key onto the TOE and verify that it is updated on the TOE. • Login to the TOE using the public key and verify that the session is established. • Verify via logs that the session was established using the configured public key. • Verify via packet capture. • Generate an ecdsa-sha2-nistp384 public key on the VM. • Copy the public key onto the TOE and verify that it is updated on the TOE. • Login to the TOE using the public key and verify that the session is established. • Verify via logs that the session was established using the configured public key. • Verify via packet capture. • Generate an ecdsa-sha2-nistp521 public key on the VM. • Copy the public key onto the TOE and verify that it is updated on the TOE. • Login to the TOE using the public key and verify that the session is established. • Verify via logs that the session was established using the configured public key. • Verify via packet capture.
Pass/Fail with explanation	Pass. The remote client is able to establish a successful SSH connection using each one of the supported public key algorithms.

6.62 FCS_SSHS_EXT.1.2 Test #2

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.2 Test#2
Objective	Test 2: The evaluator shall choose one client public key authentication algorithm supported by the TOE. The evaluator shall generate a new client key pair for that supported algorithm without configuring the TOE to recognize the associated public key for authentication. The evaluator shall use an SSH client to attempt to connect to the TOE with the new key pair and demonstrate that authentication fails. TD0631 has been applied.
Test Execution Steps	• Generate a new client key pair with ecdsa-sha2-nistp521 on the VM. • Verify the key configured on the TOE. • Login to the device using public key without updating the public key on the TOE and verify that the connection fails. • Verify via audit logs that the connection fails. • Verify via packet capture that the connection fails
Pass/Fail with explanation	Pass. The TOE is not able to establish a connection with a remote SSH client when the TOE is not configured to recognize the associated public key for authentication. This meets the testing requirements.

6.63 FCS_SSHS_EXT.1.2 Test #3

Item	Data / Description

Test ID	FCS_SSHS_EXT.1.2 Test#3
Objective	Test 3: [Conditional] If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept password-based authentication and demonstrate that user authentication succeeds when the correct password is provided by the connecting SSH client. **TD0631 has been applied.**
Test Execution Steps	• Configure the TOE to ensure that the TOE supports password-based authentication. • Log into the TOE via SSH with password authentication. • Verify the Audit logs. • Verify Via Packet Capture.
Pass/Fail with Explanation	Pass. The TOE is able to establish a connection with a remote SSH user when correct authentication credentials are presented. This meets the testing requirements.

6.64 FCS_SSHS_EXT.1.2 Test #4

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.2 Test#4
Objective	Test 4: [Conditional] If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept password-based authentication and demonstrate that user authentication fails when the incorrect password is provided by the connecting SSH client. **TD0631 has been applied.**
Test Execution Steps	• Configure SSH on the TOE. • Attempt to Log into the TOE via SSH with password-based authentication and provide incorrect password (This will fail). • Verify authentication logs reflect failures. • Verify the Packet Capture.
Pass/Fail with Explanation	Pass. The TOE is not able to establish a connection with a remote SSH user when incorrect authentication credentials are presented. This meets the testing requirements.

6.65 FCS_SSHS_EXT.1.3 Test #1

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.3 Test#1
Objective	The evaluator shall demonstrate that if the TOE receives a packet larger than that specified in this component, that packet is dropped.
Test Execution Steps	• Use SSHS tool to send bad length packet • Verify the TOE logs. • Verify the Packet Captured.
Pass/Fail with explanation	Pass. The TOE drops large packets that are received within an SSH session. This meets the testing requirements.

6.66 FCS_SSHS_EXT.1.4 Test #1

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.4 Test#1
Objective	The evaluator must ensure that only claimed ciphers and cryptographic primitives are used to establish a SSH connection. To verify this, the evaluator shall start session establishment for a SSH connection from a remote client (referred to as 'remote endpoint' below). 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 all the ciphers defined in the TSS for the TOE for SSH sessions, but no additional ones compared to the definition in the TSS. The evaluator shall perform one successful negotiation of an SSH session to verify that the TOE behaves as expected. It is sufficient to observe the successful negotiation of the session to satisfy the intent of the test. If the evaluator detects that not all ciphers defined in the TSS for SSH are supported by the TOE and/or the TOE supports one or more additional ciphers not defined in the TSS for SSH, the test shall be regarded as failed.
Test Execution Steps	• Configure the TOE to support AES-128 for encryption algorithm. • Establish an SSH session with the configured supported algorithms. • Verify via audit log AES-128 was used. • Verify AES-128 was used via packet capture • Verify that the TOE only supports all algorithms as mentioned in the ST (aes-128-cbc, aes-256-cbc, aes-128-ctr, aes-256-ctr)
Pass/Fail with explanation	Pass. The TOE is able to make SSH connections with each claimed algorithm and The TOE rejects SSH connections using a non-approved algorithm. This meets the testing requirements.

6.67 FCS_SSHS_EXT.1.5 Test #1

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.5 Test#1
Objective	Test 1: The evaluator shall configure (only if required by the TOE) the TOE to use each of the claimed host public key algorithms. The evaluator will then use an SSH client to confirm that the client can authenticate the TOE server public key using the claimed algorithm. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test. **TD0631 has been applied.**
Test Execution Steps	• Configure the TOE to use the claimed host public key algorithms. • Generate an ecdsa-sha2-nistp256 host key pair on the TOE. • Login to the TOE using the host public key and verify that the session is established. • Verify via logs that the session was established. • Verify via packet capture that the configured host key algorithm was used. • Generate an ecdsa-sha2-nistp384 host key pair on the TOE. • Login to the TOE using the host public key and verify that the session is established. • Verify via logs that the session was established.

	• Verify via packet capture that the configured host key algorithm was used. • Generate an ecdsa-sha2-nistp521 host key pair on the TOE. • Login to the TOE using the host public key and verify that the session is established. • Verify via logs that the session was established. • Verify via packet capture that the configured host key algorithm was used.
Pass/Fail with explanation	Pass. The remote client is able to establish a succesful SSH connection using each one of the claimed host public key algorithms.

6.68 FCS_SSHS_EXT.1.5 Test #2

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.5 Test#2
Objective	Test 2: The evaluator shall configure a non-TOE SSH client to only allow it to authenticate an SSH server host public key algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection from the non-TOE SSH client to the TOE SSH server and observe that the connection is rejected. **TD0631 has been applied.**
Test Execution Steps	• Configure the TOE to reject SSH sessions using the unsupported ssh-rsa algorithm. • Attempt to establish an SSH session using the ssh-rsa host public key algorithm. • Verify that the connection is refused via packet capture. • Verify that the SSH session was refused using ssh-rsa via log.
Pass/Fail with explanation	Pass. The remote client is able to establish a successful SSH connection using each one of the claimed host public key algorithms.

6.69 FCS_SSHS_EXT.1.6 Test #1 & 2

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.6 Test#1 & 2
Objective	Test 1: (conditional, if an HMAC or AEAD_AES_*_GCM algorithm is selected in the ST) The evaluator shall establish an SSH connection using each of the algorithms, except “implicit”, specified by the requirement. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test. Note: To ensure the observed algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test. Test 2: (conditional, if an HMAC or AEAD_AES_*_GCM algorithm is selected in the ST) The evaluator shall configure an SSH client to only allow a MAC algorithm that is not included in the ST selection. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails. Note: To ensure the proposed MAC algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test.

Test Execution Steps	**Test-1 Steps:** • Configure the TOE to support **HMAC-SHA1** for hashing algorithm. • Establish an SSH session with the configured supported algorithms (HMAC-SHA1). • Verify that the SSH session was encrypted using HMAC-SHA1 via capture. • Verify that the message integrity algorithm used was as configured via log. • Verify that the TOE only supports all MAC algorithms as mentioned in the ST (**hmac-sha1,hmac-sha2-256,hmac-sha2-512**). **Test-2 Steps:** • Attempt to establish an SSH session using **hmac-sha1-96-etm.** • Verify via logs that the session fails due to unsupported mac algorithm. • Verify via wireshark that the TOE does not continue negotiation.
Pass/Fail with explanation	Pass. The TOE is able to make SSH connections with each claimed algorithm and The TOE rejects SSH connections using a non-approved algorithm. This meets the testing requirements.

6.70 FCS_SSHS_EXT.1.7 Test #1 & 2

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.7 Test#1 & 2
Objective	The evaluator shall configure an SSH client to only allow the diffiehellman-group1-sha1 key exchange. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails. For each allowed key exchange method, the evaluator shall configure an SSH client to only allow that method for key exchange, attempt to connect from the client to the TOE, and observe that the attempt succeeds.
Test Execution Steps	**Test 1-Steps** • Attempt to establish an SSH session using diffiehellman-group1-sha1. • Verify that the SSH session was refused via logs. • Verify the connection is refused via packet capture. **Test 2-Steps** • Establish an SSH session with the configured supported key exchange algorithm (Diffie-hellman-group14-sha1). • Verify that the session is established via Logs. • Verify that the SSH session was encrypted using Diffie-hellman-group14-sha1 via capture. • Verify that the TOE only supports all key exchange algorithm as mentioned in the ST (Diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521).

Pass/Fail with explanation	Pass. The TOE is able to make SSH connections with each claimed data integrity algorithm and the TOE rejects SSH connections using a non-approved algorithm. This meets the testing requirements.

6.71 FCS_SSHS_EXT.1.8 Test #1t & 1b

Item	Data / Description
Test ID	FCS_SSHS_EXT.1.8 Test#1t & 1b
Objective	The evaluator needs to perform testing that rekeying is performed according to the description in the TSS. The evaluator shall test both, the time-based threshold and the traffic-based threshold. For testing of the time-based threshold, the evaluator shall use an SSH client to connect to the TOE and keep the session open until the threshold is reached. The evaluator shall verify that the SSH session has been active longer than the threshold value and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at the maximum allowed value of one hour of session time, but the value used for testing shall not exceed one hour. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. For testing of the traffic-based threshold the evaluator shall use the TOE to connect to an SSH client and shall transmit data to and/or receive data from the TOE within the active SSH session until the threshold for data protected by either encryption key is reached. It is acceptable if the rekey occurs before the threshold is reached (e.g., because the traffic is counted according to one of the alternatives given in the Application Note for FCS_SSHS_EXT.1.8). The evaluator shall verify that more data has been transmitted within the SSH session than the threshold allows and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at the maximum allowed value of one gigabyte of transferred traffic, but the value used for testing shall not exceed one gigabyte. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, the evaluator needs to verify that the threshold(s) can be configured as described in the guidance documentation and the evaluator needs to test that modification of the thresholds is restricted to Security Administrators (as required by FMT_MOF.1/Functions). In cases where data transfer threshold could not be reached due to hardware limitations it is acceptable to omit testing of this (SSH rekeying based on data transfer threshold) threshold if both the following conditions are met a) An argument is present in the TSS section describing this hardware based limitation and b) All hardware components that are the basis of such argument are definitively identified in the ST. For example, if specific Ethernet Controller or WiFi radio chip is the root cause of such limitation, these chips must be identified.

Test Execution Steps	Test 1a Steps: • Login to the TOE and configure a rekey for 3 Minutes. • Send a continuous ping and verify that a rekey generates every 10 Minutes. • Verify the login time for rekey. • Verify rekey via audit logs. Test 1b Steps • Configure the volume limit for rekeying on TOE. • Copy the file from Source to other TOE which is above 10MB in size to occur rekey. • Verify via logs that a rekey is generated in every 10 MB of data.
Pass/Fail with explanation	Pass. TOE successfully rekeyed when the time and traffic limit was reached. This meet the testing requirements.

6.72 FPT_TST_EXT.1 Test#1

Item	Data / Description
Test ID	FPT_TST_EXT.1 Test#1
Objective	It is expected that at least the following tests are performed: a) Verification of the integrity of the firmware and executable software of the TOE b) Verification of the correct operation of the cryptographic functions necessary to fulfil any of the SFRs. The evaluator shall either verify that the self-tests described above are carried out during initial start-up or that the developer has justified any deviation from this.
Test Execution Steps	• Reset or boot the TOE. • Observe boot processes for integrity testing and self-tests.
Pass/Fail with explanation	Pass. The TOE performs all claimed self-tests. This meets the testing requirements.

6.73 FPT_TUD_EXT.1 Test #1

Item	Data / Description
Test ID	FPT_TUD_EXT.1 Test #1
Objective	The evaluator performs the version verification activity to determine the current version of the product as well as the most recently installed version (should be the same version before updating). The evaluator obtains a legitimate update using procedures described in the guidance documentation and verifies that it is successfully installed on the TOE. (For some TOEs loading the update onto the TOE and activation of the update are separate steps ('activation' could be performed e.g. by a distinct activation step or by rebooting the device). In that case the evaluator verifies after loading the update onto the TOE but before

	activation of the update that the current version of the product did not change but the most recently installed version has changed to the new product version.) After the update, the evaluator performs the version verification activity again to verify the version correctly corresponds to that of the update and that current version of the product and most recently installed version match again.
Test Execution Steps	• Copy update file to the TOE. • Verify the current version of the TOE • Attempt to install a legitimate update • Verify the new version of the TOE
Pass/Fail with explanation	Pass. The TOE was successfully upgraded to the new version. This meets the testing requirements.

6.74 FPT_TUD_EXT.1 Test #2a

Item	Data / Description
Test ID	FPT_TUD_EXT.1 Test #2a
Objective	Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 1) A modified version (e.g. using a hex editor) of a legitimately signed update.
Test Execution Steps	• Verify the current version of the TOE. • Attempt to install a modified version of a legitimate update. • Verify the TOE rejects the update.
Pass/Fail with explanation	Pass. The TOE software was able to detect when an image was corrupted and rejected the image. This meets the testing requirements.

6.75 FPT_TUD_EXT.1 Test #2b

Item	Data / Description
Test ID	FPT_TUD_EXT.1 Test #2b
Objective	Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates:

	2) An image that has not been signed.
Test Execution Steps	• Verify the current version of the TOE. • Attempt to install a update without a signature. • Verify the TOE rejects the update.
Pass/Fail with explanation	Pass. The TOE software was able to detect when an image was not signed and rejected the image. This meets the testing requirements.

6.76 FPT_TUD_EXT.1 Test #2c

Item	Data / Description
Test ID	FPT_TUD_EXT.1 Test #2c
Objective	Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 3) An image signed with an invalid signature (e.g. by using a different key as expected for creating the signature or by manual modification of a legitimate signature).
Test Execution Steps	• Verify the current version of the TOE. • Attempt to install an update with a corrupt signature. • Verify the TOE rejects the update.
Pass/Fail with explanation	Pass. The TOE software was able to detect when an image had an invalid signature and rejected the image. This meets the testing requirements.

6.77 FIA_X509_EXT.1.1/Rev Test #1a

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #1a
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: a) Test 1a: The evaluator shall present the TOE with a valid chain of certificates (terminating in a trusted CA certificate) as needed to validate the leaf certificate to be used in the function and shall use this chain to demonstrate that the function succeeds. Test 1a shall be designed in a way that the chain can be 'broken' in Test 1b by either being able to remove the trust anchor from the TOEs trust store, or by setting up the trust store in a way that at least one

	intermediate CA certificate needs to be provided, together with the leaf certificate from outside the TOE, to complete the chain (e.g. by storing only the root CA certificate in the trust store).
Test Execution Steps	• Create a valid chain of certificates and present them to the TOE. • Verify that the connection succeeds.
Pass/Fail with explanation	Pass. When a complete certificate trust chain is present, the TOE is able to make a successful IKE/IPsec connection.

6.78 FIA_X509_EXT.1.1/Rev Test#1a(ECDSA)

Item	Data / Description
Test ID	FIA_X509_EXT.1.1Test#1a(ECDSA)
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: a) Test 1a: The evaluator shall present the TOE with a valid chain of certificates (terminating in a trusted CA certificate) as needed to validate the leaf certificate to be used in the function and shall use this chain to demonstrate that the function succeeds. Test 1a shall be designed in a way that the chain can be 'broken' in Test 1b by either being able to remove the trust anchor from the TOEs trust store, or by setting up the trust store in a way that at least one intermediate CA certificate needs to be provided, together with the leaf certificate from outside the TOE, to complete the chain (e.g. by storing only the root CA certificate in the trust store).
Test Execution Steps	• Create a valid chain of certificates and present them to the TOE. • Verify that the connection succeeds.
Pass/Fail with explanation	Pass. When a complete certificate trust chain is present, the TOE is able to make a successful IKE/IPsec connection.

6.79 FIA_X509_EXT.1.1/Rev Test #1b

Item	Data / Description
Test ID	FIA_X509_EXT.1.1/Rev Test #1b
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509

	certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 1b: The evaluator shall then 'break' the chain used in Test 1a by either removing the trust anchor in the TOE's trust store used to terminate the chain, or by removing one of the intermediate CA certificates (provided together with the leaf certificate in Test 1a) to complete the chain. The evaluator shall show that an attempt to validate this broken chain fails.
Test Execution Steps	• Delete the root CA from the TOE. • Attempt a connection. • Verify through logs that a successful connection cannot be established.
Pass/Fail with explanation	Pass. When an incomplete certificate trust chain is present, the TOE is not able to make a successful IKE/IPsec connection.

6.80 FIA_X509_EXT.1.1/Rev Test #2

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #2
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.
Test Execution Steps	• Use a valid and unexpired certificate on the TOE. • Change the internal time on the TOE to a date past the expiration date of the certificate. • Attempt to verify the certificate once more. • Attempt to establish a connection with the expired certificate. • Verify through logs that a successful connection cannot be established.
Pass/Fail with explanation	Pass. The TOE denied the connection because of the expired certificate. This meets the testing requirements.

6.81 FIA_X509_EXT.1.1/Rev Test #3(CRL)

Item	Data / Description

Test ID	FIA_X509_EXT.1.1 Test #3(CRL)
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 3: The evaluator shall test that the TOE can properly handle revoked certificates-– conditional on whether CRL or OCSP is selected; if both are selected, then a test shall be performed for each method. The evaluator shall test revocation of the peer certificate and revocation of the peer intermediate CA certificate i.e. the intermediate CA certificate should be revoked by the root CA. The evaluator shall 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. Revocation checking is only applied to certificates that are not designated as trust anchors. Therefore, the revoked certificate(s) used for testing shall not be a trust anchor.
Test Execution Steps	CRL • Create a valid certificate chain and upload them to the TOE and peer device. • Verify that the CRL downloads successfully and that there are no revoked certificates. • Attempt a connection between the TOE and the peer and verify that the connection is successful. • TOE logs verify unsuccessful connection negotiation when intermediate CA certificate is revoked. • Verify that the TOE successfully downloads the updated CRL. • Attempt a connection between the TOE and the peer and verify that the connection fails. • TOE logs verify unsuccessful connection negotiation when end entity certificate is revoked. • Verify that the TOE successfully downloads the updated CRL. • Attempt a connection between the TOE and the peer and verify that the connection fails.
Pass/Fail with explanation	Pass. The TOE does not communicate with peers that either have a revoked certificate or one of their intermediate CA certificates are revoked. When presented non-revoked certificates, the TOE accepts the certificate. This meets the testing requirements

6.82 FIA_X509_EXT.1.1/Rev Test #4(CRL)

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #4(CRL)
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509

	certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 4: If OCSP 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 OSCP 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 Execution Steps	• Configure a connection on the TOE with CRL checking enabled. • Configure the CA signing the CRL to use a signing certificate that does not have the CRLsign key usage bit set. • Load the new CA to the TOE. • Verify that the connection fails.
Pass/Fail with explanation	Pass. The TOE rejected the CRL when CA signing the CRL to use a signing certificate that does not have the cRLsign key usage bit set. This meets the testing requirements.

6.83 FIA_X509_EXT.1.1/Rev Test #5

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #5
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: 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 Execution Steps	• Configure StrongSwan peer. • Connect the TOE to a StrongSwan peer. • Run the StrongSwan Acumen tool to modify the first byte of the encoding certificate by incrementing 30 to 31. • Verify that the TOE rejects the connection.
Pass/Fail with explanation	Pass. The TOE rejects connections when the first byte of the certificate is modified. This meets the testing requirements.

6.84 FIA_X509_EXT.1.1/Rev Test #6

Item	Data / Description

Test ID	FIA_X509_EXT.1.1 Test #6
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 6: The evaluator shall modify any byte in the certificate signatureValue field (see RFC5280 Sec. 4.1.1.3), which is normally the last field in the certificate, and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
Test Execution Steps	• Run the StrongSwan Acumen tool to modify the last byte of the encoding certificate by incrementing 9e to 9f. • Verify that the TOE rejects the connection.
Pass/Fail with explanation	Pass. The TOE rejects connections when the last byte of the certificate is modified. This meets the testing requirements.

6.85 FIA_X509_EXT.1.1/Rev Test #7

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #7
Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The hash of the certificate will not validate.)
Test Execution Steps	• Run the StrongSwan Acumen tool to modify any byte in public key of certificate by incrementing 82 to 83 . • Verify that the TOE rejects the connection .
Pass/Fail with explanation	Pass. The TOE rejects connections when the public of the certificate is modified. This meets the testing requirements.

6.86 FIA_X509_EXT.1.1/Rev Test #8a

Item	Data / Description

Test ID	FIA_X509_EXT.1.1 Test #8a
Objective	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** **(Conditional on TOE ability to process CA certificates presented in certificate message**) The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a valid chain of EC certificates (terminating in a trusted CA certificate), where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain. **TD0527 (12/1 Update) has been applied.** Test 8a: (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 Execution Steps	• Create a certificate chain with three certificates using EC curves. • Add Root CA and TOE ICA as trust anchor for TOE. • Attempt a connection from a remote server and verify that it is successful. • Verify the connection with packet capture.
Pass/Fail with Explanation	Pass. The TOE validates the certificate chain with the EC parameter.

6.87 FIA_X509_EXT.1.1/Rev Test #8b

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #8b
Objective	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** **(Conditional on TOE ability to process CA certificates presented in certificate message)** The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a chain of EC certificates (terminating in a trusted CA certificate), where the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key information field, and is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid. **TD0527 (12/1 Update) has been applied.** Test 8b: (Conditional on support for EC certificates as indicated in FCS_COP.1(3)). 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 Execution Steps	• Replace the ICA in the earlier test with a modified certificate signed by the trusted RootCA with the named EC curves replaced by explicit curves . • Attempt a connection from the remote server and verify that it fails. • Verify the failed connection with a packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when the ICA certificate has been modified.

6.88 FIA_X509_EXT.1.1/Rev Test #8c

Item	Data / Description
Test ID	FIA_X509_EXT.1.1 Test #8c
Objective	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** The evaluator shall establish a subordinate CA certificate, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is accepted into the TOE's trust store. The evaluator shall then establish a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is rejected, and not added to the TOE's trust store. **TD0527 (12/1 Update) has been applied.**
Test Execution Steps	• Add a subordinate CA certificate into a TOE's trust store, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA and observe that it is accepted. • Add a subordinate CA certificate into a TOE's trust store, that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA and observe that it is rejected
Pass/Fail with Explanation	Pass. The TOE rejects a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters signed by a trusted EC root CA.

6.89 FIA_X509_EXT.1.2/Rev Test #1

Item	Data / Description
Test ID	FIA_X509_EXT.1.2 Test #1
Objective	The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests it to verify that the TOE accepts only certificates that have been marked as CA certificates by using basicConstraints with the CA flag set to True (and implicitly

	that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: - a self-signed root CA certificate, - an intermediate CA certificate and - a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). Test 1: The evaluator shall ensure that at least one of the CAs in the chain does not contain the basicConstraints extension. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: (i) as part of the validation of the leaf certificate belonging to this chain; when attempting to add a CA certificate without the basic Constraints extension to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).
Test Execution Steps	• Create a CA certificate that does not contain the basic Constraints extension. • TOE logs for a local certificate is signed by the CA that does not contain the basic Constraints extension. • Load the CA and local certificate onto the TOE. • Verify that the TOE identifies the signing CA certificate does not contain the basic Constraints extension rejects the certificate signed by it. • Verify the connection between TOE and Peer fails.
Pass/Fail with explanation	Pass. TOE rejects certificates signed by CA that does not contain the BasicConstraint Extension.

6.90 FIA_X509_EXT.1.2/Rev Test #2

Item	Data / Description
Test ID	FIA_X509_EXT.1.2 Test #2
Objective	The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests it to verify that the TOE accepts only certificates that have been marked as CA certificates by using basicConstraints with the CA flag set to True (and implicitly that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: - a self-signed root CA certificate, - an intermediate CA certificate and

	- a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). Test 2: The evaluator shall ensure that at least one of the CA certificates in the chain has a basicConstraints extension in which the CA flag is set to FALSE. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: (i) as part of the validation of the leaf certificate belonging to this chain; (ii) when attempting to add a CA certificate with the CA flag set to FALSE to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).
Test Execution Steps	• Create a CA certificate that has the CA flag in the basicConstraints extension set to FALSE. • A local certificate is signed by the CA that has the CA flag in the basicConstraints extension set to FALSE. • Load the CA and local certificate unto the TOE. • Verify that the TOE identifies the signing CA certificate has the CA flag in the basicConstraints extension set to FALSE. and rejects the certificate signed by it. • Verify the connection between TOE and Peer fails.
Pass/Fail with explanation	Pass. The TOE rejects connections when the False CA used to sign a certificate, this meets the testing requirement.

6.91 FIA_X509_EXT.2 Test #1 (CRL)

Item	Data / Description
Test ID	FIA_X509_EXT.2 Test #1 (CRL)
Objective	The evaluator shall perform the following test for each trusted channel: The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate, and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the guidance documentation to determine that all supported administrator-configurable options behave in their documented manner.
Test Execution Steps	• Configure the TOE to use CRL fetch for revocation checking. • Delete every CRL from the web server. • Verify that the TOE can no longer Download a new CRL. • Verify that the TOE does not establish a connection when it cannot download a CRL. • Configure the TOE to allow connections to be established when CRLs could not be retrieved. • Delete every CRL from the web server. • Verify that the TOE can no longer Download a new CRL. • Verify that the TOE establishes connection as per configured by the administrator when validity of certificate cannot be determined.

Pass/Fail with explanation	Pass. The TOE successfully rejects certificates when validation service is unavailable. This meets the testing requirements.

6.92 FIA_X509_EXT.3 Test #1

Item	Data / Description
Test ID	FIA_X509_EXT.3 Test #1
Objective	Test 1: The evaluator shall use the guidance documentation to cause the TOE to generate a Certification Request. The evaluator shall capture the generated request and ensure that it conforms to the format specified. The evaluator shall confirm that the Certification Request provides the public key and other required information, including any necessary user-input information.

Test Execution Steps	• From the TOE, generate a CSR.
	• Examine the CSR contents.
	• Ensure the CSR contains the following fields
	  ○ ***Public key***
	  ○ ***Device-specific information***
	  ○ ***Common Name***
	  ○ ***Organization***
	  ○ ***Organizational Unit***
	  ○ ***Country***

Pass/Fail with explanation	Pass. The TOE is able to generate a CSR with all the requisite information. This meets the testing requirements.

6.93 FIA_X509_EXT.3 Test #2

Item	Data / Description
Test ID	FIA_X509_EXT.3 Test #2
Objective	"Test 2: The evaluator shall demonstrate that validating a response message to a Certification Request without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the response message, and demonstrate that the function succeeds."
Test Execution Steps	• From the TOE, generate a CSR request. • Generate a signed certificate based on the generated CSR from an external CA. • Ensure that the full trust chain for the signed CA is not present on the TOE. • Load the signed certificate on the TOE. • Verify that the TOE rejects the certificate because the full trust chain of the CA is not present. • Add the intermediary certificates to the TOE certificate store to ensure that the signing CA now has a full certificate path. • Re-attempt to load the signed certificate on the TOE. • Verify that the TOE accepts the certificate because the path validation succeeded. • Remove the signing CA intermediary certificates from the TOE certificate store. • Verify that the TOE now identifies the signed certificate as invalid.

Pass/Fail with explanation	Pass. The TOE does not install CSR responses signed by a CA without a full trust path. The TOE does install a CSR response signed by a CA with a full trust path. This meets the testing requirements.

6.94 FAU_GEN.1/IPS Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall test that the interfaces used to configure the IPS polices yield expected IPS data in association with the IPS policies. A number of IPS policy combination and ordering scenarios need to be configured and tested by attempting to pass both allowed and anomalous network traffic matching configured IPS policies in order to trigger all required IPS events. Note that this activity should have been addressed with a combination of the Test assurance activities for the other IPS requirements.
Test Execution Steps	Covered by audit records in each test case.
Pass/Fail with Explanation	Pass. Covered by audit records in each test case of IPS module.

6.95 FMT_SMF.1/IPS Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall use the operational guidance to create a signature and enable it on an interface. The evaluator shall then generate traffic that would be successfully triggered by the signature. The evaluator should observe the TOE applying the corresponding reaction in the signature.
Test Steps	• Configure a custom signature filter on the TOE to deny traffic with a specific source address. • Apply the filter to the TOE's security policy. • Send modified traffic that matches the configured filter and verify traffic was denied as per filter. • Verify through a packet capture and through logs that the traffic was appropriately denied.
Pass/Fail with Explanation	Pass. Traffic matching the signature successfully triggers the TOE and applies the corresponding reaction in the signature. This meets the testing requirements.

6.96 FMT_SMF.1/IPS Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall then disable the signature and attempt to regenerate the same traffic and ensure that the TOE allows the traffic to pass with no reaction

Test Steps	• Disable the signature from FMT_SMF.1/IPS Test #1. • Generate the same traffic. • Verify through a packet capture and through logs that the traffic was appropriately allowed to flow.
Pass/Fail with Explanation	Pass. After disabling the signature, the TOE allows the same traffic to pass through it with no reaction. This meets the testing requirements.

6.97 FMT_SMF.1/IPS Test #3

Item	Data
Test Assurance Activity	Test 3: The evaluator shall use the operational guidance to import signatures and repeat the test conducted in Test 1.
Test Steps	• Select a signature among the pre-existing signatures in the TOE and configure TOE to generate alarm when traffic matching signature is encountered. • Apply the signature to the security zone upon which the TOE's interface is assigned. • Send traffic that matches the header-based signature. • Verify the attack traffic is detected by the TOE and reacts accordingly.
Pass/Fail with Explanation	Pass. The TOE has imported signatures which once enabled, detect traffic matching the signature on the configured interface. This meets the testing requirements.

6.98 IPS_ABD_EXT.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall use the instructions in the operational guidance to configure baselines or anomaly-based rules for each attributes specified in IPS_ABD_EXT.1.1. The evaluator shall send traffic that does not match the baseline or matches the anomaly-based rule and verify the TOE applies the configured reaction. This shall be performed for each attribute in IPS_ABD_EXT.1.1.
Test Steps	Throughput: • Create a policer to monitor the throughput. • Apply the policer to a firewall filter which specifies the IPv4 source address. • Apply the configuration to the interface. • Modify and send traffic to match the configured filter on the TOE. • Verify through logs that the TOE reacts according to configuration. • Verify via packet capture that the TOE reacts according to configuration. • To ensure that the counter has a base number of 1, the traffic is sent again without resetting the logs. • The TOE allowed and logged an additional two packets while the third was dropped and picked up by the policer. The policer counter increased by one from the previous 1

	• Verify via packet capture. • Create a policer to monitor the throughput. • Apply the policer to a firewall filter which specifies the IPv4 destination address. • Apply the configuration to the interface. • Modify and send traffic to match the configured filter on the TOE. • Verify through logs that the TOE reacts according to configuration. • Verify via packet capture that the TOE reacts according to configuration. • To ensure that the counter has a base number of 1, the traffic is sent again without resetting the logs. • The TOE allowed and logged an additional two packets while the third was dropped and picked up by the policer. The policer counter increased by one from the previous 1 • Verify via packet capture. Time of Day: • Create a schedule, set for five minutes on a particular day of the week. • Apply the schedule configuration to the security policy for a particular IPV6 source address. • Send traffic to match the configured filter on the TOE during the scheduled time. • Verify through logs that while the scheduler was enabled, the appropriate traffic was denied. • Verify via Packet Capture. • After the scheduler time was complete, initiate the same traffic to the TOE. • Verify via logs that traffic matching configured filter after schedule time does not get logged under policy with scheduler. • Verify via Packet Capture. • Create a schedule set for five minutes on a particular day of the week. • Apply the schedule configuration to the security policy for a particular IPV6 destination address. • Send traffic to match the configured filter on the TOE during the scheduled time. • Verify through logs that while the scheduler was enabled, the appropriate traffic was denied. • Verify via Packet Capture. • After the scheduler time was complete, initiate the same traffic to the TOE. • Verify via logs that traffic matching configured filter after schedule time does not get logged under policy with scheduler. • Verify via Packet Capture. Frequency: • Create a filter to monitor the frequency for a specific TCP source port. • Apply the configuration to the interface. • Send traffic to match the configured filter on the TOE. • Verify with logs that the frequency of traffic is logged, and that TOE applies the configured reaction.

	• Create a filter to monitor the frequency for a specific TCP destination port. • Apply the configuration to the interface. • Send traffic to match the configured filter on the TOE. • Verify with logs that the frequency of traffic is logged, and that TOE applies the configured reaction. Threshold: • Configure the TOE to send trap messages when threshold has been exceeded. • Send traffic to match configured rpm probe with specific UDP source port. • Verify through logs that the TOE sends trap messages when threshold exceeded. • Verify through Packet Capture. • Configure the TOE to send trap messages when threshold has been exceeded. • Send traffic to match configured rpm probe with specific UDP destination port. • Verify through logs that the TOE sends trap messages when threshold exceeded. • Verify through Packet Capture.
Pass/Fail with Explanation	Pass. TOE applies the configured reaction for anomaly-based rules for each attribute (throughput, time of day, frequency, threshold). This meets the testing requirements.

6.99 IPS_ABD_EXT.1 Test #2

Item	Data
Test Assurance Activity	Test 2: Repeat the test assurance activity above to ensure that baselines or anomaly-based rules can be defined for each distinct network interface type supported by the TOE.
Pass/Fail with Explanation	Pass. All distinct network interface types supported by the TOE for attributes Throughput, Time of day, Frequency and threshold have been tested as part of IPS_ABD_EXT.1 Test #1

6.100 IPS_IPB_EXT.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall use the instructions in the operational guidance to create a known-bad address list. Using a single IP address, a list of addresses or a range of addresses from that list, the evaluator shall attempt to send traffic through the TOE that would otherwise be allowed by the TOE and observe the TOE automatically drops that traffic
Test Steps	• Create a single entry of a known-bad address and an additional entry with a range of known-bad addresses. • Send traffic that matches the configured entries of known-bad addresses and verify connection succeeds. • Apply the entries to the security policy of TOE. • Send traffic that matches the configured entries of known-bad addresses.

	• Verify through a packet capture and through the TOE's logs that traffic was appropriately denied.
Pass/Fail with Explanation	Pass. TOE drops traffic matching the configured know bad address list, which would otherwise be allowed. This meets the testing requirements.

6.101 IPS_IPB_EXT.1 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall use the instructions in the operational guidance to create a known-good address list. Using a single IP address, a list of addresses or a range of addresses from that list, the evaluator shall attempt to send traffic that would otherwise be denied by the TOE and observe the TOE automatically allowing traffic
Test Steps	• Create a single entry of a known-good address and an additional entry with a range of known-good addresses. • Send traffic that matches the configured entries of known-good addresses and verify connection fails. • Apply the entries to the security policy of TOE. • Send traffic that matches the configured entries of known-good addresses. • Verify through a log that connection was created. • Verify through Wireshark Capture that Connection succeeds.
Pass/Fail with Explanation	Pass. TOE allows traffic matching the configured know good address list, which would otherwise not be allowed. This meets the testing requirements.

6.102 IPS_IPB_EXT.1 Test #3

Item	Data
Test Assurance Activity	Test 3: The evaluator shall add conflicting IP addresses to each list and ensure that the TOE handles conflicting traffic in a manner consistent with the precedence in IPS_NTA_EXT.1.1.
Test Steps	• Add the good-known addresses from Test 2 to the bad-known address list in Test 1. • Add the bad-known addresses from Test 1 to the good-known address list in Test 2. • Apply the address book entries to two security policies to deny and accept the same address entries with deny policy being applied first. • Send traffic matching the security policies applied • Verify through a packet Capture and logs that the traffic is denied. • Apply the address book entries to two security policies to deny and accept the same address entries with accept policy being applied first. • Send traffic matching the security policies applied • Verify through a packet Capture and logs that the traffic is accepted.

Pass/Fail with Explanation	Pass. TOE handles conflicting traffic in an Administrator-defined order. This meets the testing requirements.

6.103 IPS_SBD_EXT.1.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall use the instructions in the operational guidance to test that packet header signatures can be created and/or configured with the selected and/or configured reactions specified in IPS_SBD_EXT.1.5 for each of the attributes listed below. Each attribute shall be individually assigned to its own unique signature: • IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options. • IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options. • ICMP: Type; Code; Header Checksum; and Rest of Header (varies based on the ICMP type and code). • ICMPv6: Type; Code; and Header Checksum;. • TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options. • UDP: Source port; destination port; length; and UDP checksum. Using packet sniffers, the evaluator will generate traffic to trigger a signature and using packet captures will ensure that the reactions of each rule are performed as expected.
Test Steps	For each of the attributes: • IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options. • IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options. • ICMP: Type; Code; Header Checksum; and Rest of Header (varies based on the ICMP type and code). • ICMPv6: Type; Code; and Header Checksum. • TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options. • UDP: Source port; destination port; length; and UDP checksum. • Configure a filter on the TOE to drop traffic matching the attribute. • Apply the filter to the TOE's security policy. • Modify traffic to match the configured filter on the TOE. • Verify through a packet capture and through logs that the traffic was appropriately dropped.

Pass/Fail with Explanation	Pass. TOE is triggered with traffic matching configured signatures and reacts in the expected way by dropping the traffic. This meets the testing requirements.

6.104 IPS_SBD_EXT.1.1 Test #2

Item	Data
Test Assurance Activity	Test 2: Repeat the test assurance activity above to ensure that signature-based IPS policies can be defined for each distinct network interface type capable of applying signatures as supported by the TOE.
Pass/Fail with Explanation	Pass. Not applicable as all distinct network interface types supported by the TOE have been tested as part of IPS_SBD_EXT.1.1 Test #1.

6.105 IPS_SBD_EXT.1.2 Test #1

Item	Data

Test Assurance Activity	Test 1: The evaluator shall use the instructions in the operational guidance to test that packet payload string-based detection rules can be assigned to the reactions specified in IPS_SBD_EXT.1.5 using the attributes specified in IPS_SBD_EXT.1.2. However, it is not required (nor is it feasible) to test all possible strings of protocol data, the evaluator shall ensure that a selection of strings in the requirement is selected to be tested. At a minimum at least one string using each of the following attributes from IPS_SBD_EXT.1.2 should be tested for each protocol. The evaluator shall generate packets that match the string in the rule and observe the corresponding reaction is as configured.
	
	• Test at least one string of characters for ICMPv4 data: beyond the first 4 bytes of the ICMP header.
	• Test at least one string of characters for ICMPv6 data: beyond the first 4 bytes of the ICMP header.
	• TCP data (characters beyond the 20 byte TCP header):
	  ○ Test at least one FTP (file transfer) command: help, noop, stat,syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.
	  ○ HTTP (web) commands and content:
	  ▪ Test both GET and POST commands
	  ▪ Test at least one administrator-defined strings to match URLs/URIs, and web page content.
	  ○ Test at least one SMTP (email) state: start state, SMTP commands state, mail header state, mail body state, abort state.
	  ○ Test at least one string in any additional attribute type defined within [selection: [assignment: other types of TCP payload inspection];
	• Test at least one string of UDP data: characters beyond the first 8 bytes of the UDP header;

	Test at least one string for each additional attribute type defined in [assignment: other types of packet payload inspection]]
Pass/Fail with Explanation	Pass. The TOE detects when packets contain a specific strings of protocol data and reacts by dropping and logging the offending traffic. This meets the testing requirements.

6.106 IPS_SBD_EXT.1.2 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall repeat one of the tests in Test 1 but generate multiple nonfragmented packets that contain the string in the rule defined.
Test Steps	• Configure a filter on the TOE to search for the string SECURITY. • Apply the filter to the TOE's security policy. • Modify traffic to match the configured filter. • Verify through a packet capture and through logs that the modified traffic was not allowed through the TOE.
Pass/Fail with Explanation	Pass. The TOE detects non-fragmented packets containing a specific string and immediately reacts by dropping the offending traffic. This meets the testing requirements.

6.107 IPS_SBD_EXT.1.2 Test #3

Item	Data
Test Assurance Activity	Test 3: Repeat the test assurance activity above to ensure that signature-based IPS policies can be defined for each distinct network interface type capable of applying signatures as supported by the TOE
Pass/Fail with Explanation	Pass. Not applicable as all distinct network interface types supported by the TOE have been tested as part of IPS_SBD_EXT.1.2 Test #1.

6.108 IPS_SBD_EXT.1.3 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall create and/or configure rules for each attack signature in IPS_SBD_EXT.1.3. For each attack, the TOE should apply its corresponding signature and enable it to each distinct network interface type capable of applying the signatures. The evaluator shall use packet captures to ensure that the attack traffic is detected by the TOE and a reaction specified in IPS_SBD_EXT.1.5 is triggered and stops the attack. Each attack should be performed one after another so as to ensure that its corresponding signature successfully identified and appropriately reacted to a particular attack
Test Steps	IP Attacks ○ IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)

	○ Create a rule to detect when the IP fragments overlap. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. ○ IP source address equal to the IP destination (Land attack) ○ Create a rule to detect when the IP source address and destination address are equal. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. ICMP Attacks ○ Fragmented ICMP Traffic (e.g. Nuke attack) ○ Create a rule to detect ICMP fragmented packets ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. ○ Large ICMP Packet (e.g. Ping of Death) ○ Create a rule to detect Large ICMP Packets ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. TCP Attacks ○ TCP NULL Flag ○ Create a rule to detect TCP Null flags ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. ○ TCP FIN+SYN Flag ○ Create a rule to detect TCP FIN+SYN flags ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. ○ TCP FIN only Flags ○ Create a rule to detect TCP FIN only flags ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly.

	○ TCP SYN+RST Flag
	  ○ Create a rule to detect TCP SYN+RST flags
	  ○ Apply the signature rule to the security zone to which TOE's interface is assigned.
	  ○ Send traffic that matches the header-based signature.
	  ○ Verify the attack traffic is detected by the TOE and reacts accordingly.
	
	UDP Attacks
	○ UDP Bomb Attack
	  ○ Create a rule to detect UDP Bomb Attack and apply the signature rule to the security zone to which TOE's interface is assigned
	  ○ Send traffic that matches the header-based signature.
	  ○ Verify the attack traffic is detected by the TOE and reacts accordingly.
	
	○ UDP Chargen DoS Attack
	  ○ Create a rule to detect Chargen DoS Attack and apply the signature rule to the security zone to which TOE's interface is assigned
	  ○ Send traffic that matches the header-based signature.
	  ○ Verify the attack traffic is detected by the TOE and reacts accordingly.

Pass/Fail with Explanation	Pass. Each attack traffic matching configured signature is detected by the TOE and a reaction specified in IPS_SBD_EXT.1.5 is triggered and stops the attack. This meets the testing requirements.

6.109 IPS_SBD_EXT.1.4 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall configure individual signatures for each attack in IPS_SBD_EXT.1.4. For each attack, the TOE should apply its corresponding signature and enable it to each distinct network interface type capable of applying signatures. The evaluator shall use packet captures to ensure that the attack traffic is detected by the TOE and a reaction specified in IPS_SBD_EXT.1.5 is triggered and stops the attack. Each attack should be performed one after another so as to ensure that its corresponding signature successfully identified and appropriately reacted to a particular attack.

Test Steps	Flooding a host (DoS Attack)
	• ICMP flooding (Smurf attack, and ping flood)
	  ○ Create a rule to detect ICMP Flood attack.
	  ○ Apply the signature rule to the security zone to which TOE's interface is assigned.
	  ○ Send traffic that matches the header-based signature.
	  ○ Verify the attack traffic is detected by the TOE and reacts accordingly.
	
	• TCP flooding (e.g. SYN Flood)
	  ○ Create a rule to detect TCP SYN Flood attack.
	  ○ Apply the signature rule to the security zone to which TOE's interface is assigned.

	○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. Flooding a network (DoS Attack) • Flooding a network (DoS Attack) ○ Create a rule to detect Network Flood Attack. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. Protocol and Port Scanning • IP Protocol Scanning ○ Create a rule to detect IP Protocol Scanning. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. • TCP Port Scanning ○ Create a rule to detect TCP Port Scanning. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. • UDP Port Scanning ○ Create a rule to detect UDP Port Scanning. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly. • ICMP Scanning ○ Create a rule to detect ICMP Scanning. ○ Apply the signature rule to the security zone to which TOE's interface is assigned. ○ Send traffic that matches the header-based signature. ○ Verify the attack traffic is detected by the TOE and reacts accordingly.
Pass/Fail with Explanation	Pass. The TOE detects when there is attack traffic and reacts by dropping the offending traffic.

6.110 IPS_SBD_EXT.1.6 Test #1

Item	Data

Test Assurance Activity	The evaluator shall repeat one of the tests in IPS_SBD_EXT.1.2 Test 1 but generate multiple nonfragmented packets that contain the string in the rule defined. The evaluator shall verify that the malicious traffic is still detected when split across multiple non-fragmented packets.
Test Steps	• Configure a filter on the TOE to search for the string SECURITY. • Apply the filter to the TOE's security policy. • Modify traffic to match the configured filter. • Verify through a packet capture and through logs that the modified traffic was not allowed through the TOE.
Pass/Fail with Explanation	Pass. The TOE detects the malicious traffic even when split across multiple non-fragmented packets.

6.111 FAU_GEN.1/VPN Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall attempt to flood the TOE with network packets such that the TOE will be unable to process all the packets. This may require the evaluator to configure the TOE to limit the bandwidth the TOE is capable to handling (e.g., use of a 10 MB interface). The evaluator shall then review the audit logs to verify that the TOE correctly records that it is unable to process all of the received packets and verify that the TOE logging behavior is consistent with the TSS.
Test Steps	• Configure the TOE to limit the amount of half-open TCP connections. • Apply the configuration to the TOE's interface. • Send continuous traffic to the TOE. • Verify that when the configured threshold is reached a log entry is generated and a counter is incremented. • Verify with logs. • Verify with packet capture.
Pass/Fail with Explanation	Pass. The audit logs verify that the TOE correctly records that it is unable to process all of the received packets. The TOE logging behavior is consistent with the TSS. This meet the requirements.

6.112 FAU_GEN.1/VPN Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall use a remote VPN client to establish an IPsec session with the TOE and observe that the event is logged in accordance with the expectations of the PP-Module.
Pass/Fail with Explanation	N/A. VPN client testing is not included in ST.

6.113 FPF_RUL_EXT.1.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would otherwise be denied by the ruleset should be sourced and directed to a host. The evaluator shall use a packet sniffer to verify none of the generated network traffic is permitted through the TOE during initialization.
Test Steps	• Configure a filter to drop traffic from a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the chosen source address and verify that it is denied. • Reboot the TOE when ping is in progress. • Verify with logs that all traffic from chosen source address was denied. • Verify with Packet Capture that all traffic from chosen source address was denied during reboot.
Pass/Fail with Explanation	Pass. Packets that would otherwise be denied by the ruleset are not permitted through the TOE during initialization. This meets the testing requirements.

6.114 FPF_RUL_EXT.1.1 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would be permitted by the ruleset should be sourced and directed to a host. The evaluator shall use a packet sniffer to verify none of the generated network traffic is permitted through the TOE during initialization and is only permitted once initialization is complete.
Test Steps	• Configure a filter to accept traffic with a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the specific source address and verify it is accepted. • Reboot the TOE when ping is in progress. • Verify through the firewall log that traffic from specific source address is allowed after the reboot • Verify through a packet capture that all traffic is denied when the TOE is performing a reboot but once the TOE is operational all traffic from the specific source address is allowed.

Pass/Fail with Explanation	Pass. Packets that would otherwise be allowed by the ruleset are not permitted through the firewall during initialization. This meets the testing requirements.

6.115 FPF_RUL_EXT.1.4 Test #1

Item	Data

Test Assurance Activity	Test 1: The evaluator shall use the instructions in the operational guidance to test that packet filter rules can be created that permit, deny, and log packets for each of the following attributes:
	• IPv4
	  ○ Source address
	  ○ Destination Address
	  ○ Protocol
	• IPv6
	  ○ Source address
	  ○ Destination Address
	  ○ Next Header (Protocol)
	• TCP
	  ○ Source Port
	  ○ Destination Port
	• UDP
	  ○ Source Port
	  ○ Destination Port
	
	Note that these test activities should be performed in conjunction with those of FPF_RUL_EXT.1.6 where the effectiveness of the rules is tested; here the evaluator is just ensuring the guidance is sufficient and the TOE supports the administrator creating a ruleset based on the above attributes. The test activities for FPF_RUL_EXT.1.6 define the protocol/attribute combinations required to be tested. If those combinations are configured manually, that will fulfill the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.

Test Steps	• IPv4
	  ○ Source address
	  • Configure a filter to drop and accept traffic with specified IPv4 source addresses.
	  • Apply the IPv4 source address filter.
	  • Generate and send traffic that matches the applied filter.
	  • Verify the IPV4packets are dropped or accepted according to the filter applied using logs.
	  • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture.

	○ Destination Address • Configure a filter to drop and accept traffic with specified IPv4 source addresses. • Apply the IPv4 destination address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV4 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Transport Layer Protocol • Configure a filter to drop and accept traffic with a specified IPv4 transport layer protocol. • Apply the IPv4 protocol filter. • Generate and send traffic that matches the applied filter. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using Packet Capture. • IPv6 ○ Source address • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 source address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Destination Address • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 destination address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture.

	○ Transport Layer Protocol • Configure a filter to drop and accept traffic with a specified IPv6 transport layer protocol. • Apply the IPv6 protocol filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using Packet Capture. • TCP ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. • UDP ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter, • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs.

	• Verify the traffic was sent via Wireshark packet capture.
Pass/Fail with Explanation	Pass. TOE permits, denies, and logs for IPV4, IPV6, TCP and UDP packets according to packet filtering rules applied on the TOE. This meets the testing requirements.

6.116 FPF_RUL_EXT.1.4 Test #2

Item	Data
Test Assurance Activity	Test 2: Repeat the test assurance activity above to ensure that Packet filtering rules can be defined for each distinct network interface type supported by the TOE. Note that these test activities should be performed in conjunction with those of FPF_RUL_EXT.1.6 where the effectiveness of the rules is tested; here the evaluator is just ensuring the guidance is sufficient and the TOE supports the administrator creating a ruleset based on the above attributes. The test activities for FPF_RUL_EXT.1.6 define the protocol/attribute combinations required to be tested. If those combinations are configured manually, that will fulfill the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.
Test Steps	• IPv4 ○ Source address • Configure a filter to drop and accept traffic with specified IPv4 source addresses. • Apply the IPv4 source address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Destination Address • Configure a filter to drop and accept traffic with specified IPv4 source addresses. • Apply the IPv4 destination address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Transport Layer Protocol

	• Configure a filter to drop and accept traffic with a specified IPv4 transport layer protocol. • Apply the IPv4 protocol filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using Packet Capture. • IPv6 ○ Source address • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 source address filter to VPN Interface to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Destination Address • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 destination address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Transport Layer Protocol • Configure a filter to drop and accept traffic with a specified IPv6 transport layer protocol. • Apply the IPv6 protocol filter. • Generate and send traffic that matches the applied filter to VPN Interface. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using Packet Capture. • TCP ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports.

	• Apply the source port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. • UDP ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports • Apply the destination port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs • Verify the traffic was sent via Wireshark packet capture.
Pass/Fail with Explanation	Pass. TOE permits, denies, and logs for IPV4, IPV6, TCP and UDP packets according to packet filtering rules applied on the TOE VPN interface. This meets the testing requirements.

6.117 FPF_RUL_EXT.1.5 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall devise two equal Packet filtering rules with alternate operations – permit and deny. The rules should then be deployed in two distinct orders and in each case the evaluator shall ensure that the first rule is enforced in both cases by generating applicable packets and using packet capture and logs for confirmation.

Test Steps	• Configure a filter to allow and drop packets that have the same destination-address with the allow rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is allowed. • Verify allowed traffic via packet capture. • Configure a filter to drop and allow packets that have the same destination-address with the drop rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is discarded. • Verify via packet capture discarded traffic.
Pass/Fail with Explanation	Pass. TOE enforces the first rule in the firewall filter. This meets the testing requirement.

6.118 FPF_RUL_EXT.1.5 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall repeat the procedure above, except that the two rules should be devised where one is a subset of the other (e.g., a specific address vs. a network segment). Again, the evaluator should test both orders to ensure that the first is enforced regardless of the specificity of the rule.
Test Steps	• Configure the firewall rule order to allow packets to a specific destination-address and deny packets to its network segment. • Apply the filter to the TOE Interface. • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that only traffic to specific destination address are allowed and remaining addresses to network segment are discarded. • Verify the rules applied through Packet Capture. • Configure the firewall rule order to deny packets to a network segment and allow packets to a specific destination-address of the network segment. • Apply the filter to the TOE Interface • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that all traffic is dropped. • Verify the rules applied through Packet Capture.
Pass/Fail with Explanation	Pass. TOE enforces the first rule regardless of the specificity of the rule. This meets the testing requirements.

6.119 FPF_RUL_EXT.1.6 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall configure the TOE to permit and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each supported IPv4 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Any protocols not supported by the TOE must be denied. Table of protocols: IP Transport layer protocols.xlsx **TD0597 has been applied.**
Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Allow condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is allowed. • Verify with TOE logs that all combinations are allowed via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE permits and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address as configured on TOE for each defined IPV4 Transport Layer Protocol. This meets testing requirements.

6.120 FPF_RUL_EXT.1.6 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall configure the TOE to permit all traffic except to discard and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv4 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must also be denied but are not required to be logged. **TD0597 has been applied.**

Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Deny condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is denied. • Verify with TOE logs that all combinations are Denied via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE denies and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address as configured on TOE for each defined IPV4 Transport Layer Protocol. This meets testing requirements.

6.121 FPF_RUL_EXT.1.6 Test #3

Item	Data
Test Assurance Activity	Test 3: The evaluator shall configure the TOE to permit and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. Additionally, the evaluator shall configure the TOE to discard and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with different (than those permitted above) combinations of a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each supported IPv4 Transport Layer Protocol and outside the scope of all source and destination addresses configured above in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must be denied. **TD0597 has been applied.**
Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Discard condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is discarded. • Verify with TOE logs that all combinations are discarded via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE discards and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and

	wildcard destination address as configured on TOE for each defined IPV4 Transport Layer Protocol. This meets testing requirements.

6.122 FPF_RUL_EXT.1.6 Test #4

Item	Data
Test Assurance Activity	Test 4: The evaluator shall configure the TOE to permit and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Any protocols not supported by the TOE must be denied. **TD0597 has been applied.**
Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Allow condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is allowed. • Verify with TOE logs that all combinations are allowed via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE permits and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address as configured on TOE for each defined IPV6 Transport Layer Protocol. This meets testing requirements.

6.123 FPF_RUL_EXT.1.6 Test #5

Item	Data
Test Assurance Activity	Test 5: The evaluator shall configure the TOE to permit all traffic except to discard and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must also be denied but are not required to be logged. **TD0597 has been applied.**

Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Deny condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is denied. • Verify with TOE logs that all combinations are Denied via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE denies and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address as configured on TOE for each defined IPV6 Transport Layer Protocol. This meets testing requirements.

6.124 FPF_RUL_EXT.1.6 Test #6

Item	Data
Test Assurance Activity	Test 6: The evaluator shall configure the TOE to permit and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. Additionally, the evaluator shall configure the TOE to discard and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with different (than those permitted above) combinations of a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and outside the scope of all source and destination addresses configured above in order to ensure that the supported protocols are dropped (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must be denied. **TD0597 has been applied.**
Test Steps	• Configure TOE for all combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address for Discard condition. • Configure Test Machine to send traffic with selected source and destination IPs • Send Traffic with all combination to check traffic is discarded. • Verify with TOE logs that all combinations are discarded via TOE. • Verify with Packet capture.
Pass/Fail with Explanation	Pass. The TOE discards and logs packets with combinations of specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and

	wildcard destination address as configured on TOE for each defined IPV6 Transport Layer Protocol. This meets testing requirements.

6.125 FPF_RUL_EXT.1.6 Test #7

Item	Data
Test Assurance Activity	Test 7: The evaluator shall configure the TOE to permit and log protocol 6 (TCP) using a selected source port, a selected destination port, and a selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination TCP ports in order to ensure that they are permitted (i.e., by capturing the packets after passing through the TOE) and logged.
Test Steps	• Create a filter to configure the TOE to permit and log protocol 6 (TCP) using a selected source port. • Apply the filter the TOE's interface. • Generate traffic to match the filter applied to the TOE's interface. • Verify through firewall log the correct traffic was permitted through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to permit and log protocol 6 (TCP) using a selected destination port. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log that the correct traffic was permitted through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to permit and log protocol 6 (TCP) using a selected source and destination port combination. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log that the correct traffic was permitted through the interface. • Verify through Packet Capture.
Pass/Fail with Explanation	Pass. The TOE permits and logs TCP traffic with a specific source port, destination port, and a combination of both the source and destination port. This meets the testing requirement.

6.126 FPF_RUL_EXT.1.6 Test #8

Item	Data
Test Assurance Activity	Test 8: The evaluator shall configure the TOE to discard and log protocol 6 (TCP) using a selected source port, a selected destination port, and a selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination TCP ports in order to ensure that they are denied (i.e., by capturing no applicable packets passing through the TOE) and logged.

Test Steps	• Create a filter to configure the TOE to deny and log protocol 6 (TCP) using a selected source port. • Apply the filter the TOE's interface. • Generate traffic to match the filter applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to deny and log protocol 6 (TCP) using a selected destination port. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to deny and log protocol 6 (TCP) using a selected source and destination port combination. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture.
Pass/Fail with Explanation	Pass. The TOE discards and logs TCP traffic with a specific source port, destination port, and a combination of both the source and destination port. This meets testing requirement.

6.127 FPF_RUL_EXT.1.6 Test #9

Item	Data
Test Assurance Activity	Test 9: The evaluator shall configure the TOE to permit and log protocol 17 (UDP) using a selected source port, a selected destination port, and a selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination UDP ports in order to ensure that they are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Here the evaluator ensures that the UDP port 500 (IKE) is included in the set of tests.
Test Steps	• Create a filter to configure the TOE to permit and log protocol 17 (UDP) using a selected source port • Apply the filter the TOE's interface. • Generate traffic to match the filter applied to the TOE's interface. • Verify through firewall log the correct traffic was permitted through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to permit and log protocol 17 (UDP) using a selected destination port • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log that the correct traffic was permitted through the interface.

	• Verify through Packet Capture. • Create a filter to configure the TOE to permit and log protocol 17 (UDP) using a selected source and destination port combination. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log that the correct traffic was permitted through the interface. • Verify through Packet Capture.
Pass/Fail with Explanation	Pass. The TOE permits and logs UDP protocol traffic with a specific source port, destination port, and a combination of both the source and destination port.

6.128 FPF_RUL_EXT.1.6 Test #10

Item	Data
Test Assurance Activity	Test 10: The evaluator shall configure the TOE to discard and log protocol 17 (UDP) using a selected source port, a selected destination port, and a selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination UDP ports in order to ensure that they are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Again, the evaluator ensures that UDP port 500 is included in the set of tests.
Test Steps	• Create a filter to configure the TOE to deny and log protocol 17 (UDP) using a selected source port • Apply the filter the TOE's interface. • Generate traffic to match the filter applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to deny and log protocol 17 (UDP) using a selected destination port • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture. • Create a filter to configure the TOE to deny and log protocol 17 (UDP) using a selected source and destination port combination. • Apply the filter the TOE's interface. • Generate traffic to match the filters applied to the TOE's interface. • Verify through firewall log the configured traffic was denied through the interface. • Verify through Packet Capture.
Pass/Fail with Explanation	Pass. The TOE discards and logs UDP traffic with a specific source port, destination port, and a combination of both the source and destination port. This meets testing requirement.

6.129 FIA_PSK_EXT.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall compose a pre-shared key of 22 characters that contains a combination of the allowed characters in accordance with the operational guidance and demonstrates that a successful protocol negotiation can be performed with the key.
Test Steps	• Configure the TOE with a pre-shared key of 22 characters that contains a combination of the allowed characters. • Verify that a successful protocol negotiation can be performed with the key. • Verify successful connection with packet capture.
Pass/Fail with Explanation	Pass. The TOE allows a successful protocol negotiation with a pre-shared key of 22 characters.

6.130 FIA_PSK_EXT.1 Test #2

Item	Data
Test Assurance Activity	Test 2 [conditional]: If the TOE supports pre-shared keys of **multiple lengths**, the evaluator shall repeat Test 1 using the minimum length; the maximum length; and an invalid length. The minimum and maximum length tests should be successful, and the invalid length must be rejected by the TOE.
Test Steps	• Configure the TOE with pre-shared keys with minimum length - 1 • Verify that a successful protocol negotiation can be performed with the keys. • Verify successful connection with packet capture. • Configure the TOE with pre-shared keys with maximum length - 255 • Verify that a successful protocol negotiation can be performed with the keys. • Verify successful connection with packet capture. • Configure the TOE with pre-shared keys with invalid length – 260 and confirm it is not accepted
Pass/Fail with Explanation	Pass. The TOE supports pre-shared keys of multiple lengths where the minimum and maximum keys are able to perform successful protocol negotiation, but a key of an invalid length cannot be used to perform a successful protocol negotiation.

6.131 FIA_PSK_EXT.1 Test #3

Item	Data

Test Assurance Activity	Test 3 [conditional]: If the TOE **does not generate** bit-based pre-shared keys, the evaluator shall obtain a bit-based pre-shared key of the appropriate length and enter it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key.
Test Steps	• Configure the TOE with bit-based pre-shared key. • Verify that a successful protocol negotiation can be performed with a bit-based pre-shared key. • Verify successful connection with packet capture.
Pass/Fail with Explanation	Pass. The TOE performs a successful protocol negotiation with a bit-based pre-shared key.

6.132 FFW_RUL_EXT.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would otherwise be denied by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization.
Test Steps	IPV4: • Configure a filter to drop traffic from a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the chosen source address and verify that it is denied. • Reboot the TOE when ping is in progress. • Verify with logs that traffic from chosen source address was denied. • Verify with Packet Capture that all traffic from chosen source address was denied during reboot. IPV6: • Configure a filter to drop traffic from a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the chosen source address and verify that it is denied. • Reboot the TOE when ping is in progress. • Verify with logs that traffic from chosen source address was denied. • Verify with Packet Capture that all traffic from chosen source address was denied during reboot.
Pass/Fail with Explanation	Pass. Packets that would otherwise be denied by the ruleset are not permitted through the TOE during initialization. This meets the testing requirements.

6.133 FFW_RUL_EXT.1 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would be permitted by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization and is only permitted once initialization is complete.
Test Steps	IPV4: • Configure a filter to accept traffic with a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the specific source address and verify it is accepted. • Reboot the TOE when ping is in progress. • Verify through the firewall log that traffic from specific source address is allowed after the reboot. • Verify through a packet capture that all traffic is denied when the TOE is performing a reboot but once the TOE is operational all traffic from the specific source address is allowed. IPV6: • Configure a filter to accept traffic with a specific source address. • Apply the filter to the TOE's Interface. • Send continual traffic from the specific source address and verify it is accepted. • Reboot the TOE when ping is in progress. • Verify through the firewall log that traffic from specific source address is allowed after the reboot • Verify through a packet capture that all traffic is denied when the TOE is performing a reboot but once the TOE is operational all traffic from the specific source address is allowed.
Pass/Fail with Explanation	Pass. Packets that would otherwise be allowed by the ruleset are not permitted through the firewall during initialization. This meets the testing requirements.

6.134 FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #1

Item	Data

Test Assurance Activity	Test 1: The evaluator shall use the instructions in the guidance documentation to test that state full packet filter firewall rules can be created that permit, drop, and log packets for each of the following attributes:
	• IPv4
	  ○ Source address
	  ○ Destination Address

	○ Transport Layer Protocol • IPv6 ○ Source address ○ Destination Address ○ Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields • TCP ○ Source Port ○ Destination Port • UDP ○ Source Port ○ Destination Port • ICMPv4 ○ Type ○ Code • ICMPv6 ○ Type ○ Code
Test Steps	**IPv4** ○ Source address • Configure a filter to drop and accept traffic with specified IPv4 source addresses. • Apply the IPv4 source address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV4packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Destination Address • Configure a filter to drop and accept traffic with specified IPv4 destination addresses. • Apply the IPv4 destination address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV4 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Transport Layer Protocol

	• Configure a filter to drop and accept traffic with a specified IPv4 transport layer protocol. • Apply the IPv4 protocol filter. • Generate and send traffic that matches the applied filter. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using Packet Capture. **IPv6** ○ Source address • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 source address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Destination Address • Configure a filter to drop and accept traffic with specified IPv6 destination addresses. • Apply the IPv6 destination address filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ Transport Layer Protocol • Configure a filter to drop and accept traffic with a specified IPv6 transport layer protocol. • Apply the IPv6 protocol filter. • Generate and send traffic that matches the applied filter. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using Packet Capture. **TCP** ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports.

	• Apply the source port filter. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. **UDP** ○ Source Port • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter, • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ Destination Port • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. **ICMPv4** • **Type** ○ Configure a filter to accept and drop ICMPV4 packets according to its type. ○ Apply the ICMPv4 type filter ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV4 packets are dropped or accepted according to the rules applied based on type. ○ Verify the traffic was sent via Wireshark packet capture • **Code** ○ Configure a filter to accept and drop ICMPV4 packets according to its code. ○ Apply the ICMPv4 type filter ○ Generate and send traffic that matches the created filter

	○ Verify through logs that the ICMPV4 packets are dropped or accepted according to the rules applied based on code. ○ Verify the traffic was sent via Wireshark packet capture **ICMPv6** • **Type** ○ Configure a filter to accept and drop ICMPV4 packets according to its type. ○ Apply the ICMPv6 type filter ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on type. ○ Verify the traffic was sent via Wireshark packet capture • **Code** ○ Configure a filter to accept and drop ICMPV4 packets according to its code. ○ Apply the ICMPv6 type filter ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on code. ○ Verify the traffic was sent via Wireshark packet capture
Pass/Fail with Explanation	Pass. TOE can implement full packet filter firewall rules that permit, drop, and log packets for each of the specified attributes. This meets the testing requirements.

6.135 FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #2

Item	Data
Test Assurance Activity	Test 2: Repeat the test assurance activity above to ensure that state full traffic filtering rules can be defined for each distinct network interface type supported by the TOE
Test Steps	**IPv4** ○ **Source address** • Configure a filter to drop and accept traffic with specified IPv4 source addresses. • Apply the IPv4 source address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ **Destination Address** • Configure a filter to drop and accept traffic with specified IPv4 destination addresses.

	• Apply the IPv4 destination address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture. ○ **Transport Layer Protocol** • Configure a filter to drop and accept traffic with a specified IPv4 transport layer protocol. • Apply the IPv4 protocol filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using Packet Capture. **IPv6** ○ **Source address** • Configure a filter to drop and accept traffic with specified IPv6 source addresses. • Apply the IPv6 source address filter to VPN Interface to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ **Destination Address** • Configure a filter to drop and accept traffic with specified IPv6 destination addresses. • Apply the IPv6 destination address filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the IPV6 packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture. ○ **Transport Layer Protocol** • Configure a filter to drop and accept traffic with a specified IPv6 transport layer protocol. • Apply the IPv6 protocol filter. • Generate and send traffic that matches the applied filter to VPN Interface.

	• Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using logs. • Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using Packet Capture. **TCP** ○ **Source Port** • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ **Destination Port** • Configure a filter to drop and accept traffic according to specified destination ports. • Apply the destination port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the TCP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. **UDP** ○ **Source Port** • Configure a filter to drop and accept traffic according to specified source ports. • Apply the source port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs. • Verify the traffic was sent via Wireshark packet capture. ○ **Destination Port** • Configure a filter to drop and accept traffic according to specified destination ports • Apply the destination port filter to VPN Interface. • Generate and send traffic that matches the applied filter. • Verify the UDP packets are dropped or accepted according to the filter applied using logs • Verify the traffic was sent via Wireshark packet capture. **ICMPv4** • **Type** ○ Configure a filter to accept and drop ICMPV4 packets according to its type.

	○ Apply the ICMPv4 type filter to VPN Interface ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV4 packets are dropped or accepted according to the rules applied based on type. ○ Verify the traffic was sent via Wireshark packet capture • **Code** ○ Configure a filter to accept and drop ICMPV4 packets according to its code. ○ Apply the ICMPv4 type filter to VPN Interface ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV4 packets are dropped or accepted according to the rules applied based on code. ○ Verify the traffic was sent via Wireshark packet capture **ICMPv6** • **Type** ○ Configure a filter to accept and drop ICMPV4 packets according to its type. ○ Apply the ICMPv6 type filter to VPN Interface ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on type. ○ Verify the traffic was sent via Wireshark packet capture • **Code** ○ Configure a filter to accept and drop ICMPV4 packets according to its code. ○ Apply the ICMPv6 type filter to VPN Interface. ○ Generate and send traffic that matches the created filter ○ Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on code. ○ Verify the traffic was sent via Wireshark packet capture.
Pass/Fail with Explanation	Pass. TOE can implement full packet filter firewall rules that permit, drop, and log packets for each of the specified attributes on the VPN interface. This meets the testing requirements.

6.136 FFW_RUL_EXT.1.5 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall configure the TOE to permit and log TCP traffic. The evaluator shall initiate a TCP session. While the TCP session is being established, the evaluator shall introduce session establishment packets with incorrect flags to determine that the altered traffic is not accepted as part of the session (i.e., a log event is generated to show the ruleset was applied). After a TCP session is successfully established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and destination ports, sequence number, flags) one at a time in order to verify that the altered packets are not accepted as part of the established session.
Test Steps	IPV4 • Configure the TOE to permit and log TCP traffic • Apply the filter to the TOE's interface

	• Establish a TCP session and send data
	• Modify each of the session attributes one at a time:
	  ○ Source address
	  ○ Destination address
	  ○ Source port
	  ○ Destination port
	  ○ Sequence number
	  ○ Flags
	• Verify the altered packets are logged by the firewall filter
	• Verify through the packet capture that the altered packets are not accepted as part of the established session
	
	IPV6
	
	• Configure the TOE to permit and log TCP traffic with specific source and destination addresses.
	• Apply the filter to the TOE's interface.
	• Establish a TCP session and send data.
	• Modify each of the session attributes one at a time:
	  ○ Source address
	  ○ Destination address
	  ○ Source port
	  ○ Destination port
	  ○ Sequence number
	  ○ Flags
	• Verify the altered packets are logged by the firewall filter.
	• Verify through the packet capture that the altered packets are not accepted as part of the established session.

Pass/Fail with Explanation	Pass. TOE does not accept altered packets (source and destination addresses, source and destination ports, sequence number, flags) after a TCP session is successfully established. This meets testing requirements.

6.137 FFW_RUL_EXT.1.5 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall terminate the TCP session established per Test 1 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
Test Steps	IPV4 • Configure the TOE to permit and log TCP traffic. • Apply the filter to the TOE's interface. • Establish a TCP session then terminate the session. • Send a packet that matches the former TCP session. • Verify that the Firewall logs the TCP packet similar to former session.

	IPV6 • Configure the TOE to permit and log TCP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface. • Establish a TCP session then terminate the session. • Send a packet that matches the former TCP session. • Verify that the Firewall logs the TCP packet similar to former session.
Pass/Fail with Explanation	Pass. Any packet matching the TCP former session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

6.138 FFW_RUL_EXT.1.5 Test #3

Item	Data
Test Assurance Activity	Test 3: The evaluator shall expire (i.e., reach timeout) the TCP session established per Test 1 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
Test Steps	IPV4 • Configure the TOE to permit and log TCP traffic. • Apply the filter to the TOE's interface. • Establish a TCP session and wait for the session to expire. • Send a packet that matches the former TCP session. • Verify that the Firewall logs the TCP packet similar to former session. IPV6 • Configure the TOE to permit and log TCP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface. • Establish a TCP session and wait for the session to expire. • Send a packet that matches the former TCP session. • Verify that the Firewall logs the TCP packet similar to former session.
Pass/Fail with Explanation	Pass. Any TCP packet matching the former expired session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

6.139 FFW_RUL_EXT.1.5 Test #4

Item	Data
Test Assurance Activity	Test 4: The evaluator shall configure the TOE to permit and log UDP traffic. The evaluator shall establish a UDP session. Once a UDP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and destination ports) one at a time in order to verify that the altered packets are not accepted as part of the established session.

Test Steps	IPV4
	• Configure the TOE to permit and log UDP traffic.
	• Apply the filter to the TOE's interface.
	• Establish a UDP session and send data.
	• Modify each of the session attributes one at a time:
	  ○ Source address
	  ○ Destination address
	  ○ Source port
	  ○ Destination port
	• Verify the altered packets are logged by the firewall filter.
	• Verify through the packet capture that the altered packets are not accepted as part of the established session.
	
	IPV6
	
	• Configure the TOE to permit and log UDP traffic with specific source and destination addresses.
	• Apply the filter to the TOE's interface.
	• Establish a UDP session and send data.
	• Modify each of the session attributes one at a time:
	  ○ Source address
	  ○ Destination address
	  ○ Source port
	  ○ Destination port
	• Verify the altered packets are logged by the firewall filter.
	• Verify through the packet capture that the altered packets are not accepted as part of the established session.

Pass/Fail with Explanation	Pass. TOE does not accept altered packets (source and destination addresses, source and destination ports) after a UDP session is successfully established. This meets testing requirements.

6.140 FFW_RUL_EXT.1.5 Test #5

Item	Data
Test Assurance Activity	Test 5: The evaluator shall expire (i.e., reach timeout) the UDP session established per Test 4 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
Test Steps	IPV4 • Configure the TOE to permit and log UDP traffic. • Apply the filter to the TOE's interface. • Establish a UDP session and wait for the session to expire. • Send a packet that matches the former UDP session. • Verify that the Firewall logs the UDP packet similar to former session. IPV6

	• Configure the TOE to permit and log UDP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface. • Establish a UDP session and wait for the session to expire. • Send a packet that matches the former UDP session. • Verify that the Firewall logs the UDP packet similar to former session.
Pass/Fail with Explanation	Pass. Any UDP packet matching the former expired session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

6.141 FFW_RUL_EXT.1.5 Test #6

Item	Data
Test Assurance Activity	Test 6: If **ICMP** is selected, the evaluator shall configure the TOE to permit and log ICMP traffic. The evaluator shall establish a session for ICMP as defined in the TSS. Once an ICMP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, other attributes chosen in FFW_RUL_EXT.1.5) one at a time in order to verify that the altered packets are not accepted as part of the established session.

Test Steps	IPv4
	• Configure the TOE to permit and log ICMP traffic.
	• Apply the filter to the TOE's interface.
	
	• For each of the session attributes, verify the altered packets are not accepted as part of the session.
	  ○ Source address
	  ▪ Establish ICMP connection.
	  ▪ Modify session attribute.
	  ▪ Verify the altered packets are logged by the firewall filter.
	  ▪ Verify the altered packets are not accepted as part of the established session.
	
	  ○ Destination address
	  ▪ Establish ICMP connection.
	  ▪ Modify session attribute.
	  ▪ Verify the altered packets are logged by the firewall filter.
	  ▪ Verify the altered packets are not accepted as part of the established session.
	
	  ○ Type
	  ▪ Establish ICMP connection.
	  ▪ Modify session attribute.
	  ▪ Verify the altered packets are logged by the firewall filter.
	  ▪ Verify the altered packets are not accepted as part of the established session.

	○ Code ▪ Establish ICMP connection. ▪ Modify session attribute. ▪ Verify the altered packets are logged by the firewall filter. ▪ Verify the altered packets are not accepted as part of the established session. IPv6 • Configure the TOE to permit and log ICMP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface. • For each of the session attributes, verify the altered packets are not accepted as part of the session. ○ Source address ▪ Establish ICMP connection. ▪ Modify session attribute. ▪ Verify the altered packets are logged by the firewall filter. ▪ Verify the altered packets are not accepted as part of the established session. ○ Destination address ▪ Establish ICMP connection. ▪ Modify session attribute. ▪ Verify the altered packets are logged by the firewall filter. ▪ Verify the altered packets are not accepted as part of the established session. ○ Type ▪ Establish ICMP connection. ▪ Modify session attribute. ▪ Verify the altered packets are logged by the firewall filter. ▪ Verify the altered packets are not accepted as part of the established session. ○ Code ▪ Establish ICMP connection. ▪ Modify session attribute. ▪ Verify the altered packets are logged by the firewall filter. ▪ Verify the altered packets are not accepted as part of the established session.
Pass/Fail with Explanation	Pass. TOE does not accept altered packets (source and destination addresses, type and code) after a ICMP session is successfully established. This meets testing requirements.

6.142 FFW_RUL_EXT.1.5 Test #7

Item	Data
Test Assurance Activity	Test 7: If applicable, the evaluator shall terminate the ICMP session established per Test 6 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
Test Steps	IPV4 • Configure the TOE to permit and log ICMP traffic • Apply the filter to the TOE's interface • Establish a ICMP session and terminate it • Send a packet that matches the former ICMP session • Verify that the Firewall logs the ICMP packet similar to former session. IPV6 • Configure the TOE to permit and log ICMP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface • Establish a ICMP session and terminate it • Send a packet that matches the former ICMP session • Verify that the Firewall logs the ICMP packet similar to former session.
Pass/Fail with Explanation	Pass. Any packet matching the ICMP former session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

6.143 FFW_RUL_EXT.1.5 Test #8

Item	Data
Test Assurance Activity	Test 8: The evaluator shall expire (i.e., reach timeout) the ICMP session established per Test 6 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
Test Steps	IPV4 • Configure the TOE to permit and log ICMP traffic • Apply the filter to the TOE's interface • Establish a ICMP session and wait for the session to expire • Send a packet that matches the former ICMP session • Verify that the Firewall logs the ICMP packet similar to former session. IPV6 • Configure the TOE to permit and log ICMP traffic with specific source and destination addresses. • Apply the filter to the TOE's interface • Establish a ICMP session and wait for the session to expire. • Send a packet that matches the former ICMP session

	• Verify that the Firewall logs the ICMP packet similar to former session.
Pass/Fail with Explanation	Pass. Any ICMP packet matching the former expired session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

6.144 FFW_RUL_EXT.1.6 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall test each of the conditions for automatic packet rejection in turn. In each case, the TOE should be configured to allow all network traffic and the evaluator shall generate a packet or packet fragment that is to be rejected. The evaluator shall use packet captures to ensure that the unallowable packet or packet fragment is not passed through the TOE.

Test Steps	IPV4
	• Packets which are invalid fragments
	  ○ Create a filter to reject and log invalid fragments.
	  ○ Send packets which are invalid fragments.
	  ○ Verify through logs that the traffic is rejected.
	  ○ Verify through Packet Capture that the traffic is rejected.
	
	• Fragments that cannot be completely re-assembled
	  ○ Create a filter to log fragments that cannot be re-assembled.
	  ○ Send fragments that cannot be re-assembled.
	  ○ Verify through logs that the traffic is rejected.
	  ○ Verify through Packet Capture that the traffic is rejected.
	
	• Packets where the source address is defined as being on a broadcast network.
	  ○ Create a filter to log traffic where the source address is defined as being on a broadcast network.
	  ○ Apply filter to the security zone associated to TOE'S interface.
	  ○ Send traffic where the source address is defined as being on a broadcast network.
	  ○ Verify through logs that the traffic is rejected.
	  ○ Verify through Packet Capture that the traffic is rejected.
	
	• Packets where the source address is defined as being on a multicast network
	  ○ Create a filter to log traffic where the source address is defined as being on a multicast network.
	  ○ Apply filter to the security zone associated to TOE'S interface.
	  ○ Send traffic where the source address is defined as being on a multicast network.
	  ○ Verify through logs that the traffic is rejected.
	  ○ Verify through Packet Capture that the traffic is rejected.
	
	• Packets where the source address is defined as being a loopback address

	○ Create a filter to log traffic where the source address is defined as being on a loopback address. ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic where the source address is defined as being on a loopback address. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) ○ Create a filter to log traffic where packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic with source address matching unspecified address and reserved for further use ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified ○ Create a filter to log traffic with packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified. ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic with IP options: Loose Source Routing, Strict Source Routing, or Record Route. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Other packets defined in FFW_RUL_EXT.1.6- No other rules defined. IPV6: • Packets which are invalid fragments. ○ Create a filter to reject and log invalid fragments. ○ Send packets which are invalid fragments. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Fragments that cannot be completely re-assembled. ○ Create a filter to log fragments that cannot be re-assembled. ○ Send fragments that cannot be re-assembled. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets where the source address is defined as being on a broadcast network.

	○ Create a filter to log traffic where the source address is defined as being on a broadcast network. ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic where the source address is defined as being on a broadcast network. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets where the source address is defined as being on a multicast network. ○ Create a filter to log traffic where the source address is defined as being on a multicast network. ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic where the source address is defined as being on a multicast network. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets where the source address is defined as being a loopback address. ○ Create a filter to log traffic where the source address is defined as being on a loopback address. ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic where the source address is defined as being on a loopback address. ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected. • Packets where the source or destination address of the network packet is defined as being unspecified or an address "reserved for future use" ○ Create a filter to log traffic where packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" ○ Apply filter to the security zone associated to TOE'S interface. ○ Send traffic with source address matching unspecified address and reserved for further use ○ Verify through logs that the traffic is rejected. ○ Verify through Packet Capture that the traffic is rejected.
Pass/Fail with Explanation	Pass. Unallowable packet or packet fragment is rejected and logged through the TOE automatically. This meets the testing requirements.

6.145 FFW_RUL_EXT.1.6 Test #2

Item	Data
Test Assurance Activity	Test 2: For each of the cases above, the evaluator shall use any applicable guidance to enable dropped packet logging or counting. In each case above, the evaluator shall ensure that the rejected packet or packet fragment was recorded (either logged or an appropriate counter incremented).

Pass/Fail with Explanation	Pass. The requirements of this test have been completed as part of testing for FFW_RUL_EXT.1.6 Test #1.

6.146 FFW_RUL_EXT.1.7 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall configure the TOE to drop and log network traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received. The evaluator shall generate suitable network traffic to match the configured rule and verify that the traffic is dropped, and a log message generated.
Test Steps	Ipv4 • Configure a filter to log and drop traffic when the source address of the packet matches the address of the network interface • Apply the filter on TOE interface. • Generate and send traffic that matches the created filter. • Verify through the firewall filter that the traffic was denied. • Verify through a packet capture that the traffic was denied. Ipv6 • Configure a filter to log and drop traffic when the source address of the packet matches the address of the network interface. • Apply the filter on TOE interface. • Generate and send traffic that matches the created filter. • Verify through the firewall filter that the traffic was denied. • Verify through a packet capture that the traffic was denied.
Pass/Fail with Explanation	Pass. The TOE drops and logs network traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received. This meets testing requirements

6.147 FFW_RUL_EXT.1.7 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall configure the TOE to drop and log network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted, e.g. if the TOE believes that network 192.168.1.0/24 is reachable through interface 2, network traffic with a source address from the 192.168.1.0/24 network should be generated and sent to an interface other than interface 2. The evaluator shall verify that the network traffic is dropped, and a log message generated.
Test Steps	Ipv4 • Configure a filter to drop and log network traffic when the source IP address of the packet does not match the interface it was received on. • Apply the filter to the TOE's interface.

	• Modify traffic to send to the TOE. • Verify through the logs and a packet capture that the traffic is dropped. • Verify the drop packets via packet capture. Ipv6 • Configure a filter to drop and log network traffic when the source IP address of the packet does not match the interface it was received on. • Apply the filter to the TOE's interface. • Modify traffic to send to the TOE. • Verify through the logs and a packet capture that the traffic is dropped. • Verify the drop packets via packet capture.
Pass/Fail with Explanation	Pass. TOE drops and logs network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted. This meets the testing requirements.

6.148 FFW_RUL_EXT.1.8 Test #1

Item	Data
Test Assurance Activity	Test 1: If the TOE implements a mechanism that ensures that no conflicting rules can be configured, the evaluator shall try to configure two conflicting rules and verify that the TOE rejects the conflicting rule(s). It is important to verify that the mechanism is implemented in the TOE but not in the non-TOE environment. If the TOE does not implement a mechanism that ensures that no conflicting rules can be configured, the evaluator shall devise two equal stateful traffic filtering rules with alternate operations – permit and drop. The rules should then be deployed in two distinct orders and in each case the evaluator shall ensure that the first rule is enforced in both cases by generating applicable packets and using packet capture and logs for confirmation. **TD0545 applied**
Test Steps	IPV4: • Configure a filter to allow and drop packets that have the same destination-address with the allow rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is allowed. • Verify allowed traffic via packet capture. • Configure a filter to drop and allow packets that have the same destination-address with the drop rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is discarded. • Verify via packet capture discarded traffic. IPV6

	• Configure a filter to allow and drop packets that have the same destination-address with the allow rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is allowed. • Verify allowed traffic via packet capture. • Configure a filter to drop and allow packets that have the same destination-address with the drop rule being first. • Apply the filter to the TOE Interface. • Send traffic to configured destination address in filter. • Verify through the firewall log that traffic is discarded. • Verify via packet capture discarded traffic.
Pass/Fail with Explanation	Pass. TOE enforces the first rule in the firewall filter. This meets the testing requirement.

6.149 FFW_RUL_EXT.1.8 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall repeat the procedure above, except that the two rules should be devised where one is a subset of the other (e.g., a specific address vs. a network segment). Again, the evaluator should test both orders to ensure that the first is enforced regardless of the specificity of the rule.
Test Steps	IPV4: • Configure the firewall rule order to allow packets to a specific destination-address and deny packets to its network segment. • Apply the filter to the TOE Interface. • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that only traffic to specific destination address are allowed and remaining addresses to network segment are discarded. • Verify the rules applied through Packet Capture. • Configure the firewall rule order to deny packets to a network segment and allow packets to a specific destination-address of the network segment. • Apply the filter to the TOE Interface • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that all traffic is dropped. • Verify the rules applied through Packet Capture. IPV6: • Configure the firewall rule order to allow packets to a specific destination-address and deny packets to its network segment.

	• Apply the filter to the TOE Interface. • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that only traffic to specific destination address are allowed and remaining addresses to network segment are discarded. • Verify the rules applied through Packet Capture. • Configure the firewall rule order to deny packets to a network segment and allow packets to a specific destination-address of the network segment. • Apply the filter to the TOE Interface • Send traffic to configured specific destination and network segment addresses. • Verify through the firewall logs that all traffic is dropped. • Verify the rules applied through Packet Capture.
Pass/Fail with Explanation	Pass. TOE enforces the first rule in the firewall filter. This meets the testing requirement.

6.150 FFW_RUL_EXT.1.9 Test #1

Item	Data
Test Assurance Activity	For each attribute in FFW_RUL_EXT.1.2, the evaluator shall construct a test to demonstrate that the TOE can correctly compare the attribute from the packet header to the ruleset, and shall demonstrate both the permit and deny for each case. It shall also be verified that a packet is dropped if no matching rule can be identified for the packet. The evaluator shall check the log in each case to confirm that the relevant rule was applied. The evaluator shall record a packet capture for each test to demonstrate the correct TOE behavior.
Pass/Fail with Explanation	Pass. This test has been completed as part of FFW_RUL_EXT.1.2.

6.151 FFW_RUL_EXT.1.10 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall define a TCP half-open connection limit on the TOE. The evaluator shall generate TCP SYN requests to pass through the TOE to the target system using a randomised source IP address and common destination IP address. The number of SYN requests should exceed the TCP half-open threshold defined on the TOE. TCP SYN-ACK messages should not be acknowledged. The evaluator shall verify through packet capture that once the defined TCP half-open threshold has been reached, subsequent TCP SYN packets are not transmitted to the target system. The evaluator shall verify that when the configured threshold is reached that, depending upon the selection, either a log entry is generated, or a counter is incremented.
Test Steps	IPV4: • Configure the TOE to limit the amount of half-open TCP connections. • Apply the configuration to the TOE's interface. • Send continuous traffic to the TOE. • Verify that when the configured threshold is reached a log entry is generated and a counter is incremented.

	• Verify with logs. • Verify with packet capture. IPV6: • Configure the TOE to limit the amount of half-open TCP connections. • Apply the configuration to the TOE's interface. • Send continuous traffic to the TOE. • Verify that when the configured threshold is reached a log entry is generated and a counter is incremented. • Verify with logs. • Verify with packet capture.
Pass/Fail with Explanation	Pass. Half Open TCP SYN packets are not acknowledged by the TOE. When the configured threshold is reached, a log entry is generated by the TOE. This meets the testing requirements.

6.152 FFW_RUL_EXT.2.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall define stateful traffic filtering rules to permit and log traffic for each of the supported protocols and drop and log TCP and UDP ports above 1024. Subsequently, the evaluator shall establish a connection for each of the selected protocols in order to ensure that it succeeds. The evaluator shall examine the generated logs to verify they are consistent with the guidance documentation.
Test Steps	**TCP** • Configure the TOE to drop and log TCP ports above 1024. • Apply the filter to the TOE's interface. • Establish a TCP connection with port 1023. • Verify through the firewall log that the connection is successful. • Verify through a packet capture that the connection is successful. • Establish a TCP connection with port 1025. • Verify through the firewall log that the connection is unsuccessful. • Verify through a packet capture that the connection is unsuccessful. **UDP** • Configure the TOE to drop and log UDP ports above 1024. • Apply the filter to the TOE's interface. • Establish a UDP connection with port 1022. • Verify through the firewall log that the connection is successful. • Verify through a packet capture that the connection is established. • Establish a UDP connection with port 1026. • Verify through the firewall log that the connection is unsuccessful.

	• Verify through a packet capture that the connection is not established.
Pass/Fail with Explanation	Pass. The TOE permits and logs traffic for each of the supported protocols and drops and logs traffic with TCP and UDP ports above 1024.

6.153 FFW_RUL_EXT.2.1 Test #2

Item	Data
Test Assurance Activity	Test 2: Continuing from Test 1, the evaluator shall determine (e.g., using a packet sniffer) which port above 1024 opened by the control protocol, terminate the connection session, and then verify that TCP or UDP (depending on the protocol selection) packets cannot be sent through the TOE using the same source and destination addresses and ports.
Test Steps	TCP • Configure the TOE to drop and log TCP ports above 1024. • Apply the filter to the TOE's interface. • Establish a supported TCP connection then terminate the session. • Note the port opened by the Control Protocol for the former connection. • Send a packet matching the former TCP session. • Verify through the firewall log and packet capture that the TOE logs the packet • Verify through the packet capture that the connection is rejected. UDP • Configure the TOE to drop and log UDP ports above 1024. • Apply the filter to the TOE's interface. • Establish a supported UDP connection then terminate the session. • Note the port opened by the Control Protocol for the former connection. • Send a packet matching the former UDP session. • Verify through the firewall log and packet capture that the TOE logs the packet • Verify through the packet capture that the connection is rejected.
Pass/Fail with Explanation	Pass. TCP or UDP packets using the same source and destination addresses and ports as former sessions are not sent through the TOE. This meets the testing requirements.

6.154 FFW_RUL_EXT.2.1 Test #3

Item	Data
Test Assurance Activity	Test 3: For each additionally supported protocol, the evaluator shall repeat the procedure above for the protocol. In each case the evaluator must use the applicable RFC or standard in order to determine what range of ports to block in order to ensure the dynamic rules are created and effective.
Pass/Fail with Explanation	Pass. No additional supported protocol selected. Selected FTP protocol is covered in FFW_RUL_EXT.2.1 Test #1 and FFW_RUL_EXT.2.1 Test #2

7 Security Assurance Requirements

7.1 ADV_FSP.1 Basic Functional Specification

7.1.1 ADV_FSP.1

7.1.1.1 ADV_FSP.1 Activity 1

Objective	The evaluator shall examine the interface documentation to ensure it describes the purpose and method of use for each TSFI that is identified as being security relevant.
Evaluator Findings	The evaluator examined the AGD (interface documentation) to verify that it describes the purpose and method of use for each TSFI that is identified as being security relevant. The evaluator examined the entire AGD. The evaluator verified the AGD describes the purpose and method of use for each security relevant TSFI by verifying the AGD satisfies all of the Guidance Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.1.1.2 ADV_FSP.1 Activity 2

Objective	The evaluator shall examine the interface documentation to ensure it describes the purpose and method of use for each TSFI that is identified as being security relevant.
Evaluator Findings	The evaluator examined the AGD (interface documentation) to develop a mapping of the interfaces to SFRs. The evaluator examined the entire AGD. Each Guidance Evaluation Activity is associated with a specific SFR. The Evaluation Findings for each Guidance Evaluation Activity identify the relevant interfaces, thus providing a mapping. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.1.1.3 ADV_FSP.1 Activity 3

Objective	The evaluator shall check the interface documentation to ensure it identifies and describes the parameters for each TSFI that is identified as being security relevant.
Evaluator Findings	The evaluator examined the AGD (interface documentation) to verify that it identifies and describes the parameters for each TSFI that is identified as being security relevant. The evaluator examined the entire AGD. The evaluator verified the AGD describes the parameters for each security relevant TSFI by verifying the AGD satisfies all of the Guidance Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.2 AGD_OPE.1 Operational User Guidance

7.2.1 AGD_OPE.1

7.2.1.1 AGD_OPE.1 Activity 1

Objective	The evaluator shall ensure the Operational guidance documentation is distributed to Security Administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that Security Administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration.

Evaluator Findings	The evaluator checked the requirements below are met by the guidance documentation. Guidance documentation shall be distributed to administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration. Upon investigation, the evaluator found that the CC guidance will be published with the CC certificate on www.niap-ccevs.org.. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.2.1.2 AGD_OPE.1 Activity 2

Objective	The evaluator shall ensure that the Operational guidance is provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.
Evaluator Findings	The evaluator ensured that the Operational guidance is provided for every Operational Environment that the product supports as claimed in the Security Target. The section titled Supported Platforms of the AGD was used to determine the verdict of this assurance activity. The AGD specifies that the platforms supported are: • NFX350-S1 • NFX350-S2 • NFX350-S3 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.2.1.3 AGD_OPE.1 Activity 3

Objective	The evaluator shall ensure that the Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. 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 TOE.
Evaluator Findings	The evaluator ensured that the Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. While performing the Guidance Evaluation Activities for the cryptographic SFRs, the evaluator ensured guidance contained the necessary instructions for configuring the cryptographic engines. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.2.1.4 AGD_OPE.1 Activity 4

Objective	The evaluator shall ensure the Operational guidance makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs.
Evaluator Findings	The entire AGD was used to determine the verdict of this work unit. Each confirmation command indicates tested options. Additionally, the section titled Understanding Protocol Support and Unsupported Junos-FIPS Configuration Statements specifies features that are not assessed and tested by the EAs. The evaluator ensured the Operational guidance makes it

	clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.2.1.5 AGD_OPE.1 Activity 5 **[TD0536]**

Objective	In addition, the evaluator shall ensure that the following requirements are also met. a) The guidance documentation shall contain instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. 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 TOE. b) The documentation must describe the process for verifying updates to the TOE for each method selected for FPT_TUD_EXT.1.3 in the Security Target. The evaluator shall verify that this process includes the following steps: i) Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory). ii) Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes instructions that describe at least one method of validating the hash/digital signature. c) The TOE will likely contain security functionality that does not fall in the scope of evaluation under this cPP. The guidance documentation shall make it clear to an administrator which security functionality is covered by the Evaluation Activities.
Evaluator Findings	The evaluator verified the guidance documentation contains instructions for configuring any cryptographic engines in AGD_OPE.1 Test #3. The evaluator verified the guidance documentation describes the process for verifying updates in FPT_TUD_EXT.1 Guidance 2. The evaluator verified the guidance documentation makes it clear which security functionality is covered by the Evaluation Activities in AGD_OPE.1 Test #4. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.3 AGD_PRE.1 Preparative Procedures

7.3.1 AGD_PRE.1

7.3.1.1 AGD_PRE.1 Activity 1

Objective	The evaluator shall examine the Preparative procedures to ensure they include a description of how the Security Administrator verifies that the operational environment can fulfil its role to support the security functionality (including the requirements of the Security Objectives for the Operational Environment specified in the Security Target).
Evaluator Findings	The evaluator examined the Preparative procedures to ensure they include a description of how the administrator verifies that the operational environment can fulfil its role to support the security functionality. The evaluator reviewed the sections titled **FIPS User Role and Responsibilities** of the AGD. The evaluator found that these sections describe how the Operational Environment must meet:

	• OE.PHYSICAL • OE.NO_GENERAL_PURPOSE • OE.NO_THRU_TRAFFIC_PROTECTION • OE.TRUSTED_ADMN • OE.UPDATES • OE.ADMIN_CREDENTIALS_SECURE • OE.RESIDUAL_INFORMATION • OE.CONNECTIONS (IPS) • OE.CONNECTIONS (VPN) Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

7.3.1.2 AGD_PRE.1 Activity 2

Objective	The evaluator shall examine the Preparative procedures to ensure they are provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.
Evaluator Findings	The evaluator checked the requirements below are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that the guidance documentation describes each of the devices in the operating environment, including, • Syslog Server • CRL Server • SSH Client • Management Console • IPsec Peer The section titled **Supported Platforms** of AGD identifies the following supported platform: • NFX350-S1 • NFX350-S2 • NFX350-S3 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.3.1.3 AGD_PRE.1 Activity 3

Objective	The evaluator shall examine the preparative procedures to ensure they include instructions to successfully install the TSF in each Operational Environment.
Evaluator Findings	The evaluator checked the requirements are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that AGD describes all of the functions necessary to install and configure the TOE to work in the target operating environment, including, • Insert list of functions, such as • Configuring Administrative Accounts and Passwords • Configuring SSH and Console Connections

	• Configuring the Remote Syslog Server • Configuring Audit Log Options • Configuring Event Logging • Configuring a Secure Logging Channel • Configuring VPNs (IPsec) • Configuring Security Flow Policies • Configuring Traffic Filtering Rules • Configuring Network Attacks Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.3.1.4 AGD_PRE.1 Activity 4

Objective	The evaluator shall examine the preparative procedures to ensure they include instructions to manage the security of the TSF as a product and as a component of the larger operational environment.
Evaluator Findings	The evaluator ensured the preparative procedures include instructions to manage the security of the TSF as a product and as a component of the larger operational environment. The entire AGD was used to determine the verdict of this work unit. The same commands, configurations, and interfaces used to install the TOE are also used for ongoing management, so this is satisfied by AGD_PRE.1 Test #3. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.3.1.5 AGD_PRE.1 Activity 5

Objective	In addition, the evaluator shall ensure that the following requirements are also met. The preparative procedures must a) include instructions to provide a protected administrative capability; and b) identify TOE passwords that have default values associated with them and instructions shall be provided for how these can be changed.
Evaluator Findings	The evaluator ensured the preparative procedures include instructions to provide a protected administrative capability and changing default passwords. The sections titled **Configuring a Network Device collaborative Protection Profile for an Authorized Administrator** and **Configuring SSH and Console Connection** were used to determine the verdict of this work unit. The AGD describes changing the default password associated with the root account and configuring SSH for remote administration. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.4 ALC Assurance Activities

7.4.1 ALC_CMC.1

7.4.1.1 ALC_CMC.1 Activity 1

Objective	When evaluating that the TOE has been provided and is labelled with a unique reference, the evaluator performs the work units as presented in the CEM.
Evaluator Findings	The evaluator verified that the ST, TOE and Guidance are all labeled with the same hardware versions and software. The information is specific enough to procure the TOE and it includes hardware models and software versions. The evaluator checked the TOE software version and hardware identifiers during testing by examining the actual machines used for testing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.4.2 ALC_CMS.1

7.4.2.1 ALC_CMS.1 Activity 1

Objective	When evaluating the developer's coverage of the TOE in their CM system, the evaluator performs the work units as presented in the CEM.
Evaluator Findings	The evaluator verified that the ST, TOE and Guidance are all labeled with the same hardware versions and software. The information is specific enough to procure the TOE and it includes hardware models and software versions. The evaluator checked the TOE software version and hardware identifiers during testing by examining the actual machines used for testing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.5 ATE_IND.1 Independent Testing – Conformance

7.5.1 ATE_IND.1

7.5.1.1 ATE_IND.1 Activity 1

Objective	The evaluator performs the CEM work units associated with the ATE_IND.1 SAR. Specific testing requirements and EAs are captured for each SFR in Sections 2, 3 and 4. The evaluator should consult Appendix 709 when determining the appropriate strategy for testing multiple variations or models of the TOE that may be under evaluation.
Evaluator Findings	The evaluator examined the TOE to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST. Upon investigation, the evaluator found that each instance of the TOE used in testing was consistent with TOE description found in the Security Target. Additionally, the evaluator found that the TOE version is consistent with what was specified in the Security Target. The evaluator examined the TOE to determine that it has been installed properly and is in a known state. The details of the installed TOE and any configuration performed with the TOE are found in the separate Test Reports. The evaluator prepared a test plan that covers all of the testing actions for ATE_IND.1 in the CEM and in the SFR-related Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

7.6 AVA_VAN.1 Vulnerability Survey

7.6.1 AVA_VAN.1

7.6.1.1 AVA_VAN.1 Activity 1 **[TD0564, Labgram #116]**

Objective	The evaluator shall document their analysis and testing of potential vulnerabilities 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 web sites listed below. The sources of the publicly available information are provided below. • http://nvd.nist.gov/vuln/search • http://cve.mitre.org/cve/search_cve_list.html • https://www.cvedetails.com/ • https://www.kb.cert.org/vuls/search/ • http://www.zerodayinitiative.com/advisories • https://www.exploit-db.com • https://www.rapid7.com/db/vulnerabilities • https://kb.juniper.net The evaluator performed the public domain vulnerability searches using the following key words. The search was performed on June 9, 2022. • JunOS 20.3 • NFX350 • Intel Xeon D-2146NT • Intel Xeon D-2166NT • Intel Xeon D-2187NT • FreeBSD 11 • Junos OS Kernel • Junos OS Dataplane libcrypto • Junos OS Control plane libcrypto • Junos OS libmd • Junos OS libquicksec • Junos OS openssl • IPv4 • IPv6 • TCP • UDP • ICMP • Wind River Linux 8 (WRL8) • IPsec • Firewall • VPN • IPS

	The evaluation lab examined each result provided from the above sources to determine if the current TOE version or component within the environment was vulnerable. Based upon the analysis, any issues found that were generated were patched in the TOE version and prior versions, mitigating the risk factor. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

7.6.1.2 AVA_VAN.1 Activity 2

Objective	The evaluator shall perform the following activities to generate type 4 flaw hypotheses:
	• Fuzz testing
	  ○ Examine effects of sending:
	  ▪ mutated packets carrying each 'Type' and 'Code' value that is undefined in the relevant RFC for each of ICMPv4 (RFC 792) and ICMPv6 (RFC 4443)
	  ▪ mutated packets carrying each 'Transport Layer Protocol' value that is undefined in the respective RFC for IPv4 (RFC 791) IPv6 (RFC 2460) should also be covered if it is supported and claimed by the TOE.
	
	  Since none of these packets will belong to an allowed session, the packets should not be processed by the TOE, and the TOE should not be adversely affected by this traffic. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.
	
	  ○ Mutation fuzz testing of the remaining fields in the required protocol headers. This testing requires sending mutations of well- formed packets that have both carefully chosen and random values inserted into each header field in turn (i.e. testing is to include both carefully chosen and random insertion test cases). The original well-formed packets would be accepted as part of a normal existing communication stream and may still be accepted as valid packets when subject to the carefully chosen mutations (the individual packet alone would be valid although its contents may not be valid in the context of preceding and/or following packets), but will often not be valid packets when random values are inserted into fields. The carefully chosen values should include semantically significant values that can be determined from the type of the data that the field represents, such as values indicating positive and negative integers, boundary conditions, invalid binary combinations (e.g. for flag sets with dependencies between bits), and missing start or end values. Randomly chosen values may not result in well-formed packets but are included nonetheless to see whether they can lead to the device entering an insecure state. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.

Evaluator Findings	The evaluator documented the fuzz testing results with respect to this requirement. The evaluation lab examined each result from fuzz testing to determine if the TOE improperly processes packets. Based upon the analysis, no unexpected results occurred. Therefore, no Type 4 hypotheses were generated. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

8 Conclusion

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

End of Document