CACI idt GoSilent Server and Cube v26.01 Security Target

Document Version: 1.2

Document Date: 22 July 2026

Authored By: CACI idt

2342 Richmond Hwy

Arlington, VA 22202

Revision History

Version	Date	Changes
Version 0.1	June 27, 2025	Initial draft
Version 0.2	July 7, 2025	Further content included
Version 0.3	July 17, 2025	Addressed evaluator comments
Version 0.4	August 18, 2025	Addressed evaluator Ors
Version 0.5	October 21, 2025	Added CAVP information, updated CDP info
Version 0.6	November 3, 2025	Addressed evaluator Ors, no session resumption
Version 0.7	December 4, 2025	Added ESV certificate info
Version 0.8	March 23, 2026	Applied latest TDs, completed FCS_RBG_EXT.1
Version 1.0	April 7, 2026	Addressed validator comments
Version 1.1	May 8, 2026	Addressed evaluator comments
Version 1.2	July 22, 2026	Corrected AGD reference

Table of Contents

List of Figures

List of Tables

1 Introduction

This Security Target (ST) defines the CACI idt GoSilent Server and Cube v26.01 Target of Evaluation (TOE) for the purposes of Common Criteria (CC) evaluation.

The GoSilent Server and Cube v26.01 provides firewall and VPN services in a portable form factor to secure the connectivity and remote communications of one or more network devices.

1.1 Security Target and TOE Reference

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

Table 1: TOE and ST Identification

Category	Identifier
ST Title	CACI idt GoSilent Server and Cube v26.01 Security Target
ST Version	1.2
ST Date	22 July 2026
ST Author	CACI idt
TOE Identifier	GoSilent Server and Cube v26.01
TOE Version	v26.01
TOE Developer	CACI idt
Key Words	IPsec, VPN Gateway, Firewall

1.2 TOE Type

The TOE is a distributed TOE which consists of the CACI idt GoSilent Server and Cube operating together as a single solution to provide firewall and VPN capabilities for remote network devices to secure their communications. The GoSilent Cube is a hardware user device and the GoSilent Server is a virtualized appliance. The TOE conforms to distributed TOE Use Case 3 as defined in the *collaborative Protection Profile for Network Devices,* Version 3.0e, whereby a network device requires a management component to satisfy all SFRs.

1.3 TOE Overview

The GoSilent Server acts as the centralized management and external access system for the TOE integrated into one system. The GoSilent Server provides the management GUI for configuration of GoSilent Server as well as the GoSilent Cube user devices. Operationally, it acts as the central peer for all IPsec connections from the GoSilent Cube devices and provides gateway connectivity to external systems located behind a GoSilent Cube.

The GoSilent Cube is a portable enterprise-grade firewall and VPN, ideal for sensitive communications, secure remote network access, and Internet of Things (IoT) deployments. GoSilent Cube can be setup within minutes by non-technical users. Physical Ethernet connections are supported on both the user side and VPN side of GoSilent Cube.

Together, GoSilent Cube and GoSilent Server provide a secure communications path to one or more systems located "behind" each GoSilent Cube. These systems connect to GoSilent Cube via physical Ethernet. Each GoSilent Cube establishes an IPsec VPN to the GoSilent Server, and all traffic from the user systems is routed over that VPN. Physical Ethernet is supported on the VPN side of GoSilent Cube.

The TOE applies policies to restrict the traffic that is permitted to pass between the user systems and external networks.

Management of the system is performed via a GUI provided by GoSilent Server, accessed via a browser on remote PCs using TLS/HTTPS. Authorized administrators may configure GoSilent Server as well as the GoSilent Cubes.

GoSilent Cubes also provide a GUI accessed from a browser on a user system via a TLS/HTTPS connection. This GUI only provides authorized administrators with the ability to configure that specific GoSilent Cube with enough information to connect to GoSilent Server. All additional configuration information is downloaded from GoSilent Server once the IPsec VPN is established.

Both GoSilent Server and GoSilent Cube generate audit records that are stored locally as well as sent to a remote syslog server via a TLS connection.

1.4 Usage and Major Security Features of the TOE

The TOE conforms to the *collaborative Protection Profile for Network Devices,* Version 3.0e, *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e and *PP-Module for Virtual Private Network Gateways*, Version 1.3.

The use case for a product conforming to these documents is to protect data transmitted across an untrusted network.

1.5 TOE Description

1.5.1 Evaluated Configuration

The GoSilent Server is a software TOE/virtual network device (vND) while GoSilent Cube is a physical network device (pND). The following platforms apply to these components.

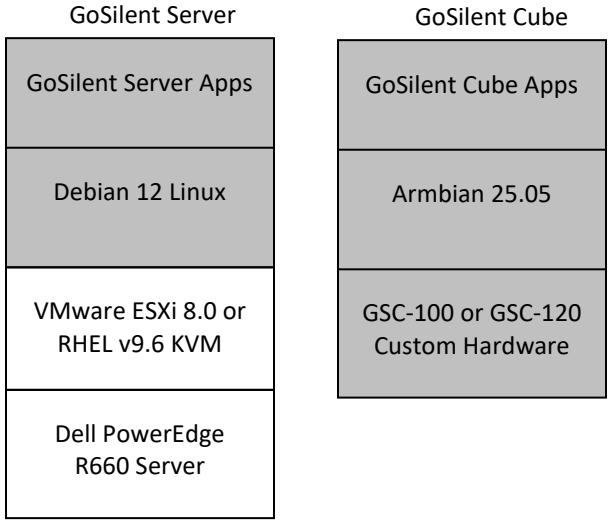
Table 2: GoSilent Server and Cube v26.01 Platforms

TOE Component	Hardware Model	CPU	CPU Microarchitecture	Virtualization	OS
GoSilent Server	Dell PowerEdge R660 Server	Intel Xeon Silver 4514Y	Raptor Cove	VMware ESXi v8.0	Debian 12
				Red Hat RHEL v9.6 KVM	
GoSilent Cube	CACI idt GSC-100	AllWinner H5/ Cortex A-53	ARM v8-A	N/A	Armbian 25.05
	CACI idt GSC-120				

1.5.2 Physical Boundary

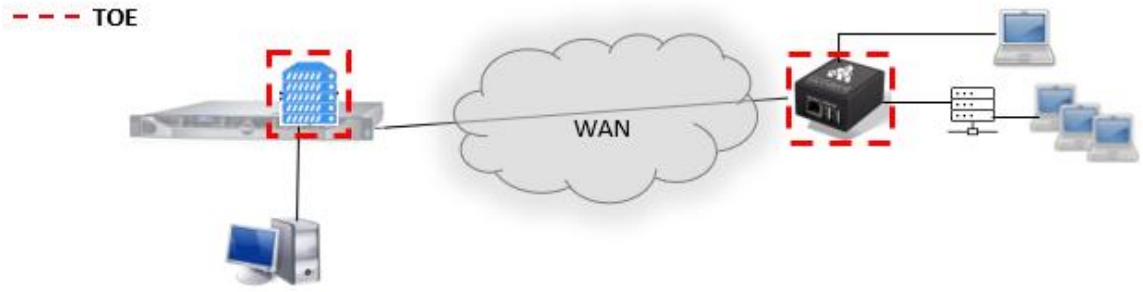
The physical boundaries of the TOE are the vND of the GoSilent Server virtual appliance and the custom physical enclosure of the GoSilent Cube pND. Figure 1 below illustrates the boundaries of the TOE (shaded areas in the figure).

Figure 1: TOE Boundary



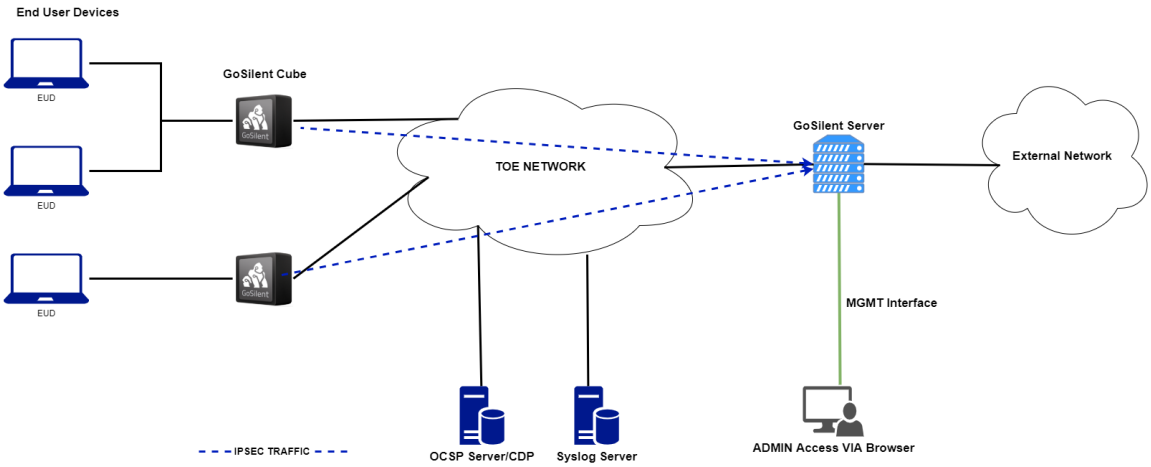
The TOE is deployed in a distributed configuration with the GoSilent Server virtual appliance installed on a virtualization platform located at the enterprise destination and the GoSilent Cube at the user device side. The GoSilent Cube can be connected between a specific device and the LAN, or between an access switch with multiple user devices connected and the WAN. Figure 2 below shows the physical TOE deployment and the logical TOE boundary.

Figure 2: Representative TOE Deployment



The TOE management interfaces are depicted in Figure 3.

Figure 3: TOE Interfaces



The TOE interfaces are as follows:

- **GUI (GoSilent Server).** Administrative GUI interface for managing GoSilent Server configurations and GoSilent Cubes. The GUI can be accessed via the GoSilent Server MGMT interface or from GoSilent Cube clients whose users are authorized administrators on GoSilent Server. Console access via the virtualization component is not included in the evaluation.
- **GUI (GoSilent Cube).** User GUI interface that provides limited functionality for initial cube configuration and authorized admin access to the GoSilent Server GUI.
- **Remote Logging.** Forwards TOE audit events to a syslog server for storage.
- **OCSP/CRL.** Certificate checking for IPSec connections are conducted by OCSP while CRLs are leveraged for syslog.
- **VPN Gateway.** VPN connections via IPSec.

The following items are required for the operational environment.

Table 3: Hardware and Software Environmental Components

Components	Mandatory/ Optional	Description
Ethernet dongle	Mandatory	Provides the WAN USB port of each GoSilent Cube for connection of the Cube to the WAN.
TOE Network	Mandatory	Provides connectivity between the TOE components. Since IPsec tunnels are layered on top of this network, the network itself is not required to be trusted.
External Network	Mandatory	GoSilent Server provides user systems with connectivity to systems on an External network, which may be limited to an Enterprise Network or provide much wider access.

Components	Mandatory/ Optional	Description
GoSilent Cube user networks	Mandatory	Physical Ethernet networks located on the user side of GoSilent Cube that provide connectivity between user systems and GoSilent Cube. In the simplest case, this is a direct connection using an Ethernet cable interconnecting the GoSilent Cube RJ45 Device port and a user system. To support multiple user systems, an Ethernet switch can be used.
Syslog Server	Mandatory	The Syslog Server receives audit messages from the TOE over a TLS connections and may be connected to the TOE Network or the External Network.
OCSP Server	Optional	Server accessed via OCSP to determine X.509 certificate revocation status for IPsec connections. The OCSP Server is connected to the TOE Network.
Certificate Distribution Points (CDP)	Optional	Servers accessed using HTTP to determine X.509 certificate revocation status for syslog connections. The CDPs may be connected to the TOE Network or the External Network.
User systems	Mandatory	IT systems using GoSilent Cube as a secure path to external systems.

The Cube devices are delivered to customers via trusted courier. The following checks should be performed upon receipt:

- Confirm that the correct device has been delivered per the packing slip and original order;
- Inspect all packaging to confirm there are no signs of tampering or damage incurred during transport.

The GoSilent Server software can be obtained digitally via URL provided to the customer directly by CACI idt.

1.5.3 TOE Documentation (Operational Guidance)

The following document is essential to understanding and controlling the TOE in the evaluated configuration and is included in the TOE physical boundary.

- [AGD] *CACI idt GoSilent Server and Cube v26.01 User Guidance*, June 2026, Version 1.1, (Can be obtained from the NIAP PCL (www.niap-ccevs.org) posting for this evaluation).

1.6 Security Functions Provided by the TOE

The TOE provides the security functions required by the *collaborative Protection Profile for Network Devices,* Version 3.0e, *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e and *PP-Module for Virtual Private Network Gateways*, Version 1.3.

1.6.1 Security Audit

The TOE generates logs for auditable events. These logs are stored locally in protected storage and are also forwarded to an external audit server. When the maximum storage utilization has been reached, audit records are rotated as described in this ST.

1.6.2 Communication

The TOE provides mechanisms for configuring, registering, and enabling components in order to establish secure communications with each other.

1.6.3 Cryptographic Support

The TOE implements key generation and other cryptographic services to protect TOE communications including data in transit and at rest.

1.6.4 User Data Protection

The TOE provides mechanisms to protect user data and prevent its persistence by overwriting storage space with zeros when memory is deallocated.

1.6.5 Firewall & Packet Filtering

The TOE provides firewall functionality for all traffic passed through the TOE by enforcing stateful network traffic filtering based on examination of network packets and the application of information flow rules.

1.6.6 Identification and Authentication

The TOE implements mechanisms to identify and authenticate all administrators to ensure only authorized access to TOE functionality or TSF data is granted.

1.6.7 Security Management

The TOE provides a suite of management functionality for each TOE component, which is only configurable and accessible by authorized administrators.

1.6.8 Protection of the TSF

The TOE implements a variety of protection mechanisms including authentication, self-tests, and trusted update functions to ensure the integrity of the TOE and that its TSF data is protected from unauthorized access.

1.6.9 TOE Access

The TOE provides session monitoring and management functions for local and remote administrative sessions.

1.6.10 Trusted Paths/Channels

The TOE provides secure channels between itself and local/remote administrators, including logging channels to ensure data in transit is protected.

1.7 Product Functionality not Included in the Scope of the Evaluation

The following product functionality is not included in the CC evaluation:

- Cloud instances of GoSilent Server. GoSilent server is supported on virtual platforms and cloud instances; however, cloud deployments are not addressed by this evaluation.

- WiFi connectivity. The Cube is capable of accepting wireless connections from a client, however no security claims are made with respect to wireless connectivity and is therefore excluded from the evaluation.

2 Conformance Claims

This section identifies the TOE conformance claims, conformance rationale, and relevant Technical Decisions (TDs).

2.1 CC Conformance Claims

This ST supports the following conformance claims:

- *Common Criteria for Information Technology Security Evaluations Part 1*, Version 3.1, Revision 5, April 2017
- *Common Criteria for Information Technology Security Evaluations Part 2*, Version 3.1, Revision 5, April 2017 (extended)
- *Common Criteria for Information Technology Security Evaluations Part 3*, Version 3.1, Revision 5, April 2017 (conformant)

2.2 Protection Profile Conformance

This ST and the TOE it describes are exact conformance to the following CC specifications:

- *PP-Configuration for Network Devices, Stateful Traffic Filter Firewalls, and Virtual Private Network Gateways, Version 2.0,* 25 April 2024 (CFG_NDcPP-FW-VPNGW_V2.0)

  This PP-Configuration includes the following components:

  - Base-PP: *collaborative Protection Profile for Network Devices*, Version 3.0e (CPP_ND_V3.0E)
  - PP-Module: *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e (MOD_CPP_FW_V1.4E)
  - PP-Module: *PP-Module for VPN Gateways*, Version 1.3 (MOD_VPNGW_V1.3)

2.3 Conformance Rationale

This ST provides exact conformance to CPP_ND_V3.0E, MOD_CPP_FW_V1.4E and MOD_VPNGW_V1.3. The security problem definition, security objectives, and security requirements in this ST are all taken from these documents, performing only the operations defined there.

2.3.1 Technical Decisions (TDs)

All NIAP TDs issued to date and applicable to CPP_ND_V3.0E, MOD_CPP_FW_V1.4E and MOD_VPNGW_V1.3 have been considered. The following table identifies all applicable TDs.

Note, if a TD pertains to an SFR claimed by the ST, it is considered applicable to the evaluation (marked as Yes in second column of the table below) even if the modification of the SFR does not directly affect the ST's SFR claim.

Table 4: Relevant Technical Decisions

Relevant Technical Decisions	Applicable to the Evaluation (Yes/No)	Notes and/or Exclusion Rationale (if applicable)
	NDcPP	
TD1033 - Sunset Dates for NDcPP Configurations	Yes	
TD0990 - NIT Technical Decision: CTR_DRBG in FCS_RBG_EXT.1.2	Yes	
TD0973 - NIT Technical Decision: FCS_(D)TLSS_EXT.1.3 Test 2 DHE Ciphersuite Conditionality	Yes	
TD0923 - NIT Technical Decision: Auditable event for FAU_STG_EXT.1 in FAU_GEN.1.2	Yes	
TD0921: NIT Technical Decision: Addition of FIPS PUB 186-5 and Correction of Assignment	Yes	
TD0900 - NIT Technical Decision: Clarification to Local Administrator Access in FIA_UIA_EXT.1.3	Yes	
TD0899 - NIT Technical Decision: Correction of Renegotiation Test for TLS 1.2	Yes	
TD0886 - Clarification to FAU_STG_EXT.1 Test 6	Yes	
TD0880 - NIT Decision: Removal of Duplicate Selection in FMT_SMF.1.1	Yes	
TD0879 - NIT Decision: Correction of Chapter Headings in CPP_ND_V3.0E	Yes	
TD0868 - NIT Technical Decision: Clarification of time frames in FCS_IPSEC_EXT.1.7 and FCS_IPSEC_EXT.1.8	Yes	The presentations from this TD are used in this ST rather than the presentations in the VPN GW Module
TD0836 - NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1	Yes	

Relevant Technical Decisions	Applicable to the Evaluation (Yes/No)	Notes and/or Exclusion Rationale (if applicable)
	FW	
TD0924: NIT Technical Decision: FFW_RUL_EXT.1.2 Expected Rule Granularity Level	Yes	
TD0827 - Aligning MOD_CPP_FW_v1.4E with CPP_ND_V3.0E	Yes	
TD0551 - NIT Technical Decision for Incomplete Mappings of OEs in FW Module v1.4+Errata	Yes	
TD0545 - NIT Technical Decision for Conflicting FW rules cannot be configured (extension of RfI#201837)	Yes	
	VPNGW	
TD1034 - Clarification on RFC 8784 in FIA_PSK_EXT.1	Yes	
TD0986 - Alignment of Updated IPsec Changes to MOD_VPNGW_V1.3	Yes	
TD0944 - Adding FIPS 186-5 in MOD_VPNGW_V1.3	Yes	
TD0941 - Updated Conformance Claims for MOD_VPNGW 1.3	Yes	
TD0824 - Aligning MOD_VPNGW 1.3 with NDcPP 3.0E	Yes	
TD0811 - Correction to Referenced SFR in FIA_PSK_EXT.3 Test	No	FIA_PSK_EXT.3 is not included in this ST
TD0781 - Correction to FIA_PSK_EXT.3 EA for MOD_VPNGW_v1.3	No	FIA_PSK_EXT.3 is not included in this ST

3 Security Problem Definition

The security problem definition has been taken directly from the claimed cPPs specified in Section 2.2 and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any Organizational Security Policies (OSPs) that the TOE is expected to enforce.

3.1 Threats

The threats included in Table 5 are drawn directly from the cPP and modules. Threats drawn from the modules have "/FW" or "/VPNG" appended to the name to ensure names are not duplicated.

Table 5: Threats

ID	Threat
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_COMMUNICATION_CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.

ID	Threat
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker. Threat agents may also be able to take advantage of weak administrative passwords to gain privileged access to the device.
T.SECURITY_FUNCTIONALITY_FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.
T.NETWORK_DISCLOSURE/FW	An attacker may attempt to "map" a subnet to determine the machines that reside on the network, and obtaining the IP addresses of machines, as well as the services (ports) those machines are offering. This information could be used to mount attacks to those machines via the services that are exported.
T.NETWORK_ACCESS/FW	With knowledge of the services that are exported by machines on a subnet, an attacker may attempt to exploit those services by mounting attacks against those services.
T.NETWORK_MISUSE/FW	An attacker may attempt to use services that are exported by machines in a way that is unintended by a site's security policies. For example, an attacker might be able to use a service to "anonymize" the attacker's machine as they mount attacks against others.
T.MALICIOUS_TRAFFIC/FW	An attacker may attempt to send malformed packets to a machine in hopes of causing the network stack or services listening on UDP/TCP ports of the target machine to crash.
T.DATA_INTEGRITY/VPNG	Devices on a protected network may be exposed to threats presented by devices located outside the protected network that may attempt to modify the data without authorization. If known malicious external devices are able to communicate with devices on the protected network or if devices on the protected network can communicate with those external

ID	Threat
	devices then the data contained in the communications may be susceptible to a loss of integrity.
T.NETWORK_ACCESS/VPNG	Devices located outside the protected network may seek to exercise services located on the protected network that are intended to only be accessed from inside the protected network or only accessed by entities using an authenticated path into the protected network. Devices located outside the protected network may, likewise, offer services that are inappropriate for access from within the protected network. From an ingress perspective, VPN gateways can be configured so that only those network servers intended for external consumption by entities operating on a trusted network (e.g., machines operating on a network where the peer VPN gateways are supporting the connection) are accessible and only via the intended ports. This serves to mitigate the potential for network entities outside a protected network to access network servers or services intended only for consumption or access inside a protected network. From an egress perspective, VPN gateways can be configured so that only specific external services (e.g., based on destination port) can be accessed from within a protected network, or moreover are accessed via an encrypted channel. For example, access to external mail services can be blocked to enforce corporate policies against accessing uncontrolled email servers, or, that access to the mail server must be done over an encrypted link.
T.NETWORK_DISCLOSURE/VPNG	Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to conduct unauthorized activities. If known malicious external devices are able to communicate with devices on the protected network, or if devices on the protected network can establish communications with those external devices (e.g., as a result of a phishing episode or by inadvertent responses to email messages), then those internal devices may be susceptible to the unauthorized disclosure of information. From an infiltration perspective, VPN gateways serve not only to limit access to only specific destination network addresses and ports within a protected network, but whether network traffic will be encrypted or transmitted in plaintext. With these limits, general network port scanning can be prevented from reaching protected networks or machines, and access to information on a protected network can be limited to that obtainable from specifically configured ports on identified network nodes (e.g., web pages from a designated corporate web server). Additionally, access can be limited to only specific source addresses and ports so that specific networks or network nodes can be blocked from accessing a protected

ID	Threat
	network thereby further limiting the potential disclosure of information. From an exfiltration perspective, VPN gateways serve to limit how network nodes operating on a protected network can connect to and communicate with other networks limiting how and where they can disseminate information. Specific external networks can be blocked altogether or egress could be limited to specific addresses or ports. Alternately, egress options available to network nodes on a protected network can be carefully managed in order to, for example, ensure that outgoing connections are encrypted to further mitigate inappropriate disclosure of data through packet sniffing.
T.NETWORK_MISUSE/VPNG	Devices located outside the protected network, while permitted to access particular public services offered inside the protected network, may attempt to conduct inappropriate activities while communicating with those allowed public services. Certain services offered from within a protected network may also represent a risk when accessed from outside the protected network. From an ingress perspective, it is generally assumed that entities operating on external networks are not bound by the use policies for a given protected network. Nonetheless, VPN gateways can log policy violations that might indicate violation of publicized usage statements for publicly available services. From an egress perspective, VPN gateways can be configured to help enforce and monitor protected network use policies. As explained in the other threats, a VPN gateway can serve to limit dissemination of data, access to external servers, and even disruption of services – all of these could be related to the use policies of a protected network and as such are subject in some regards to enforcement. Additionally, VPN gateways can be configured to log network usages that cross between protected and external networks and as a result can serve to identify potential usage policy violations.
T.REPLAY_ATTACK/VPNG	If an unauthorized individual successfully gains access to the system, the adversary may have the opportunity to conduct a “replay” attack. This method of attack allows the individual to capture packets traversing throughout the network and send the packets at a later time, possibly unknown by the intended receiver. Traffic is subject to replay if it meets the following conditions: • Cleartext: an attacker with the ability to view unencrypted traffic can identify an appropriate segment of the communications to replay as well in order to cause the desired outcome • No integrity: alongside cleartext traffic, an attacker can make arbitrary modifications to captured traffic

ID	Threat
	and replay it to cause the desired outcome if the recipient has no means to detect these

3.2 Assumptions

The assumptions included in the following table are drawn directly from the cPP and modules. Assumptions drawn from the modules have "/FW" or "/VPNG" appended to the name to ensure names are not duplicated.

Table 6: Assumptions

ID	Assumption
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or cPP_ND_v3.0e, 06-Dec-2023 41 interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).

ID	Assumption
A.TRUSTED_ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.COMPONENTS_RUNNING	For distributed TOEs it is assumed that the availability of all TOE components is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. It is also assumed that in addition to the availability of all components it is also checked as appropriate that the audit functionality is running properly on all TOE components.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_ADMINISTRATOR	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UPDATES	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATION	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

ID	Assumption
A.CONNECTIONS/VPNG	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE's security policies will be enforced on all applicable network traffic flowing among the attached networks.

3.3 Organizational Security Policies

The OSPs included in the following table are drawn directly from the cPP and modules. OSPs drawn from the modules have "/FW" or "/VPNG" appended to the name to ensure names are not duplicated.

Table 7: OSPs

ID	OSP
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which Administrators consent by accessing the TOE.

4 Security Objectives

The security objectives have been taken directly from the cPP and modules and are reproduced here for the convenience of the reader.

4.1 Security Objectives for the TOE

The TOE security objectives included in the following table are drawn directly from the cPP and modules. Security objectives drawn from the modules have "/FW" or "/VPNG" appended to the name to ensure names are not duplicated.

Table 8: Security Objectives for the TOE

ID	Security Objective
O.RESIDUAL_INFORMATION/FW	The TOE shall implement measures to ensure that any previous information content of network packets sent through the TOE is made unavailable either upon deallocation of the memory area containing the network packet or upon allocation of a memory area for a newly arriving network packet or both.
O.STATEFUL_TRAFFIC_FILTERING/FW	The TOE shall perform stateful traffic filtering on network packets that it processes. For this the TOE shall support the definition of stateful traffic filtering rules that allow to permit or drop network packets. The TOE shall support assignment of the stateful traffic filtering rules to each distinct network interface. The TOE shall support the processing of the applicable stateful traffic filtering rules in an administratively defined order. The TOE shall deny the flow of network packets if no matching stateful traffic filtering rule is identified. Depending on the implementation, the TOE might support the stateful traffic filtering of Dynamic Protocols (optional).
O.ADDRESS_FILTERING/VPNG	To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption or denial of services, and network-based reconnaissance, compliant TOE's will implement packet filtering capability. That capability will restrict the flow of network traffic between protected networks and other attached networks based on network addresses of the network nodes originating (source) or receiving (destination) applicable network traffic as well as on established connection information.
O.AUTHENTICATION/VPNG	To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (IPSec) will allow a VPN peer to establish VPN connectivity with another VPN peer and ensure that any such connection attempt is both authenticated and authorized. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity.

ID	Security Objective
O.CRYPTOGRAPHIC_FUNCTIONS/VPNG	To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption of services, and network-based reconnaissance, compliant TOE's will implement cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE.
O.FAIL_SECURE/VPNG	There may be instances where the TOE's hardware malfunctions or the integrity of the TOE's software is compromised, the latter being due to malicious or non-malicious intent. To address the concern of the TOE operating outside of its hardware or software specification, the TOE will shut down upon discovery of a problem reported via the self-test mechanism and provide signature-based validation of updates to the TSF.
O.PORT_FILTERING/VPNG	To further address the issues associated with unauthorized disclosure of information, etc., a compliant TOE's port filtering capability will restrict the flow of network traffic between protected networks and other attached networks based on the originating (source) or receiving (destination) port (or service) identified in the network traffic as well as on established connection information.
O.SYSTEM_MONITORING/VPNG	To address the issues of administrators being able to monitor the operations of the VPN gateway, it is necessary to provide a capability to monitor system activity. Compliant TOEs will implement the ability to log the flow of network traffic. Specifically, the TOE will provide the means for administrators to configure packet filtering rules to 'log' when network traffic is found to match the configured rule. As a result, matching a rule configured to 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security associations (SAs) is auditable, not only between peer VPN gateways, but also with certification authorities (CAs).
O.TOE_ADMINISTRATION /VPNG	TOEs will provide the functions necessary for an administrator to configure the packet filtering rules, as well as the cryptographic aspects of the IPsec protocol that are enforced by the TOE.

4.2 Security Objectives for the Operational Environment

Security objectives for the operational environment assist the TOE in correctly providing its security functionality. These objectives, which are found in the table below, track with the assumptions about the TOE operational environment. Security objectives drawn from the modules have "/FW" or "/VPNG" appended to the name to ensure names are not duplicated.

Table 9: Security Objectives for the Operational Environment

ID	Security Objective
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services

ID	Security Objective
	necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.COMPONENTS_RUNNING	For distributed TOEs, the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused interVM communications mechanisms), and • correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualisation features such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through

ID	Security Objective
	VM introspection, measured VM boot, or VM snapshot for forensic analysis.
OE.CONNECTIONS /VPNG	The TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

5 Security Requirements

This section identifies the Security Functional Requirements (SFRs) and the Security Assurance Requirements (SARs) for the TOE. The Security Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation*, Version 3.1, Revision 5; the *collaborative Protection Profile for Network Devices,* Version 3.0e, *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e and *PP-Module for Virtual Private Network Gateways*, Version 1.3.

5.1 Functional Requirements

5.1.1 Extended Functional Requirements

All the extended requirements in this ST have been drawn from *collaborative Protection Profile for Network Devices,* Version 3.0e, *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e and *PP-Module for Virtual Private Network Gateways*, Version 1.3 and are itemized below. The cPP and modules define the extended SFRs. Since they have not been redefined in this ST, the cPP and modules should be consulted for more information regarding these extensions to CC Part 2.

From *collaborative Protection Profile for Network Devices,* Version 3.0e:

- FAU_GEN_EXT.1
- FAU_STG_EXT.1
- FAU_STG_EXT.4
- FCO_CPC_EXT.1
- FCS_HTTPS_EXT.1
- FCS_IPSEC_EXT.1 (some SFR elements)
- FCS_RBG_EXT.1
- FCS_TLSC_EXT.1
- FCS_TLSS_EXT.1
- FIA_PMG_EXT.1
- FIA_UIA_EXT.1
- FIA_X509_EXT.1/ITT
- FIA_X509_EXT.1/Rev
- FIA_X509_EXT.2
- FIA_X509_EXT.3
- FPT_APW_EXT.1
- FPT_ITT.1
- FPT_SKP_EXT.1
- FPT_STM_EXT.1
- FPT_TST_EXT.1
- FPT_TUD_EXT.1
- FTA_SSL_EXT.1

From *PP-Module for Stateful Traffic Filter Firewalls*, Version 1.4e:

- FFW_RUL_EXT.1

From *PP-Module for Virtual Private Network Gateways*, Version 1.3:

- FCS_IPSEC_EXT.1 (some SFR elements)

- FIA_PSK_EXT.1
- FIA_PSK_EXT.2
- FPF_RUL_EXT.1
- FPT_TST_EXT.3

5.1.2 Conventions

The CC allows the following types of operations to be performed on the functional requirements: assignments, selections, refinements, and iterations. The following font conventions are used within this document to identify operations defined by CC:

- Assignment: Indicated with *italicized* text;
- Refinement: Indicated with **bold** text;
- Selection: Indicated with underlined text;
- Iteration: The cPP and module authors indicate iterations within the cPP and modules by appending to the SFR a forward slash followed by text. The ST author indicates iterations using the same approach.
- The font formatting used in the cPP, module or TD is not retained, with the exception of cPP or module strikethrough for deleted text. Italicized text and bold text are replaced with plain text.
- If a TD has been issued pertaining to an SFR included in this ST, that TD is referenced in the heading of the SFR.
- Some SFR titles have been modified. FW has been appended to the title of SFRs from MOD_CPP_FW_V1.4E. VPNGW has been appended to the title of SFRs from MOD_VPNGW_V1.3. All SFRs without FW or VPNGW appended are from CPP_ND_V3.0E.

5.1.3 Security Functional Requirements (SFRs)

The TOE functional requirements for this ST are taken directly from the cPPs which are derived from *Common Criteria for Information Technology Security Evaluation Part 2*, Version 3.1, Revision 5.

Table 10: SFRs

SFR Class	Requirement	Description
FAU: Security audit	FAU_GEN.1	Audit Data Generation (FW)
	FAU_GEN.1/VPN	Audit Data Generation (VPNGW)
	FAU_GEN.2	User Identity Association
	FAU_GEN_EXT.1	Security Audit Data Generation for Distributed TOE component
	FAU_STG_EXT.1	Protected Audit Event Storage
	FAU_STG_EXT.4	Protected Local Audit Event Storage for Distributed TOEs
FCO: Communication	FCO_CPC_EXT.1	Component Registration Channel Definition
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.1/IKE	Cryptographic Key Generation (for IKE Peer Authentication) (VPNGW)

SFR Class	Requirement	Description
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption) (VPNGW)
	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_IPSEC_EXT.1	IPsec Protocol (VPNGW)
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_TLSC_EXT.1	TLS Client
	FCS_TLSS_EXT.1	TLS Server
FDP: User data protection	FDP_RIP.2	Full Residual Information Protection (FW)
FFW: Firewall	FFW_RUL_EXT.1	Stateful traffic filtering (FW)
FIA: Identification and Authentication	FIA_AFL.1	Authentication Failure Handling
	FIA_PMG_EXT.1	Password Management
	FIA_PSK_EXT.1	Pre-Shared Key Composition (VPNGW)
	FIA_PSK_EXT.2	Generated Pre-Shared Keys (VPNGW)
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_X509_EXT.1/ITT	X.509 Certificate Validation
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication (VPNGW)
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
	FMT_MOF.1/Services	Management of Security Functions Behaviour
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_MTD.1/CryptoKeys	Management of TSF Data (VPNGW)
	FMT_SMF.1	Specification of Management Functions
	FMT_SMF.1/FFW	Specification of Management Functions (FW)
	FMT_SMF.1/VPN	Specification of Management Functions (VPNGW)
	FMT_SMR.2	Restrictions on Security Roles
FPF: Packet filtering	FPF_RUL_EXT.1	Packet Filtering Rules (VPNGW)
	FPT_APW_EXT.1	Protection of Administrator Passwords

SFR Class	Requirement	Description
FPT: Protection of the TSF	FPT_FLS.1/SelfTest	Failure with Preservation of Secure State (VPNGW)
	FPT_ITT.1	Basic Internal TSF Data Transfer Protection
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing (VPNGW)
	FPT_TST_EXT.3	Self-Test with Defined Methods (VPNGW)
	FPT_TUD_EXT.1	Trusted Update (VPNGW)
FTA: TOE Access	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-Initiated Termination
	FTA_SSL_EXT.1	TSF-Initiated Session Locking
	FTA_TAB.1	Default TOE Access Banner
FTP: Trusted Path/Channels	FTP_ITC.1	Inter-TSF Trusted Channel
	FTP_ITC.1/VPN	Inter-TSF Trusted Channel (VPNGW)
	FTP_TRP.1/Admin	Trusted Path

5.1.3.1 Security Audit (FAU)

5.1.3.1.1 FAU_GEN.1 Audit Data Generation (FW) (TD0923)

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

a. Start-up and shut-down of the audit functions;
b. All auditable events for the not specified level of audit; and
c. All administrative actions comprising:
   - Administrative login and logout (name of Administrator account shall be logged if individual accounts are required for Administrators).
   - Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
   - Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
   - [Resetting passwords (name of related Administrator account shall be logged];
d. Specifically defined auditable events listed in ~~Table 2~~ Table 11.

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

a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of ~~Table 2~~ Table 11.

Table 11: Audit Requirements

Requirement	Requirement	Description
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_GEN_EXT.1	None	None
FAU_STG_EXT.1	Configuration of local audit settings.[1]	Identity of account making changes to the audit configuration.
FAU_STG_EXT.4	None	None
FCO_CPC_EXT.1	• Enabling communications between a pair of components. • Disabling communications between a pair of components.	Identities of the endpoint pairs enabled or disabled.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure
FCS_RBG_EXT.1	None.	None.
FCS_TLSC_EXT.1	• Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1	• Failure to establish a TLS Session	Reason for failure
FDP_RIP.2	None.	None.
FFW_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface

[1] Per TD0923, this audit requirement is trivially satisfied in TOEs such as this that do not have configurable local audit storage.

Requirement	Requirement	Description
FIA_AFL.1	• Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UAU.7	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanisms.	Origin of the attempt (e.g., IP address).
FIA_X509_EXT.1/ITT	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Services	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMF.1/FFW	All management activities of TSF data (including creation, modification and deletion of firewall rules).	None.
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_ITT.1	• Initiation of the trusted channel. • Termination of the trusted channel.	Identification of the initiator and target of failed trusted channels establishment attempt.

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

5.1.3.1.2 FAU_GEN.1/VPN Audit Data Generation (VPN Gateway) (VPNGW)

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

a. Start-up and shutdown of the audit functions
b. Indication that TSF self-test was completed

c. Failure of self-test
d. All auditable events for the [not specified] level of audit; and
e. [auditable events defined in the Auditable Events for Mandatory Requirements table].

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

a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [additional information defined in the Auditable Events for Mandatory Requirements table for each auditable event, where applicable].

Table 12: VPNGW Audit Requirements

Requirement	Requirement	Description
FAU_GEN.1/VPN	No events specified	N/A
FCS_CKM.1/IKE	No events specified	N/A
FIA_PSK_EXT.1	No events specified	N/A
FIA_PSK_EXT.2	No events specified	N/A
FMT_SMF.1/VPN	All administrative actions	No additional information.
FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	• Source and destination addresses • Source and destination ports • Transport layer protocol
FPT_FLS.1/SelfTest	No events specified	N/A
FPT_TST_EXT.3	No events specified	N/A
FTP_ITC.1/VPN	Initiation of the trusted channel	No additional information.

5.1.3.1.3 FAU_GEN_EXT.1 Security Audit Data Generation

FAU_GEN_EXT.1.1 The TSF shall be able to generate audit records for each TOE component. The audit records generated by the TSF of each TOE component shall include the subset of security relevant audit events which can occur on the TOE component.

5.1.3.1.4 FAU_GEN.2 User identity association

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

5.1.3.1.5 FAU_STG_EXT.1 Protected Audit Event Storage

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

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

- The TOE shall be a distributed TOE that stores audit data on the following TOE components: [*GoSilent Server and GoSilent Cube*],

].

FAU_STG_EXT.1.3 The TSF shall maintain a [log file] of audit records in the event that an interruption of communication with the remote audit server occurs.

FAU_STG_EXT.1.4 The TSF shall be able to store [persistent] audit records locally with a minimum storage size of [*50MB for GoSilent Server and 1.5MB for GoSilent Cube*].

FAU_STG_EXT.1.5 The TSF shall [overwrite previous audit records according to the following rule: [*the oldest of the audit files is discarded*]] when the local storage space for audit data is full.

FAU_STG_EXT.1.6 The TSF shall provide the following mechanisms for administrative access to locally stored audit records [manual export, ability to view locally].

5.1.3.1.6 FAU_STG_EXT.4 Protected Local Audit Event Storage for Distributed TOEs

FAU_STG_EXT.4.1 The TSF of each TOE component which stores security audit data locally shall perform the following actions when the local storage space for audit data is full: [*the actions specified in Table 13*].

Table 13: Component Actions

Component	Action Performed
GoSilent Server	[overwrite previous audit records according to the following rule: [*oldest of the audit files is discarded*]]
GoSilent Cube	[overwrite previous audit records according to the following rule: [*oldest of the audit files is discarded*]]

5.1.3.2 Communication (FCO)

5.1.3.2.1 FCO_CPC_EXT.1 Component Registration Channel Definition

FCO_CPC_EXT.1.1 The TSF shall require a Security Administrator to enable communications between any pair of TOE components before such communication can take place.

FCO_CPC_EXT.1.2 The TSF shall implement a registration process in which components establish and use a communications channel that uses [

- A channel that meets the secure channel requirements in [FPT_ITT.1],

] for at least TSF data.

FCO_CPC_EXT.1.3 The TSF shall enable a Security Administrator to disable communications between any pair of TOE components.

5.1.3.3 Cryptographic Support (FCS)

5.1.3.3.1 FCS_CKM.1 Cryptographic Key Generation (TD0921)

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

- ECC schemes using 'NIST curves' [P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4, or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2, or ISO/IEC 14888-3, "IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6;

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

5.1.3.3.2 FCS_CKM.1/IKE Cryptographic Key Generation (for IKE Peer Authentication) (VPNGW) (TD0944)

FCS_CKM.1.1/IKE

The TSF shall generate asymmetric cryptographic keys used for IKE peer authentication in accordance with a specified cryptographic key generation algorithm: [

- FIPS PUB 186-5, "Digital Signature Standard (DSS)," Appendix A.2 for ECDSA schemes, and implementing "NIST curves" P-384 and [P-521]

] and [

- no other key generation algorithm

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

5.1.3.3.3 FCS_CKM.2 Cryptographic Key Establishment

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

- Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";

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

5.1.3.3.4 FCS_CKM.4 Cryptographic Key Destruction

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

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

- logically addresses the storage location of the key and performs a [single] overwrite consisting of [zeroes];

]

that meets the following: No Standard.

5.1.3.3.5 FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/ Decryption) (VPNGW)

FCS_COP.1.1/DataEncryption The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [GCM] and [no other] mode and cryptographic key sizes [256 bits] and [no other cryptographic key sizes] that meet the following: AES as specified in ISO 18033-3, [GCM as specified in ISO 19772], and [no other standards].

5.1.3.3.6 FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification) (TD0921)

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

- Elliptic Curve Digital Signature Algorithm

]

and cryptographic key sizes [

- For ECDSA: [*384, 512 bits*]

]

that meet the following: [

- For ECDSA schemes implementing [P-384, P-521] curves that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST Recommended” curves; or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves; or ISO/IEC 14888-3, "IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.

].

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

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

5.1.3.3.8 FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

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

5.1.3.3.9 FCS_HTTPS_EXT.1 HTTPS Protocol

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

5.1.3.3.10 FCS_IPSEC_EXT.1 IPsec Protocol (VPNGW) (TD0824, TD0868, TD0986)

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

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

FCS_IPSEC_EXT.1.3 The TSF shall implement [tunnel mode].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [AES-GCM-256 (specified in RFC 4106)] and [no other algorithm] together with a Secure Hash Algorithm (SHA)-based HMAC [no HMAC algorithm].

FCS_IPSEC_EXT.1.5

The TSF shall implement the protocol: [

- IKEv2 as defined in RFC 7296 [with mandatory support for NAT traversal as specified in RFC 7296, section 2.23] and [RFC 4868 for hash functions]

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv2] protocol uses the cryptographic algorithms [AES-GCM-256 (specified in RFC 5282)].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

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

  ]

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

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

  ]

].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [384/512] bits.

FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in [IKEv2] exchanges of length [

- according to the security strength associated with the negotiated DH group
- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash

].

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

- 20 (384-bit Random ECP) according to RFC 5114 and

[

- [21 (521-bit Random ECP)] according to RFC 5114

].

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

FCS_IPSEC_EXT.1.13 The TSF shall ensure that [IKEv2] protocols perform peer authentication using [ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys that conform to RFC 8784].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: Distinguished Name (DN), [no other reference identifier types].

5.1.3.3.11 FCS_RBG_EXT.1 Random Bit Generation (TD0990)

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

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[*1*] software-based noise source] with a minimum of [*256 bits*] of entropy at least equal to

[

- the greatest security strength according to ISO/IEC 18031:2011 Table C.2 — "Security strengths, Entropy and Seed length requirements for the AES-128, 192 and 256 Block Cipher",

].

5.1.3.3.12 FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The TSF shall implement [TLS 1.3 (RFC 8446), TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

[

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_AES_256_GCM_SHA384] and no other ciphersuites.

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches [the reference identifier per RFC 6125 Section 6, IPv4 address in the CN or in the SAN, and no other attribute types].

FCS_TLSC_EXT.1.3 The TSF shall not establish a trusted channel if the server certificate is invalid [

- without any administrator override mechanism.

].

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

FCS_TLSC_EXT.1.5 The TSF shall [

- present the signature_algorithms extension with support for the following algorithms: [
  - ecdsa_secp384r1 with sha384(0x0503),
  - ecdsa_secp521r1 with sha512(0x0603),
  -] and no other algorithms;

].

FCS_TLSC_EXT.1.6 The TSF [provides] the ability to configure the list of supported ciphersuites as defined in FCS_TLSC_EXT.1.1.

FCS_TLSC_EXT.1.7 The TSF shall prohibit the use of the following extensions:

- Early data extension
- Post-handshake client authentication according to RFC 8446, Section 4.2.6.

FCS_TLSC_EXT.1.8 The TSF shall [not use PSKs].

FCS_TLSC_EXT.1.9 The TSF shall [reject [TLS 1.2, TLS 1.3] renegotiation attempts].

5.1.3.3.13 FCS_TLSS_EXT.1 TLS Server Protocol

FCS_TLSS_EXT.1.1 The TSF shall implement [TLS 1.3 (RFC 8446), TLS 1.2 (RFC 5246)] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_AES_256_GCM_SHA384] and no other ciphersuites.

FCS_TLSS_EXT.1.2 The TSF shall authenticate itself using X.509 certificate(s) using [ECDSA over NIST curves [secp384r1, secp521r1] and no other curves].

FCS_TLSS_EXT.1.3 The TSF shall perform key exchange using: [

- EC Diffie-Hellman key agreement over NIST curves [secp384r1, secp521r1] and no other curves;

].

FCS_TLSS_EXT.1.4 The TSF shall support [no session resumption].

FCS_TLSS_EXT.1.5 The TSF [does not provide] the ability to configure the list of supported ciphersuites as defined in FCS_TLSS_EXT.1.1.

FCS_TLSS_EXT.1.6 The TSF shall prohibit the use of the following extensions:

- Early data extension

FCS_TLSS_EXT.1.7 The TSF shall [not use PSKs].

FCS_TLSS_EXT.1.8 The TSF shall [reject [TLS 1.2, TLS 1.3] renegotiation attempts].

5.1.3.4 User Data Protection (FDP)

5.1.3.4.1 FDP_RIP.2 Full Residual Information Protection (FW)

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

5.1.3.5 Firewall (FFW)

5.1.3.5.1 FFW_RUL_EXT.1 Stateful Traffic Filtering (FW) (TD0924)

FFW_RUL_EXT.1.1 The TSF shall perform stateful traffic filtering on network packets processed by the TOE.

FFW_RUL_EXT.1.2 The TSF shall allow the definition of stateful traffic filtering rules using the following network protocol fields:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source address
  - Destination Address
  - Transport Layer Protocol
  - [no other field]
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

and distinct interface.

FFW_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with stateful traffic filtering rules: permit or drop with the capability to log the operation.

FFW_RUL_EXT.1.4 The TSF shall allow the stateful traffic filtering rules to be assigned to each distinct network interface.

FFW_RUL_EXT.1.5 The TSF shall:

a) accept a network packet without further processing of stateful traffic filtering rules if it matches an allowed established session for the following protocols: TCP, UDP, [no other protocols] based on the following network packet attributes:
    1. TCP: source and destination addresses, source and destination ports, sequence number, Flags;
    2. UDP: source and destination addresses, source and destination ports;
    3. [no other protocols].
b) Remove existing traffic flows from the set of established traffic flows based on the following: [session inactivity timeout].

FFW_RUL_EXT.1.6 The TSF shall enforce the following default stateful traffic filtering rules on all network traffic:

a) The TSF shall drop and be capable of [counting] packets which are invalid fragments;
b) The TSF shall drop and be capable of [counting] fragmented packets which cannot be re-assembled completely;
c) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a broadcast network;
d) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a multicast network;
e) The TSF shall drop and be capable of logging network packets where the source address of the network packet is defined as being a loopback address;
f) The TSF shall drop 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 drop 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) The TSF shall drop and be capable of logging network packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified; and
i) [no other rules].

FFW_RUL_EXT.1.7 The TSF shall be capable of dropping and logging according to the following rules:

a) The TSF shall drop and be capable of logging network packets where the source address of the network packet is equal to the address of the network interface where the network packet was received;
b) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is a link-local address;
c) The TSF shall drop and be capable of logging network packets where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received.

FFW_RUL_EXT.1.8 The TSF shall process the applicable stateful traffic filtering rules in an administratively defined order.

FFW_RUL_EXT.1.9 The TSF shall deny packet flow if a matching rule is not identified.

FFW_RUL_EXT.1.10 The TSF shall be capable of limiting an administratively defined number of half-open TCP connections. In the event that the configured limit is reached, new connection attempts shall be dropped and the drop event shall be [logged].

5.1.3.6 Identification and Authentication (FIA)

5.1.3.6.1 FIA_AFL.1 Authentication Failure Handling

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [1-10] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until an Administrator defined time period has elapsed].

5.1.3.6.2 FIA_PMG_EXT.1 Password Management

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

a. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers and the following special characters: ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [*no other characters*]];
b. Minimum password length shall be configurable to between [*8*] and [*40*] characters.

5.1.3.6.3 FIA_PSK_EXT.1 Pre-Shared Key Composition (VPNGW)

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for IPsec and [IKEv2].

FIA_PSK_EXT.1.2 The TSF shall be able to accept the following as pre-shared keys: [generated bit-based] keys.

5.1.3.6.4 FIA_PSK_EXT.2 Generated Pre-Shared Keys (VPNGW)

FIA_PSK_EXT.2.1 The TSF shall be able to [

- accept externally generated pre-shared keys

]

5.1.3.6.5 FIA_UAU.7 Protected Authentication Feedback (Refinement)

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

5.1.3.6.6 FIA_UIA_EXT.1 User Identification and Authentication (TD0900)

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

- Display the warning banner in accordance with FTA_TAB.1;
- [[*reset GoSilent Cube to the factory default configuration*]].

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

FIA_UIA_EXT.1.3 The TSF shall provide the following remote authentication mechanisms [Web GUI password] and [no other mechanism]. The TSF shall provide the following local authentication mechanisms [password-based].

FIA_UIA_EXT.1.4 The TSF shall authenticate any administrative user's claimed identity according to each authentication mechanism specified in FIA_UIA_EXT.1.3.

5.1.3.6.7 FIA_X509_EXT.1/ITT X.509 Certificate Validation

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

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

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

5.1.3.6.8 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
    - Server certificates presented for DTLS/TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
    - Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.

- OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

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

5.1.3.6.9 FIA_X509_EXT.2 X.509 Certificate Authentication (VPNGW)

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

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [not accept the certificate].

5.1.3.6.10 FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [Common Name].

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

5.1.3.7 Security Management (FMT)

5.1.3.7.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behaviour

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

5.1.3.7.2 FMT_MOF.1/Services Management of Security Functions Behaviour

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

5.1.3.7.3 FMT_MTD.1/CoreData Management of TSF Data

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

5.1.3.7.4 FMT_MTD.1/CryptoKeys Management of TSF Data (VPNGW)

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to [[manage]] the [cryptographic keys and certificates used for VPN operation] to [Security Administrators].

5.1.3.7.5 FMT_SMF.1 Specification of Management Functions (TD0880)

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

- Ability to administer the TOE remotely;
- Ability to configure the access banner;
- Ability to configure the remote session inactivity time before session termination;
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- [

- Ability to start and stop services;
- Ability to modify the behaviour of the transmission of audit data to an external IT entity;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- Ability to configure the lifetime for IPsec SAs;
- Ability to configure the list of supported (D)TLS ciphers;
- Ability to configure the interaction between TOE components;
- 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 X509.v3 certificates as trust anchors;
- Ability to generate Certificate Signing Request (CSR) and process CA certificate response;
- Ability to administer the TOE locally;
- Ability to configure the local session inactivity time before session termination or locking;
- Ability to configure the authentication failure parameters for FIA_AFL.1;

].

5.1.3.7.6 FMT_SMF.1/FFW Specification of Management Functions (FW)

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

- Ability to configure firewall rules;

5.1.3.7.7 FMT_SMF.1/VPN Specification of Management Functions (VPNGW)

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

- Definition of packet filtering rules
- Association of packet filtering rules to network interfaces
- Ordering of packet filtering rules by priority

[

- No other capabilities

]].

5.1.3.7.8 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- Security Administrator.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.1.3.8 Packet Filtering (FPF)

5.1.3.8.1 FPF_RUL_EXT.1 Packet Filtering Rules (VPNGW)

FPF_RUL_EXT.1.1 The TSF shall perform packet filtering on network packets processed by the TOE.

FPF_RUL_EXT.1.2 The TSF shall allow the definition of packet filtering rules using the following network protocols and protocol fields: [

- IPv4 (RFC 791)
  - source address
  - destination address
  - protocol
- IPv6 (RFC 8200)
  - source address
  - destination address
  - next header (protocol)
- TCP (RFC 793)
  - source port
  - destination port
- UDP (RFC 768)
  - source port
  - destination port

].

FPF_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with packet filtering rules: permit and drop with the capability to log the operation.

FPF_RUL_EXT.1.4 The TSF shall allow the packet filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.5 The TSF shall process the applicable packet filtering rules (as determined in accordance with FPF_RUL_EXT.1.4) in the following order: [Administrator-defined].

FPF_RUL_EXT.1.6 The TSF shall drop traffic if a matching rule is not identified.

5.1.3.9 Protection of the TSF (FPT)

5.1.3.9.1 FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

5.1.3.9.2 FPT_FLS.1/SelfTest Failure with Preservation of Secure State (Self-Test Failures) (VPNGW)

FPT_FLS.1.1/SelfTest The TSF shall shut down when the following types of failures occur: [failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests].

5.1.3.9.3 FPT_ITT.1 Basic internal TSF data transfer protection

FPT_ITT.1.1 The TSF shall protect TSF data from disclosure and detect its modification when it is transmitted between separate parts of the TOE through the use of [IPsec].

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

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

5.1.3.9.5 FPT_STM_EXT.1 Reliable Time Stamps

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

FPT_STM_EXT.1.2 The TSF shall [allow the Security Administrator to set the time, obtain time from the underlying virtualization system].

5.1.3.9.6 FPT_TST_EXT.1 TSF Testing (VPNGW) (TD0836)

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests

- During initial start-up (on power on) to verify the integrity of the TOE firmware and software;
- Prior to providing any cryptographic service and [continuously] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;
- [no other] self-tests [*none*].

to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2 The TSF shall respond to [all failures] by [[*GoSilent Server displays an error message on the console (accessible via ESXi/KVM) and halts normal processing until rebooted; GoSilent Cube indicates an error condition via its 2 LEDs and halts processing until rebooted*]].

5.1.3.9.7 FPT_TST_EXT.3 Self-Test with Defined Methods (VPNGW)

FPT_TST_EXT.3.1 The TSF shall run a suite of the following self-tests [[when loaded for execution]] to demonstrate the correct operation of the TSF: [integrity verification of stored executable code].

FPT_TST_EXT.3.2 The TSF shall execute the self-testing through [a TSF-provided cryptographic service specified in FCS_COP.1/SigGen].

5.1.3.9.8 FPT_TUD_EXT.1 Trusted Update (VPNGW)

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

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

FPT_TUD_EXT.1.3 The TSF shall provide means to authenticate firmware/software updates to the TOE using a digital signature mechanism and [no other mechanisms] prior to installing those updates.

5.1.3.10 TOE Access (FTA)

5.1.3.10.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

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

- terminate the session] after a Security Administrator-specified time period of inactivity.

5.1.3.10.2 FTA_SSL.3 TSF-initiated Termination

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

5.1.3.10.3 FTA_SSL.4 User-initiated Termination

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

5.1.3.10.4 FTA_TAB.1 Default TOE Access Banners

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

5.1.3.11 Trusted Path/Channels (FTP)

5.1.3.11.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 The TSF shall be capable of using [TLS] to provide a trusted communication channel between itself and ~~another trusted IT product~~ authorized IT entities supporting the following capabilities: audit server, [no other capabilities] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modification or~~ disclosure and detection of modification of the channel data.

FTP_ITC.1.2 The TSF shall permit [the TSF] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*transmission of audit records to the audit server*].

5.1.3.11.2 FTP_ITC.1/VPN Inter-TSF Trusted Channel (VPN Communications) (VPNGW)

FTP_ITC.1.1/VPN The TSF shall be capable of using IPsec to provide a communication channel between itself and authorized IT entities supporting VPN communications that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2/VPN The TSF shall permit [the authorized IT entities] to initiate communication via the trusted channel.

FTP_ITC.1.3/VPN The TSF shall initiate communication via the trusted channel for [remote VPN gateways or peers].

5.1.3.11.3 FTP_TRP.1/Admin Trusted Path

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

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

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

5.1.4 TOE SFR Dependencies Rationale for SFRs

The cPPs contain all the requirements claimed in this ST. As such, dependencies are not applicable since the cPPs have been approved.

5.2 Assurance Requirements

5.2.1 Extended Assurance Requirements

There are no extended assurance requirements defined in the cPP and modules and therefore, none are included in this ST.

5.2.2 Security Assurance Requirements (SARs)

The TOE assurance requirements for this ST are taken directly from the cPPs which are derived from *Common Criteria for Information Technology Security Evaluation Part 3*, Version 3.1, Revision 5.

Table 14: Security Assurance Requirements

Functional Class	Functional Components	Component Description
Security Target	ASE_CCL.1	Conformance Claims
	ASE_ECD.1	Extended Components Definition
	ASE_INT.1	ST Introduction
	ASE_OBJ.1	Security Objectives for the Operational Environment
	ASE_REQ.1	Stated Security Requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE Summary Specification
Development	ADV_FSP.1	Basic Functionality Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative Procedures
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_IND.1	Independent Testing – Sample
Vulnerability Assessment	AVA_VAN.1	Vulnerability Survey

5.2.3 Assurance Measures

The TOE satisfied the identified assurance requirements. This section identifies the Assurance Measures applied by CACI to satisfy the assurance requirements. The following table lists the details.

Table 15: TOE Security Assurance Measures

SAR Component	How the SAR will be met
ASE_TSS.1	Refinement: The TOE summary describes how the TOE meets each SFR. In the case of entropy analysis, the TSS is used in conjunction with required supplementary information on Entropy.
ADV_FSP.1	The functional specification describes the TOE Security Functions Interfaces (TSFIs). It is not necessary to have a formal or complete specification of these interfaces. Additionally, because TOEs conforming to this cPP will necessarily have interfaces to the Operational Environment that are not directly invokable by TOE administrators, there is little point specifying that such interfaces be described in and of themselves since only indirect testing of such interfaces may be possible. For this cPP, the Evaluation Activities for this family focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation. No dedicated "functional specification" documentation is necessary to satisfy the Evaluation Activities specified in [SD]. The Security Target, AGD documentation, supplementary information, or combination of thereof constitutes "functional specification" documentation. This documentation must contain the description of all security-relevant interfaces. The Evaluation Activities in [SD] are associated with the applicable SFRs; since these are directly associated with the SFRs, the tracing in element ADV_FSP.1.2D is implicitly already done and no additional documentation is necessary.
AGD_OPE.1	The operational user guidance does not have to be contained in a single document. Guidance to users, Administrators and application developers can be spread among documents or web pages. The developer should review the Evaluation Activities contained in [SD] to ascertain the specifics of the guidance that the evaluator will be checking for. This will provide the necessary information for the preparation of acceptable guidance.
AGD_PRE.1	As with the operational guidance, the developer should look to the Evaluation Activities to determine the required content with respect to preparative procedures.
ALC_CMC.1	This component is targeted at identifying the TOE such that it can be distinguished from other products or versions from the same developer and can be easily specified when being procured by an end user. A label could consist of a 'hard label' (e.g., stamped into the metal, paper label) or a 'soft label' (e.g., electronically presented when queried). The evaluator performs the CEM work units associated with ALC_CMC.1.
ALC_CMS.1	Given the scope of the TOE and its associated evaluation evidence requirements, the evaluator performs the CEM work units associated with ALC_CMS.1.
ATE_IND.1	Testing is performed to confirm the functionality described in the TSS as well as the guidance documentation (includes "evaluated configuration" instructions). The focus of the testing is to confirm that the requirements specified in Section 5.1.7 are being met.

SAR Component	How the SAR will be met
	The Evaluation Activities in [SD] identify the specific testing activities necessary to verify compliance with the SFRs. The evaluator produces a test report documenting the plan for and results of testing, as well as coverage arguments focused on the platform/TOE combinations that are claiming conformance to this cPP. CACI will provide the TOE for testing.
AVA_VAN.1	CACI will provide the TOE for testing. CACI will provide a document identifying the list of software and hardware components.

5.2.4 Rationale for Security Assurance Requirements

The functional specification describes the external interfaces of the TOE, such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.

6 TOE Summary Specification

6.1 Security Audit (FAU)

6.1.1 FAU_GEN.1, FAU_GEN.1/VPN, FAU_GEN.2, FAU_GEN_EXT.1

The TOE generates the audit records specified at FAU_GEN.1 containing fields that include the timestamp, IP address (if applicable), action, user (if applicable) and a contextual message indicating success or failure of the action.

GoSilent Server and GoSilent Cube generate audit records specified in FAU_GEN.1, FAU_GEN.1/VPN, FAU_GEN.2, and FAU_GEN_EXT.1 for operation and administration of each component. The specific events audited on each component correspond to the SFRs applicable to each component and can be found in Table 11 and Table 12. Auditing is always enabled, and each audit record includes:

- Date and time of the event,
- Type (i.e., category and action) of event,
- Subject (i.e., user and domain) identity,
- Result (success or failure) of the event, and
- Description (where applicable access mode, target object, etc.).

The TOE includes the user identity in audit events resulting from actions of identified users.

For audit records involving the generating/import of, changing, or deleting of cryptographic keys, the record identifies the key via reference to the certificate or key identifier associated with the key. For Post-quantum Pre-shared Keys (PPKs), the associated PPK-ID is included in the audit record.

The GoSilent Server and GoSilent Cube components generate audit records associated with the SFRs they implement as specified in Table 11 and Table 12. Note that the audit requirement for FAU_STG_EXT.1 is 'trivially satisfied' per TD0923 since the local audit storage is not configurable in this TOE.

6.1.2 FAU_STG_EXT.1 & FAU_STG_EXT.4

The TOE is a distributed TOE that persistently stores audit data locally on each TOE component. GoSilent Server can store up to 50MB of audit information and the Cube can store up to 1.5MB. When that space is exhausted, the oldest records are discarded so that new records can be saved. The records are stored in a series of 5 files, 10MB each on GoSilent Server and 300KB each on the Cube. The current file is always "messages". When it fills, it is initially renamed messages1 and a new (empty) messages file is created. As additional files fill, the number appended to each of the existing files is incremented to make room for a new messages1 file. The name of the messages4 file is not incremented; instead, that file is deleted to limit the amount of saved data.

Each component also transmits a copy of each audit record to an external syslog server in real time; the syslog server is configured on GoSilent Server and communicated to all GoSilent Cubes. Each Cube transmits the syslog server traffic via TLS through the established IPsec tunnel. Once the IPsec header has been stripped by GoSilent Server, the syslog (TLS) traffic is then forwarded to the syslog server based on the IP destination address. If the connection to the syslog server is unavailable, audit records continue to be saved locally. Once the connection is re-established, audit logs saved locally but not previously transmitted to the syslog server and sent.

GoSilent Server provides a mechanism for authorized administrators to view local audit records via the management GUI. GoSilent provides the capability to export audit record files for external review.

Only authorized administrators may view audit records and no capability to modify or delete the audit records is provided.

6.2 Communication (FCO)

6.2.1 FCO_CPC_EXT.1

GoSilent Cube registration (enablement) involves some manual configuration by an authorized administrator on GoSilent Server prior to the enablement process taking place. The X.509 certificate for GoSilent Server and the specific GoSilent Cube are communicated out of band and imported by an authorized administrator on that component. Guidance information instructs administrators to use a secure communication channel for communication of the certificates.

When the necessary steps have been taken by an administrator to prepare the Cube for enablement, the channel described in FCO_CPC_EXT.1 and FTP_ITC.1/VPN is used to execute and finalize the registration process which occurs during the first connection of a Cube to GoSilent Server.

Any attempt by a GoSilent Cube to establish an IPsec tunnel (connect) to GoSilent Server is rejected by the GoSilent Server if enablement has not been completed. Each GoSilent Cube only establishes an IPsec tunnel with GoSilent Server. Attempts to establish an IPsec tunnel from one GoSilent Cube to another are always rejected. Alternatively, a cube may also be suspended from within the GoSilent Server GUI, whereby subsequent connections from the cube are rejected.

6.3 Cryptographic Support (FCS)

6.3.1 FCS_CKM.1, FCS_CKM.1/IKE, & FCS_CKM.2

As described in FCS_CKM.1, the TOE generates P-384 and P-521 ECC keys in support of TLS client connections with the syslog server, and P-384 and P-521 ECC keys in support of Administrative GUI TLS server sessions. When performing Elliptic Curve Diffie-Hellman, the TOE acts as a sender with the TLS client connections and a recipient with the TLS server connections.

These keys are also used to authenticate the TOE component to the administrator in TLS exchanges for the management GUI.

As described in FCS_CKM.1/IKE, the TOE generates ECDSA P-384 and P-521 Elliptic Curve keys as specified in FIPS Pub 186-5 “Digital Signature Standard (DSS)” Appendix B.4 and implements all “shall” and “should” statements and does not implement any “shall not” or “should not” statements.

These keys are used to perform Diffie-Hellman in IPsec and adhere to a size equivalent to or greater than a strength of 192 bits.

The TOE acts as a sender and a recipient when performing Diffie-Hellman.

The TOE supports key establishment as follows:

TLS Client to syslog server – Elliptic-curve (P-384 and P-521)

TLS Server for administrative GUI - Elliptic-curve (P-384 and P-521)

IKE/IPsec - Elliptic-curve (P-384 and P-521)

ECDSA key pairs for X509 certificates can also be generated through the TOE GUI where the option of keys using P-384 and P-521 curves can be selected.

6.3.2 FCS_CKM.4

The TOE leverages the underlying filesystem to facilitate the destruction of cryptographic keys. Cryptographic keys in volatile and non-volatile memory are overwritten with zeroes when a key is deleted. Keys in volatile memory are also destroyed when a component is powered down or rebooted. The following keys and other sensitive information are maintained on each component:

ECDSA private key – stored on the hard drive and overwritten when a factory reset is performed

IPsec session key – stored in volatile memory and overwritten when a session is terminated

TLS session key - stored in volatile memory and overwritten when a session is terminated

Administrator passwords – Plaintext value is stored in volatile memory when supplied by a user and overwritten after validation; configured administrator passwords are stored on the hard drive as hashed (SHA-256) values only.

No configurations or circumstances exist that do not conform to the key destruction requirement.

6.3.3 FCS_COP.1/DataEncryption

The TOE performs data encryption and decryption functions using AES 256 bit encryption in GCM mode as follows:

- TLS Client to syslog server – AES-GCM, 256 bit keys
- TLS Server for administrative GUI - AES-GCM, 256 bit keys
- IKE/IPsec –AES-GCM, 256 bit keys

6.3.4 FCS_COP.1/SigGen

The TOE provides cryptographic signature generation and verification services using:

ECDSA P-384 and P-521 SigGen to support IPsec and TLS functions.

ECDSA P-384 and P-521 SigVer to support IPsec, TLS, X.509, trusted update functions including firmware integrity checking.

The relevant NIST CAVP certificate numbers are listed in Table 16.

6.3.5 FCS_COP.1/Hash

The TOE provides cryptographic hashing services using SHA-256, SHA-384 and SHA-512.

SHA-384 and SHA-512 are used for SigGen and SigVer operations; SHA-256 is only used for hashing configured passwords. The hash algorithms are also used in the associated HMAC algorithms.

SHA-384 and SHA-512 are used with TLS client connections to the syslog server.

SHA-384 and SHA-512 are used with IKE/IPsec connections.

SHA-384 and SHA-512 is used for TLS server connections for the administrative web GUI.

SHA-256 is used as a conditioning function in the CTR DRBG.

The relevant NIST CAVP certificate numbers are listed in Table 16.

6.3.6 FCS_COP.1/KeyedHash

The TOE provides keyed-hashing message authentication services using HMAC-SHA-384 and HMAC-SHA-512.

TLS client connections to the syslog server use key sizes [384] bits, [1024] bit block size, and [384] bit message digest size. The TOE uses HMAC-SHA-384 and HMAC-SHA-512 TLS Key Derivation Function (KDF) and TLS message authentication, and the administrative GUI uses HMAC-SHA-384 and HMAC-SHA-512 with key sizes [384, 512] bits, [1024] bit block size, and [384, 512] bit message digest size.

The relevant NIST CAVP certificate numbers are listed in Table 16.

6.3.7 FCS_IPSEC_EXT.1

The TOE implements the IPsec architecture per Request for Comment (RFC) 4301 using tunnel mode. Only IKEv2 is supported. AES in GCM mode using 256-bit keys is supported for both IKEv2 and Encapsulating Security Payload (ESP). IPsec support includes:

Network Address Translation (NAT) traversal as specified in RFC 7296 section 2.23

Hash functions as specified in RFC 4868

Security Association (SA) lifetimes based on time, configurable between 1 and 24 hours

Child SA lifetimes based on time, configurable between 1 and 8 hours

Diffie–Hellman (DH) groups 20 and 21 (security strengths 192 and 256 bits)

Size of the random secret value “x” and nonce used for key establishment, as generated by the DRBG, is at least twice the security strength of that associated with the negotiated DH group. The nonces used in IKE exchanges are generated in a manner such that the probability that a specific nonce value will be repeated during the life of a specific IPsec SA is less than 1 in 2^[192]. Nonce lengths are dynamically determined based on the negotiated Diffie-Hellman group or the pseudorandom function (PRF) hash as follows:

- Nonce lengths must be sufficiently large to support all TOE-chosen proposals during the exchange, as nonces may be exchanged before the Diffie-Hellman group is negotiated.
- For Diffie-Hellman Groups 20 and 21, the nonce length is at least 192 bits.
- When the negotiated PRF hash is used to determine nonce length, the nonce is set to at least half the output size of the PRF hash:
    - For HMAC-SHA-384, the nonce length is at least 192 bits.
    - For HMAC-SHA-512, the nonce length is at least 256 bits.

Child SA key sizes must be the same or less than the IKEv2 SA key size (256 bits). During the IKEv2 CHILD_SA process, the TOE performs checks to ensure that encryption strengths for the selected algorithm are less than or equal to that of the IKEv2 SA.

Peer authentication using ECDSA certificates with key sizes 384 or 512 bits.

Distinguished Names as reference identifiers.

The GoSilent Server has a nominal Security Policy Database (SPD) that denies all traffic. When an IPsec tunnel to GoSilent Server is established, an SPD entry is dynamically added that permits all traffic received from user systems with an IP source address that is consistent with the IP subnet address

configured for the GoSilent Cube IPsec interface. No traffic from a user system is forwarded to the TOE Network other than through an IPsec tunnel (to GoSilent Server).

The GoSilent Server has a nominal SPD that denies all traffic to any GoSilent Cube. When an IPsec tunnel from each GoSilent Cube is established, an SPD entry is dynamically added that permits all traffic destined for an IP address that is consistent with the IP subnet address configured for the GoSilent Cube IPsec interface.

The TOE supports RFC 8784 PPKs with IPsec. Administrators may import (load) externally-generated bit-based PPKs and their associated PPK-IDs onto GoSilent Server and GoSilent Cube, and then configure a PPK-ID to be used with a specific remote endpoint. If an endpoint is configured for a connection, then PPKs must be used for that connection and the same PPK-ID must be configured for both GoSilent Server and GoSilent Cube. If no PPK-ID is configured, standard IKEv2 protocols are used for the connection.

6.3.8 FCS_RBG_EXT.1

The TOE implements a Counter DRBG (AES), compliant with ISO/IEC 18031:2011 (SP800-90B), to generate random bits needed for asymmetric key, symmetric key, nonce, and salt generation. The DRBG is seeded with 256 bits of entropy from one software-based noise source.

Additional detail is provided in the proprietary Entropy Description.

6.3.9 FCS_TLSC_EXT.1

Both the GoSilent Server and GoSilent Cube act as a TLS client to provide a trusted channel to the syslog server. The TOE initiates this connection. This client (on both TOE components) supports TLSv1.2 and TLSv1.3 with the following configurable ciphersuites:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_AES_256_GCM_SHA384

Any SSL/TLS version stated in the Server Hello other than TLS v1.2 or TLSv1.3 causes the connection to be terminated.

The connection is always terminated if the server certificate is invalid.

The TOE always presents the Supported Elliptic Curves Extension indicating support for P-384 and P-521 in the Client Hello. The TOE always presents the Signature Algorithms Extension indicating support for P-384 and P-521. Neither Pre-Shared Keys (PSKs) nor secure renegotiation are supported.

Administrators may select which of the 2 supported ciphersuites are used.

The TSF does not support the use of the following extensions:

- Early data extension
- Post-handshake client authentication according to RFC 8446, Section 4.2.6.

The TOE automatically parses the reference identifier from the connection parameters, using the hostname or IP address as the reference identifier. When validating the server certificate, the TSF matches the reference identifier against the SAN:DNS or SAN:IP address field in the presented certificate (if present) per RFC 6125 section 6 and falls back to the CN if the SAN is not present. The TOE treats the CN as a string value and compares the configured syslog server string to that value. The TOE does not support wildcards. The CN string must conform to dotted decimal for IPv4 (RFC 3986) or eight colon

separated groups of four lowercase hexadecimal digits for IPv6 (RFC 5952). The canonical text representation recommendation for IPv6 in chapter 4 of RFC 5952 is not enforced.

6.3.10 FCS_TLSS_EXT.1, FCS_HTTPS_EXT.1

The TOE acts as a TLS/HTTPS server in accordance with RFC 2818 to provide a web GUI to administrators. All MUST and REQUIRED requirements specified in RFC 2818 are followed. This server supports TLSv1.2 and TLSv1.3 with the following ciphersuites:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_AES_256_GCM_SHA384

These ciphersuites are preconfigured in the TOE and can't be configured by administrators.

Any SSL/TLS version stated in the Client Hello other than TLS v1.2 or TLS v1.3 causes the connection to be terminated.

Key Establishment is performed using Elliptic Curve Diffie-Hellman P-384 and P-521 keys.

This connection is always initiated by the remote end. The TOE authenticates itself with X.509 certificates using ECDSA over NIST curves secp384r1 or secp521r1.

The following functionality is not supported:

- Session resumption
- Session tickets
- Early data extension
- PSKs
- Secure renegotiation

6.4 User Data Protection (FDP)

6.4.1 FDP_RIP.2

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 erased when the resource is released. Therefore, no residual information from packets in a previous information stream can traverse through the TOE.

6.5 Firewall (FFW) & Packet Filtering (FPF)

6.5.1 FFW_RUL_EXT.1 & FPF_RUL_EXT.1

The TOE performs stateful packet filtering on all packets received from or being sent to the External Network, MGMT interface or WAN interface (excepting IPsec traffic) of the GoSilent Server.

The boot sequence of the TOE aids in establishing the secure domain and preventing tampering 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 OS

- Self-tests including firmware integrity tests are executed
- The init utility is started (mounts file systems, sets up network cards, and generally starts all the processes that usually are run on the system at startup)
- Daemon programs such as Internet Service Daemon (INETD), Syslogd are started; Routing and forwarding tables are initialized
- Application daemons are loaded, enabling access to the GoSilent Server management GUI
- 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 the kernel and daemons, and these interfaces cannot send or receive packets unless previously configured by an administrator. Since the daemon for the management GUI is not loaded until after the kernel and INETD are initialized, no modification to the security attributes can be made by a user or process other than via the management process.

The TOE applies a uniform policy to the traffic flows to and from all GoSilent Cube users. By default, no traffic is allowed to flow from GoSilent Cube users to the External Network, no traffic is allowed to flow from the External Network to GoSilent Cube users, and no traffic is allowed to flow between GoSilent Server interfaces.

By default, the TOE allows administrators to define traffic filtering rules based on a distinct interface and the following network protocol fields:

- ICMPv4 (Type, Code)
- ICMPv6 (Type, Code)
- IPv4 (Source address, Destination address, Transport Layer Protocol)
- IPv6 (Source address, Destination address, Transport Layer Protocol)
- TCP (Source port, Destination port)
- UDP (Source port, Destination port)

The TOE does not implement a mechanism to ensure conflicting rules can't be configured.

The TOE maintains a session table which tracks all known TCP and UDP sessions based on the information in incoming packets. Specifically, the lookup is based on an exact match of the following network packet attributes:

- TCP
  - Source and Destination IP address
  - Source and Destination Port
  - Protocol Flags (e.g. SYN, ACK, RST, and FIN flags),
  - Sequence numbers
- UDP
  - Source and Destination IP address
  - Source and Destination Port

The TSF allows the definition of packet filtering rules by using the following network protocols and protocol fields:

- IPv4 (RFC 791)
  - Source address

  - Destination address
  - Protocol
- IPv6 (RFC 8200)
  - Source address
  - Destination address
  - Next header (protocol)
- TCP (RFC 793)
  - Source port
  - Destination port
- UDP (RFC 768)
  - Source port
  - Destination port

Conformance with the above listed RFC's has been determined through testing and development and as a requirement of this evaluation.

For TCP sessions, the TOE removes existing traffic flows from the list of established sessions based on the completion of the session. An exchange of SYN flags indicates the start of a new session. An exchange of FIN flags indicates the end of data transmission to finish a TCP connection, at which point the session is terminated and the session is removed from the table. TCP session are also removed after a configurable session inactivity timeout threshold. By default, this threshold is 432000 seconds. Once this timeout period has been reached, the TOE will then terminate the connection and remove it from the table. New packets that match the session will start a new session.

UDP sessions are removed after the session inactivity timeout threshold. Once this threshold has been reached, the TOE terminates the connection and removes it from the table. New packets that match the session will start a new session.

The default session timeout thresholds for TCP and UDP connections are indicated below.

Traffic for known sessions is permitted to flow. For traffic not associated with known sessions, rules within information flow policies are processed in an administrator-defined order to determine if the traffic should be forwarded. By default, the TOE behavior is to deny packets when there is no rule match. The TOE performs stateful network traffic filtering on network packets using the network traffic protocols and network fields specified in FFW_RUL_EXT.1.5 and FPF_RUL_EXT.1.3. The TOE allows permit, deny, and log operations to be associated with rules and these rules can be assigned to distinct network interfaces. The log operation may be combined with either the permit or deny operations, or may be used by itself.

The TOE tracks the number of half-open TCP connections. When the configured limit for half-open TCP connections is exceeded, TCP SYN are discarded until the number drops below the configured limit. An audit record is generated if the firewall rule is configured to log these events. Half-open TCP connections are automatically discarded after a 60 second timeout period. The following default timeout parameters (in seconds) are configurable in GoSilent Server within the Networking tab:

- Generic IP Timeout: 600
- TCP Connection Timeout: 432000
- UDP Protocol Timeout: 30
- UDP Stream Timeout: 180

In situations where the TOE receives packets faster than they can be processed (flooding), the TOE silently discards the excess packets. An audit record is generated when flooding is detected, but not for each packet.

The TOE can enforce the following reject rules with logging on traffic:

- invalid fragments (counted by TSF);
- fragmented IP packets which cannot be re-assembled completely (counted by TSF);
- 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;
- where the source address of the network packet is equal to the address of the network interface where the network packet was received;
- where the source or destination address of the network packet is a link-local address;
- where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received;
- when a new TCP connection attempt would exceed the configured limit of half-open TCP connections;
- 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 TOE logs the actions taken on packets when they are processed by the INPUT, OUTPUT, and FORWARD rulesets.

In the event of a component failure, such as the firewall failing to initialize, the TOE will not enter a state where network traffic is able to flow. If a component fails during operation, the TOE will enter into a non-operational state and reboot.

6.6 Identification and Authentication (FIA)

6.6.1 FIA_AFL.1, FIA_PMG_EXT.1, FIA_UIA_EXT.1 & FIA_UAU.7

Management of the TOE is primarily performed locally or remotely through GoSilent Server. However, management of GoSilent Cube directly provides the initial configuration information for it to connect to GoSilent Server. Administrator access to GoSilent Cube is performed locally. Unless otherwise stated, the information in the remainder of this section applies to all management access to GoSilent Server and GoSilent Cube.

Identification and authentication are required for both local and remote administrator access. Management access to the TOE is via an HTTPS session.

The TOE supports local authentication where it looks up the username in its local configuration and compares the hash of the password to the saved value. If the credentials are valid, the user is successfully authenticated and is authorized to access the management interface.

Authentication of an administrator is through use of a username/password. The minimum password may be configured from 8 to 40 characters, that incorporate a combination of lowercase letters, uppercase letters, numbers and special characters (“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”). During entry of the password, each character entered is masked with a “*” when progress is reflected on the screen. If an authentication attempt fails, (either the username is not recognized or the password is incorrect) the same “Login failed” error message is presented.

The TOE tracks the number of sequential failed authentication attempts for each user account. Upon meeting the configured limit for failed authentication attempts, the TOE locks the account in question for an administrator configured period of time. During this time, entering the correct password for the locked account will still result in an authentication failure. Any successful authentication resets the counter to zero.

For both local and remote connections on GoSilent Server, the TOE only provides the warning banner as described in FTA_TAB.1 prior to authentication. The GoSilent Cube also provides a function to reset the Cube to factory default configuration via the GUI.

Local access to the GoSilent Server is via an HTTPS session through the MGMT interface. The source IP address must match the configured whitelist address. The workstation used to access the MGMT interface should be collocated with the GoSilent Server. Administrator account lockout is not applicable to local access. Remote administrator access is via an HTTPS session from GoSilent Cube client systems.

Local access for GoSilent Cube is via the Ethernet interface labelled ‘Device’. Administrator account lockout is not applicable to local access. Remote configuration of some features on the GoSilent Cube can be achieved via GoSilent Server. Those configuration settings are downloaded by the GoSilent Cube when it establishes an IPsec tunnel to the GoSilent Server.

6.6.2 FIA_X509_EXT.1/ITT, FIA_X509_EXT.1/Rev, FIA_X509_EXT.2, FIA_X509_EXT.3

The TOE uses X.509 certificates to:

- provide IPsec connections between GoSilent Server and Cube devices
- verify the identity of the Syslog server

When a TOE component receives a certificate asserting the identity of a remote system during an IPsec connection, the TOE ensures the current time is within the validity time of the certificate, the certificate

has not been revoked (verified via OCSP for IPsec or CRLs for TLS), contains the appropriate extendedKeyUsage (EKU) purpose set (Server Authentication), and the certificate chain terminates with a trusted CA certificate. The certificate chain is validated by verifying each certificate (except for the end entity certificate) in the chain is currently valid, has not been revoked, contains the basic constraints extension with the CA flag set to TRUE, and is signed by a trusted CA or is explicitly configured as a trusted CA. The entire chain is validated regardless of whether the only a leaf certificate is presented. If the TOE cannot establish a connection to the OCSP server or CDP, the TOE will reject the certificate. The certificate of the OCSP responder must include OCSP Signing in its EKU.

Revocation checking for the entire certificate chain (to the Root) is performed when certificates are loaded into the TOE, when certificates are received from the remote side during TLS and IPsec connection establishment, and when OCSP responses are received.

The EKU checks for Code Signing and Client Authentication are not applicable to this TOE. Trusted updates are not verified using X.509 certificates, and TLS mutual authentication is not supported.

On GoSilent Server, distinct leaf certificates are supported for IPsec (VPN connections) and the administrator GUI. The same certificate can be used for both functions, or separate certificates can be used and are therefore determined during import when selecting the usage for the certificate. On GoSilent Cube, a distinct leaf certificate is supported for IPsec connections with each configured Server profile. The certificates used is determined by the certificate configured in the profile selected by the user. On the Cube, the client certificate to be used is chosen during the configuration of the server profile (Virtual Server setup). When receiving a certificate, GoSilent Server will verify it against its own trust store, and check the revocation status via OCSP (for IPsec) and CRL (for TLS).

For a Certificate Signing Request, a Common Name may be specified to include in the certificate.

6.6.3 FIA_PSK_EXT.1, FIA_PSK_EXT.2

The TOE supports RFC 8784 PPKs with IPsec. Administrators may import (load) externally-generated bit-based PPKs and their associated PPK-IDs onto GoSilent Server and GoSilent Cube, and then configure a PPK-ID to be used with a specific remote endpoint. If an endpoint is configured for a connection, then PPKs must be used for that connection and the same PPK-ID must be configured for both GoSilent Server and GoSilent Cube. If no PPK-ID is configured, standard IKEv2 protocols are used for the connection.

6.7 Security Management (FMT)

6.7.1 FMT_MOF.1/ManualUpdate, FMT_MOF.1/Services, FMT_MTD.1/CoreData, FMT_MTD.1/CryptoKeys, FMT_SMF.1, FMT_SMF.1/FFW, FMT_SMF.1/VPN, FMT_SMR.2

The information in this section applies to both GoSilent Server and GoSilent Cube unless otherwise stated. On GoSilent Server, the management functionality provided is the same for local and remote administrators.

Management of the TOE is primarily performed through GoSilent Server. However, initial management of GoSilent Cube, providing enough configuration information for it to connect to GoSilent Server, is required. The TOE supports the single administrator role of Security Administrator; no other users have elevated privileges. The TOE can be administered both locally and remotely (on GoSilent Server), and all administrators are authorized for both local and remote access.

The GoSilent Server provides the following management capabilities to authorized administrators both locally and remotely:

- Perform manual updates of GoSilent Server (which includes validation using digital signatures);
- Enable and disable audit logging (although note that the evaluated configuration requires auditing to be enabled at all times);
- Manage (import) cryptographic keys and certificates, including PSKs for IPsec;
- Ability to configure the access banner;
- Ability to configure the session inactivity time for local and remote sessions before session termination;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- Ability to configure firewall rules;
- Ability to configure the cryptographic functionality;
- Ability to configure the lifetime for IPsec SAs;
- Ability to configure the transmission of audit data, including the ciphersuite used when connecting to the audit server;
- 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 X509.v3 certificates as trust anchors;
- Ability to configure packet filtering rules.

GoSilent Cube provides the following management capabilities to authorized administrators:

- Perform manual updates of the GoSilent Cube the administrator is connected to (which includes validation using digital signatures);
- Manage (import) cryptographic keys and certificates, including PSKs for IPsec;
- Ability to configure the cryptographic functionality;
- 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 X509.v3 certificates as trust anchors;
- Ability to configure the packet filtering rules.

The Cube provides the ability to pull additional configuration information from the GoSilent Server it is registered with upon connection establishment and includes the following:

- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- Ability to configure the transmission of audit data;

Maintenance of the local audit log files is automated. Maintenance of the files is described in ST section 6.1.2. Protection of the trust store is described in ST section 6.8.1.

6.8 Protection of the TSF (FPT)

6.8.1 FPT_APW_EXT.1 & FPT_SKP_EXT.1

The TOE does not provide access to the filesystem or any administrative interface, including those designed specifically for that purpose, that would allow a user or administrator to view plaintext

administrative passwords, pre-shared keys, symmetric keys, or private keys and additionally prevents the reading of any plaintext administrative passwords by instead storing their SHA-256 hash values.

All administrative password hashes, pre-shared keys, symmetric keys, and private keys are stored in non-volatile memory at non-fixed locations that are otherwise not accessible by administrative users.

6.8.2 FPT_FLS.1/SelfTest, FPT_TST_EXT.1, FPT_TST_EXT.3

At power-on tests are performed on each component to confirm the integrity of the firmware and a statistical assessment of the entropy source. The integrity of the firmware is tested using the cryptographic services described in FCS_COP.1/SigGen, which covers all of the executable code. The noise source health test performs a statistical assessment of 1000 samples. DRBG randomness test continuously compares the current and previously generated blocks to see if they are the same. Identical blocks indicate a failure of the generation function and produce an error and an error return value.

If any test fails, that component does not enter an operational state and the network interfaces are not activated. For GoSilent Server, an error is displayed on the local console accessible via ESXi or KVM. For GoSilent Cube, the LEDs indicate the system is not operational. In the event of an initial start-up failure the red and green status LEDs will blink in unison; in the event of a DRBG randomness test failure the red power LED will remain on, and the green status LED will be extinguished.

These tests are sufficient to demonstrate the TSF is operating correctly, as they confirm the integrity of all firmware modules prior to their execution, thereby confirming the modules have not been modified or replaced in any unauthorized manner, and they ensure the DRBG continues to operate successfully providing sufficient entropy in response to any requests.

6.8.3 FPT_ITT.1

All communication between the TOE components uses IPsec to protect the traffic.

IPsec connections are always initiated by GoSilent Cube; GoSilent Server only receives incoming connections from GoSilent Cube devices.

6.8.4 FPT_TUD_EXT.1

Authorized administrators can query the current version of the TOE software via the administration GUI. The administrator can initiate a manual update of the TOE component that they are connected to. Updates are downloaded from a specified Uniform Resource Identifier (URI) using HTTPS. On the GoSilent Server, administrators manually load the update file onto the system. On the GoSilent Cube, an option is provided on the GUI to download updates from the CACI update server. The updates are signed with a CACI key. Once the image has been downloaded, the TOE checks the signature of the image (against the CACI public key) before the image is applied.

When an update is successfully verified and installed, GoSilent Server will reboot automatically and the software update takes effect. If the update is unsuccessful or otherwise fails, GoSilent Server will not proceed with the installation and continue to execute the current software version without interruption.

Prior to the installation of a software update, the Cube will tear down any active tunnels to ensure the integrity of the Cube and update process. Once the update is successfully verified and installed on the Cube, it will reestablish the tunnel with GoSilent Server. Should the verification of the update fail or is otherwise unsuccessful, the Cube will not proceed with the installation and continue executing the currently installed software version without interruption.

6.8.5 FPT_STM_EXT.1

GoSilent Server and GoSilent Cube each maintain a system clock used to provide date and time information for TOE operations, including reliable timestamping for audit logs, and validation of certificate expiration dates. The time can be manually set by authorized administrators only.

GoSilent Server receives time information from the virtualization platform to maintain the time across system reboots. When a VM is started, the virtualization system (ESXi or KVM) provides the VM with the virtualization system's internal time value. Upon startup, the GoSilent Server sets its internal clock to that time from the virtualization system. Subsequently, the GoSilent Server internal clock is used and subsequent synchronization with the virtualization system clock is not performed. There is a delay in updating the clock until the next GoSilent Server restart. Time information obtained from the underlying virtualization platform is considered to be reliable because access to the virtualization platform, including the time setting function, is restricted to authorized administrators of the platform only. When setting the time manually on GoSilent Server, only authorized administrators have access to and can modify that function.

For GoSilent Cube, guidance directs the administrator to manually set the clock on each reboot since it does not include a real-time clock.

6.9 TOE Access (FTA)

6.9.1 FTA_SSL_EXT.1, FTA_SSL.3, FTA_SSL.4

Following an administrator configured period of inactivity (of both local and remote interactive sessions) the session will be automatically terminated, requiring re-authentication by the administrator before the access to TOE functionality can be gained.

The administrator is able to terminate their GUI session by using the logout button.

6.9.2 FTA_TAB.1

The TOE displays an administrator configurable message prior to local and remote login via the GoSilent Server GUI. The Cube also displays an administrator configurable message prior to local login via the Cube GUI.

Prior to local and remote login to GoSilent Server and local login to the Cube, the GUI displays a consent banner to the administrator warning that proceeding with authentication is consenting to the terms of use of the TOE. The user is then prompted to enter their username and password.

6.10 Trusted Path/Channels (FTP)

6.10.1 FTP_ITC.1, FTP_ITC.1/VPN

Trusted channels are used for connections between GoSilent Cube and GoSilent Server (FCS_IPSEC_EXT.1) as well as with the syslog server (FCS_TLSC_EXT.1). TLS syslog server connections are always initiated by GoSilent Server/GoSilent Cube (acting as the TLS client); assured identification is via validation of the server X.509 certificate. IPsec connections between TOE components are always initiated by the GoSilent Cube.

6.10.2 FTP_TRP.1/Admin

The TOE provides a trusted path for remote administration of the GoSilent Server and GoSilent Cube (FCS_TLSS_EXT.1). These logically distinct communications paths use TLS and HTTP to protect any transmitted data from unauthorized disclosure and leverage the ciphersuites listed in FCS_TLSS_EXT.1.

6.11 CAVP Algorithm Certificate Details

Each of these cryptographic algorithms have been validated as identified in the table below.

Table 16: CAVP Algorithms

Algorithm	Standard	Modes Supported	Library	CAVP Certificate
Cryptographic Key Generation (FCS_CKM.1)				
ECDSA KeyGen (FIPS186-4)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4, using 'NIST curves' [P-384, P-521]	P-384, P-521	CACI idt GoSilent Crypto Library v1.0	A7437
ECDSA KeyVer (FIPS186-4)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4, using 'NIST curves' [P-384, P-521]	P-384, P-521	CACI idt GoSilent Crypto Library v1.0	A7437
Cryptographic Key Generation (for IKE Peer Authentication) (FCS_CKM.1/IKE) (VPNGW)				
ECDSA KeyGen (FIPS186-5)	FIPS PUB 186-5, "Digital Signature Standard (DSS)," Appendix B.4 for ECDSA schemes, and implementing "NIST curves" P-384 and [P-521]	P-384, P-521	CACI idt GoSilent Crypto Library v1.0	A7437
ECDSA KeyVer (FIPS186-5)	FIPS PUB 186-5, "Digital Signature Standard (DSS)," Appendix B.4 for ECDSA schemes, and implementing "NIST curves" P-384 and [P-521]	P-384, P-521	CACI idt GoSilent Crypto Library v1.0	A7437

Algorithm	Standard	Modes Supported	Library	CAVP Certificate
Cryptographic Key Establishment (FCS_CKM.2)				
Elliptic curve-based key establishment (SP 800-56Ar3) (KAS-ECC)	NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"4, using 'NIST curves' [P-384, P-521]	P-384, P-521	CACI idt GoSilent Crypto Library v1.0	A7437
Cryptographic Operation (Signature Generation and Verification) (FCS_COP.1/SigGen)				
ECDSA SigGen (FIPS186-4)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-384, P-521]; ISO/IEC 14888-3, Section 6.4	P-384 with SHA-384, P-521 with SHA-512	CACI idt GoSilent Crypto Library v1.0	A7437
ECDSA SigVer (FIPS186-4)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-384, P-521]; ISO/IEC 14888-3, Section 6.4	P-384 with SHA-384, P-521 with SHA-512	CACI idt GoSilent Crypto Library v1.0	A7437
Cryptographic Operation (Hash Algorithm) (FCS_COP.1/Hash)				
SHA2-256	ISO/IEC 10118-3:2004		CACI idt GoSilent Crypto Library v1.0	A7437
SHA2-384				
SHA2-512				
Cryptographic Operation (Keyed Hash Algorithm) (FCS_COP.1/KeyedHash)				
HMAC-SHA-384	ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"	384 bits key length	CACI idt GoSilent Crypto Library v1.0	A7437

Algorithm	Standard	Modes Supported	Library	CAVP Certificate
HMAC-SHA-512		512 bits key length	CACI idt GoSilent Crypto Library v1.0	A7437
Cryptographic Operation (AES Data Encryption/Decryption) (FCS_COP.1/DataEncryption) (VPNGW)				
AES-GCM	AES: ISO 18033-3 GCM: ISO 19772	256 bits key length Payload Length: 1024	CACI idt GoSilent Crypto Library v1.0	A7437
Random Bit Generation (FCS_RBG_EXT.1)				
CTR_DRBG	Random bit generation (DRBG) services in accordance with ISO/IEC 18031:2011 using CTR_DRBG (AES)	AES-256	CACI idt GoSilent Crypto Library v1.0	A7437
Entropy				
Userspace CPU Time Jitter	SP 800-90B	SHA3-256 (conditioning)	Userspace CPU Time Jitter RNG Entropy Source v3.4.1	E300, A7546
Kernel CPU Time Jitter	SP 800-90B	SHA3-256 (conditioning)	Kernelspace CPU Time Jitter RNG Entropy Source v3.4.1	E301, A7333

6.12 Cryptographic Key Destruction

The table below describes the key zeroization provided by the TOE and as referenced in FCS_CKM.4.

Table 17: Key Zeroization

Keys/CSPs	Storage Location	How Key is Protected	How Key is Derived	When Key is Destroyed	Method of Zeroization
AES-256 key used for TLS	DRAM	Plaintext	ECDHE (P-384, P-521)	Session end or shutdown	Overwrite with zeroes
Key Agreement Keys used for IPsec	DRAM	Plaintext	ECDHE (P-384, P-521)	Session end or shutdown	Overwrite with zeroes
Authentication Keys used for IPsec	Protected file	AES-256 encrypted	KeyGen	X.509 cert deleted	Overwrite with zeroes

Keys/CSPs	Storage Location	How Key is Protected	How Key is Derived	When Key is Destroyed	Method of Zeroization
PPKs	Protected file	AES-256 encrypted	Imported	PPK deleted	Overwrite with zeroes
AES-256 Data Encryption Keys used for IPsec	DRAM	Plaintext	ECDHE (P-384, P-521), RFC 8784	Session end or shutdown	Overwrite with zeroes
Passwords	Protected file	SHA-256 hash only	SHA-256 hash of admin input	Account deleted	Overwrite with zeroes

6.13 SFR Distribution Between Components

The following table addresses the SFR distribution requirements between TOE components, as required in Section 3.1 of the Firewall Module. Since SFRs drawn from the VPN Gateway Module are not addressed by the distribution requirements in the Firewall Module, their distribution requirements in the following table have been specified to be consistent with the requirements for similar

Table 18: SFR Distribution Between Components

SFR	Dist. Requirement	GoSilent Server	GoSilent Cube
FAU_GEN.1	All	X	X
FAU_GEN.1/VPN	All	X	X
FAU_GEN.2	All	X	X
FAU_GEN_EXT.1	All	X	X
FAU_STG_EXT.1	All	X	X
FAU_STG_EXT.4	Feature Dependent	X	X
FCO_CPC_EXT.1	All	X	X
FCS_CKM.1	One	X	X
FCS_CKM.1/IKE	One	X	X
FCS_CKM.2	All	X	X
FCS_CKM.4	All	X	X
FCS_COP.1/DataEncryption	All	X	X
FCS_COP.1/SigGen	All	X	X
FCS_COP.1/Hash	All	X	X
FCS_COP.1/KeyedHash	All	X	X
FCS_HTTPS_EXT.1	Feature Dependent	X	X
FCS_IPSEC_EXT.1	Feature Dependent	X	X
FCS_RBG_EXT.1	All	X	X
FCS_TLSC_EXT.1	Feature Dependent	X	X
FCS_TLSS_EXT.1	Feature Dependent	X	X
FDP_RIP.2	Feature Dependent	X	X
FFW_RUL_EXT.1	One	X	
FIA_AFL.1	One	X	X

SFR	Dist. Requirement	GoSilent Server	GoSilent Cube
FIA_PMG_EXT.1	One	X	X
FIA_PSK_EXT.1	Feature Dependent	X	X
FIA_PSK_EXT.2	Feature Dependent	X	X
FIA_UIA_EXT.1	One	X	X
FIA_UAU.7	Feature Dependent	X	X
FIA_X509_EXT.1/ITT	Feature Dependent	X	X
FIA_X509_EXT.1/Rev	Feature Dependent	X	X
FIA_X509_EXT.2	Feature Dependent	X	X
FIA_X509_EXT.3	Feature Dependent	X	X
FMT_MOF.1/ManualUpdate	All	X	X
FMT_MOF.1/Services	Feature Dependent	X	X
FMT_MTD.1/CoreData	All	X	X
FMT_MTD.1/CryptoKeys	Feature Dependent	X	X
FMT_SMF.1	Feature Dependent	X	X
FMT_SMF.1/FFW	Feature Dependent	X	
FMT_SMF.1/VPN	Feature Dependent	X	
FMT_SMR.2	One	X	X
FPF_RUL_EXT.1	Feature Dependent	X	
FPT_APW_EXT.1	Feature Dependent	X	X
FPT_FLS.1/SelfTest	All	X	X
FPT_ITT.1	Feature Dependent	X	X
FPT_SKP_EXT.1	All	X	X
FPT_STM_EXT.1	All	X	X
FPT_TST_EXT.1	All	X	X
FPT_TST_EXT.3	All	X	X
FPT_TUD_EXT.1	All	X	X
FTA_SSL_EXT.1	Feature Dependent	X	X
FTA_SSL.3	Feature Dependent	X	X
FTA_SSL.4	Feature Dependent	X	X
FTA_TAB.1	One	X	X
FTP_ITC.1	One	X	X
FTP_ITC.1/VPN	One	X	X
FTP_TRP.1/Admin	One	X	X

7 Acronyms

The following acronyms and abbreviations appear in the document and are defined in the table below.

Table 19: Acronyms and Abbreviations

Acronym/Abbreviations	Definition
AES	Advanced Encryption Standard
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria
CRL	Certificate Revocation List
CDP	CRL Distribution Point
cPP	collaborative Protection Profile
CPU	Central Processing Unit
CSP	Cryptographic Service Provider
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
ECC	Elliptic Curve Cryptography
ECDHE	Elliptic Curve Diffie-Hellman Ephemeral
ECDSA	Elliptic Curve Digital Signature Algorithm
EKU	Extended Key Usage
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standards
GUI	Graphical User Interface
HMAC	Keyed-Hash Message Authentication Code
HTTP	Hypertext Transfer Protocol
IKE	Internet Key Exchange
IoT	Internet of Things
IP	Internet Protocol
ISO/IEC	International Organization for Standardization / International Electrotechnical Commission
IV	Initialization Vector
KDF	Key Derivation Function

Acronym/Abbreviations	Definition
LAN	Local Area Network
MOD_FW	PP-Module for Stateful Traffic Filter Firewalls
MOD_VPNGW	PP-Module for Virtual Private Network Gateways
NAT	Network Address Translation
NIAP	Nation Information Assurance Partnership
NDcPP	collaborative Protection Profile for Network Devices
OCSP	Online Certificate Status Protocol
OSP	Organization Security Policy
PKCS	Public-Key Cryptography Standards
pND	Physical Network Device
POST	Power-On Self-Test
PP	Protection Profile
PPK	Post-quantum Pre-shared Key
PRF	Pseudorandom Function
PSK	Pre-Shared Key
RBG	Random Bit Generator
RFC	Request For Comment
SA	Security Association
SFR	Security Functional Requirement
SPD	Security Policy Database
ST	Security Target
TD	Technical Decision
TLS	Transport Layer Security
TLS/HTTP	HTTP over TLS or HTTPS
TOE	Target of Evaluation
TSS	TOE Summary Specification
URI	Uniform Resource Identifier
VM	Virtual Machine
vND	Virtual Network Device

Acronym/Abbreviations	Definition
VPN	Virtual Private Network
VS	Virtualization Server
WAN	Wide Area Network