Security Target for the Fortinet FortiGate™-50B, 200A, 300A, 310B, 500A, 800, 1000A, 3016B, 3600, 3600A, 3810A-E4, 5001SX, 5001FA2, 5001A-DW and FortiWiFi-50B Unified Threat Management Solutions and FortiOS 3.0 CC Compliant Firmware: EAL4+

EWA Document No. 1523-011-D002

Fortinet Document No. ST0003

Version 0.25, 17 November 2008

Prepared for:

Fortinet, Incorporated
326 Moodie Drive
Ottawa, Ontario
Canada K2H 8G3

Prepared by:

Electronic Warfare Associates-Canada, Ltd.
55 Metcalfe St., Suite 1600
Ottawa, Ontario
K1P 6L5

Security Target for the Fortinet FortiGate™-50B, 200A, 300A, 310B, 500A, 800, 1000A, 3016B, 3600, 3600A, 3810A-E4, 5001SX, 5001FA2, 5001A-DW and FortiWiFi-50B Unified Threat Management Solutions and FortiOS 3.0 CC Compliant Firmware: EAL4+

EWA Document No. 1523-011-D002

Fortinet Document No. ST0003

Version 0.25, 17 November 2008

<Original> Approved by:

Project Engineer: S. Moore 17 November 2008

Project Manager: G. Gibbs 17 November 2008

Program Director: E. Connor 17 November 2008

(Signature) (Date)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 IDENTIFICATION

This document is the Security Target (ST) for the FortiGate™ Unified Threat Management Solutions detailed in Table 1.

Product	Firmware[1] Version	Hardware Version[2]	FIPS 140-2 Certificate Number
FortiGate-50B	3.00, build 8880, 080917	C5GB38	Crypto Module Certificate: 945 Algorithm Certificates: see note 3
FortiGate-200A	3.00, build 8880, 080917	C4AY89	Crypto Module Certificate: 807, 905 Algorithm Certificates: see note 2
FortiGate-300A	3.00, build 8880, 080917	C4FK88	Crypto Module Certificate: 807, 905 Algorithm Certificates: see note 2
FortiGate-310B	3.00, build 8880, 080917	C4ZF35	Crypto Module Certificate: Crypto module testing for this unit has been successfully completed under the CMVP. Posting to the 'In Review' section of the CMVP Modules in Process List is pending. Algorithm Certificates: see note 4
FortiGate-500A	3.00, build 8880, 080917	C4BE21	Crypto Module Certificate: 807,905 Algorithm Certificates: see note 2

[1] The firmware is assigned a version number that is identical to the version number of the software that is loaded onto it. The firmware version number is shown here because the operational program for the FortiGate series is stored in firmware.

[2] For the purposes of the ST, only the first 6 characters of the hardware version are relevant. The complete version includes a padding field for compatibility with other Fortinet version naming conventions and a field for non-CC relevant changes such as the amount of memory, CPU clock speed or external labelling.

Product	Firmware[1] Version	Hardware Version[2]	FIPS 140-2 Certificate Number
FortiGate-800	3.00, build 8880, 080917	C4UT39	Crypto Module Certificate: 905 Algorithm Certificates: see note 2
FortiGate-1000A	3.00, build 8880, 080917	C4WA49	Crypto Module Certificate: 810 Algorithm Certificates: see note 2
FortiGate-3016B	3.00, build 8880, 080917	C4XA14	Crypto Module Certificate: Listed under ‘Co-ordination’ on the CMVP Modules in Process List as of 23 Oct 2008 Algorithm Certificates: see note 4
FortiGate-3600	3.00, build 8880, 080917	C4KW75	Crypto Module Certificate: 810 Algorithm Certificates: see note 1
FortiGate-3600A	3.00, build 8880, 080917	V3BU94	Crypto Module Certificate: Listed under ‘Co-ordination’ on the CMVP Modules in Process List as of 23 Oct 2008 Algorithm Certificates: see note 4
FortiGate-3810A-E4	3.00, build 8880, 080917	C3GV75	Crypto Module Certificate: Listed under ‘Co-ordination’ on the CMVP Modules in Process List as of 23 Oct 2008 Algorithm Certificates: see note 4
FortiGate-5001SX	3.00, build 8880, 080917	P4CF76	Crypto Module Certificate: 789 Algorithm Certificates: see note 1
FortiGate-5001FA2	3.00, build 8880, 080917	P4CF76	Crypto Module Certificate: 789 Algorithm Certificates: see note 1

Product	Firmware[1] Version	Hardware Version[2]	FIPS 140-2 Certificate Number
FortiGate-5001A-DW	3.00, build 8880, 080917	P4CJ36	Crypto Module Certificate: Crypto module testing for this unit has been successfully completed under the CMVP. Posting to the 'In Review' section of the CMVP Modules in Process List is pending. Algorithm Certificates: see note 4
FortiWiFi-50B	3.00, build 8880, 080917	C5WF27	Crypto Module Certificate: Listed as 'In Review' on the CMVP Modules in Process List as of 23 Oct 2008 Algorithm Certificates: see note 3

Table 1 - TOE Identification Details

Note 1 – The following FIPS 140-2 algorithm certificates are applicable:

Triple-DES:	486, 487, 490
AES:	471, 472, 476
SHS:	539, 540, 544
HMAC:	228, 229, 233
RSA:	193
RNG:	251

Note 2 – The following FIPS 140-2 algorithm certificates are applicable:

Triple-DES:	486, 487, 489
AES:	471, 472, 475
SHS:	539, 540, 543
HMAC:	228, 229, 232
RSA:	193
RNG:	251

Note 3 – The following FIPS 140-2 algorithm certificates are applicable:

Triple-DES:	489, 583, 584
AES:	475, 613, 614
SHS:	543, 661, 662

- HMAC: 232, 316, 317
- RSA: 285
- RNG: 345

Note 4 – The following FIPS 140-2 algorithm certificates are applicable:

- Triple-DES: 582, 583, 584
- AES: 612, 613, 614
- SHS: 660, 661, 662
- HMAC: 315, 316, 317
- RSA: 284, 285
- RNG: 345

The products listed in Table 1 are collectively termed the FortiGate Series or FortiGate Family of Unified Threat Management Solutions.

Documentation for the FortiGate Series operated in Common Criteria mode consists of the standard FortiOS version 3.0 documentation set plus a FIPS-CC-specific technical note.

This ST has been prepared in accordance with the Common Criteria for Information Technology Security Evaluation (CC), Version 2.3, August 2005, CCIMB-2005-08-001 -002 and -003, with all current interpretations[3].

1.2 FORTIGATE™ UNIFIED THREAT MANAGEMENT SOLUTIONS OVERVIEW

The FortiGate family of Unified Threat Management Solutions span the full range of network environments, from the remote office and branch office (ROBO) to service provider, offering cost-effective systems for any size of application. They are hardware security systems designed to protect computer networks from abuse. They reside between the network they are protecting and an external network such as the internet, restricting the information flow between the networks to that permitted by a policy (set of rules) defined by the Security Administrator. They detect and eliminate the most damaging, content-based threats from email and Web traffic such as viruses, worms, intrusions, inappropriate Web content and more in real-time; without degrading network performance. In addition to providing stateful application-level protection, the FortiGate series deliver a full range of network-level services including; Virtual Private Network (VPN), Network Address

[3] This ST claims conformance with a Protection Profile (PP), and includes SFRs from PPs that are based upon Version 2.1 of the CC. However the ST also includes requirements which are in addition to the requirements levied by the PPs. These additional requirements are drawn from Version 2.3 of the CC.

Translation (NAT)[4], intrusion prevention, web filtering, antivirus, antispam and traffic shaping; using dedicated, easily managed platforms.

Each FortiGate unit consists of a hardware box and the FortiOS™ custom Unified Threat Management Solution firmware. Administration of the system may be performed locally using an administrator console or remotely via a network management station. The FortiGate Unified Threat Management Solution can operate either alone or as part of a cluster in order to provide high availability of services. The models offered in the FortiGate Series share common source code but different firmware builds due to different device drivers. The different models in the series provide for increased performance and additional protected ports.

All FortiGate Unified Threat Management Solutions employ Fortinet's unique FortiASIC™ content processing chip and the powerful, secure, FortiOS™ operating system to achieve breakthrough price/performance. Their unique, ASIC-based architecture analyzes content and behaviour in real time, enabling key applications to be deployed right at the network edge, where they are most effective at protecting enterprise networks. They provide a critical layer of real-time, network-based antivirus protection that complements host-based antivirus software and supports "defense-in-depth"strategies without compromising performance or cost. They can be deployed to provide antivirus protection, antispam protection and content filtering in conjunction with existing firewall, VPN, and related devices, or to provide complete network protection.

The FortiGate series support the IPSec industry standard for VPN, allowing VPNs to be configured between a FortiGate model and any client or gateway/firewall that supports IPSec VPN. The FortiGate series also provide SSL VPN services.

The FortiGate's firewall, VPN, antivirus and intrusion prevention functionality are within the scope of this evaluation. Features such as antispam, content filtering and traffic shaping have been placed outside the TOE boundary for this evaluation. Section 2 provides a detailed description of the product functionality which is included in the TOE and a list of the product functionality which is excluded from the TOE.

1.3 CC CONFORMANCE

This ST contains functional requirements based upon functional components in CC Part 2 as well as a number of explicitly-defined functional requirements. The Target of Evaluation (TOE) for this ST, the FortiGate Unified Threat Management Solution, is therefore conformant with CC Part 2 extended.

[4] Network Address Translation is only applied after an information flow has been allowed by the rules which implement the FortiGate's security policy enforcement. For this reason the use of NAT by the FortiGate is not a security relevant feature of the TOE.

The TOE for this ST is conformant with the Intrusion Detection System Sensor Protection Profile (IDSS PP), Version 1.2, April 27, 2005.

The FortiGate Unified Threat Management Solution also includes security functional requirements listed in the following Protection Profiles (PP):

- U.S. Government Traffic-Filter Firewall Protection Profile for Medium Robustness Environments, Version 1.1, January 9, 2006 (TFFW PP MR);
- U.S. Government Firewall Protection Profile for Medium Robustness Environments, Version 1.0, October 28, 2003 (FW PP MR); and
- U.S. Government Virtual Private Network (VPN) Boundary Gateway Protection Profile for Medium Robustness Environments, Version 1.0, February 23, 2006 (VPN PP MR).

In addition to the security functional requirements levied by conformance with the IDSS PP and the requirements taken from the TFFW PP MR, FW PP MR and VPN PP MR, the TOE also satisfies the requirements of the following additional functional requirements drawn from Part 2 of the CC

- FIA_ATD.1(2)[5] – User attribute definition (authorized proxy users)
- FIA_ATD.1(3) – User attribute definition (VPN Peers)
- FMT_MOF.1(8)[6] – Management of security functions behaviour (cryptographic self-test frequency)
- FMT_MOF.1(9) – Management of security functions behaviour (audit storage exhaustion)
- FMT_MOF.1(10) – Management of security functions behaviour (session termination)
- FMT_MOF.1(11) – Management of security functions behaviour (alarm acknowledgement)

[5] The IDSS PP, FW PP MR, TFFW PP MR and VPN PP MR all include one iteration of the FIA_ATD.1 requirement which specifies user attributes for administrators. This ST introduces two additional interations of the requirement, in order to define the security attributes for authorized proxy users and VPN Peers.

[6] The three MR PPs specify seven iterations of the FMT_MOF.1 requirement. The IDSS PP introduces an additional iteration of the requirement which in this ST has been listed as iteration (13). The ST also includes five additional iterations of the requirement, numbered from (8) through (12) to cover features of the TOE which are in addition to the requirements of the PPs.

- FMT_MOF.1(12) – Management of security functions behaviour (self-tests)
- FMT_MSA.2 – Secure security attributes
- FMT_MTD.1(1)[7] – Management of TSF data (audit data)
- FMT_MTD.1(5) – Management of TSF data (user accounts)
- FMT_MTD.1(6) – Management of TSF data (TOE banner)
- FMT_MTD.1(7) – Management of TSF data (AV and IPS signatures)
- FPT_AMT.1 – Abstract Machine Testing
- FPT_FLS.1 – Failure with preservation of secure state
- FRU_FLT.1 – Degraded fault tolerance

Additionally, this ST includes the following explicit security functional requirements which are not drawn from any of the PPs listed above. These requirements were added in order to specify the Anti Virus and Intrusion Prevention capabilities of the FortiGate Unified Threat Management Solution.

- FAV_ACT_EXP.1 – Anti Virus Actions
- FIP_ACT_EXP.1 – Intrusion Prevention Actions

Although the TFFW PP MR, the FW PP MR and the VPN PP MR include extended security assurance requirements, this ST has not used the extended requirements and instead has drawn all of its security assurance requirements from Part 3 of the CC. Therefore the ST is conformant with CC Part 3.

The TOE for this ST is conformant to the CC Part 3 assurance requirements for EAL 4, augmented with ALC_FLR.3 – Systematic Flaw Remediation.

[7] The FW PP MR and TFFW PP MR include four iterations of the FMT_MTD.1 requirement. However only the last three of these iterations describe actual requirements of the TOE. The first of the iterations is intended to allow the ST author to describe additional TSF data management capabilities of the TOE. Since there are four additional TSF data management functions which need to be included, these have been given the iteration numbers (1), (5), (6) and (7). The VPN PP MR includes two of the requirements from the FW PP MR and TFFW PP MR (iterations (2) and (3)) and a third iteration which in this ST has been given iteration number (8). The IDSS PP adds one additional TSF data management function which has been given iteration number (9).

1.4 CONVENTIONS

The CC permits four types of operations to be performed on functional requirements: selection, assignment, refinement, and iteration. These operations, when performed on requirements deriving from the PP, or when performed on requirements that derive from CC Part 2 and which do not appear in the PP, are identified in this ST in the following manner:

- Selection: Indicated by surrounding brackets, e.g., [selected item]. To improve readability selections of [none] are generally not shown, however in cases where such a selection has been omitted, the omission is noted in Section 7.2.

- Assignment: Indicated by surrounding brackets and italics, e.g., [*assigned item*]. To improve readability assignments of [*none*] are generally not shown, however in cases where such an assignment has been omitted, the omission is noted in Section 7.2.

- Refinement: Refined components are identified in three ways; (1) they are listed in Table 6 - Security Functional Requirements by using bold text, (2) the word **Refinement:** (in bold text) is added to the requirement statement in Section 5, and a description of the refinement is included in Section 7.2 IDSS PP TAILORING. It should be noted that the IDSS PP includes numerous refinements to functional requirements taken from the CC. However these refinements are NOT indicated in this document. The only refinements marked in this document are those which have been made to the text of the requirements listed in the IDSS PP or to the text of a requirement drawn from the CC which is not included in the IDSS PP.

- Iteration: Indicated by assigning a number in parenthesis to the end of the functional component identifier as well as by modifying the functional component title to distinguish between iterations, e.g., 'FDP_IFC.1(1), Subset information flow control (unauthenticated policy)' and 'FDP_IFC.1(2) Subset information flow control (authenticated policy)'.

This ST is based on the IDSS PP. As noted previously, the ST also includes some requirements taken from CC Part 2 and Part 3 that are not in the protection profile. The ST also includes requirements taken from the FW PP MR, the TFFW PP MR and the VPN PP MR. Deviations in phrasing from the IDSS PP text are noted as refinements. For non-IDSS PP requirements deviations from the CC text are noted as refinements.

1.5 TERMINOLOGY

The following terminology is used in this ST:

Administrator

An Administrator is responsible for administering the TOE. The TOE has three administrative roles; Audit Administrator, Security Administrator, and Cryptographic Administrator. Administration is performed using the Administrator Interfaces which consist of the Local Console, Network Web-Based GUI, and Network CLI. Wherever possible, the ST uses the specific administrator role. However in some instances a function may be available to any member of one of the three administrative roles. In these cases the ST uses the generic term 'Administrator' to denote that the function may be performed by any member of an administrative role.

Attack Potential

The perceived potential for success of an attack, should an attack be launched, expressed in terms of an attacker's expertise, resources and motivation.

Controlled Subject

Entity under control of the TOE Security Policy (TSP).

Presumed Address

The TOE can make no claim as to the real address of any source or destination subject, therefore the TOE can only suppose that these addresses are accurate. Therefore, a 'presumed address' is used to identify source and destination addresses.

Protection Profile

Both the Common Criteria and Fortinet use the term Protection Profile. The appropriate definitions for both usages of the term may be found in Section 10. Within the document, the context generally makes it clear which usage is appropriate. However, for clarity, the CC usage is generally noted by the abbreviation PP while the Fortinet usage is denoted by spelling out the complete term.

User

A User is an entity that uses the TOE's services to pass information through the TOE over the Network Interfaces. Authentication is required for some services. An 'authenticated proxy user' denotes a user who has been identified and authenticated by the TOE.

Local Console

A management console (may be a computer workstation or VT100 type terminal) connected directly to the TOE. Although the Local Console falls outside the TOE Boundary it is located in the same physical location as the TOE and therefore is provided with the same physical protection as is provided for the TOE.

Network Management Station

A computer located remotely from the TOE but which is able to establish a network connection to the TOE. The Network Management Station falls outside the TOE Boundary.

Firewall Rules	Firewall rules are configuration parameters set by the Security Administrator that allow or deny data flow through the TOE. These rules may optionally include the use of a firewall protection profile that enforces Anti-Virus (AV) and Intrusion Prevention System (IPS) configuration parameters.

1.6 DOCUMENT ORGANIZATION

Section 1, Introduction, provides the document management and overview information necessary to identify the ST along with references to the PP to which conformance is being claimed.

Section 2, Target of Evaluation Description, defines the TOE and establishes the context of the TOE by referencing generalized security requirements.

Section 3, TOE Security Environment, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.

Section 4, Security Objectives, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment.

Section 5, IT Security Requirements, specifies the security functional and assurance requirements that must be satisfied by the TOE and the Information Technology (IT) environment.

Section 6, TOE Summary Specification, describes the security functions and assurance measures that are included in the TOE to enable it to meet the IT security functional and assurance requirements.

Section 7, Protection Profile Claims, provides reference to the PP to which adherence is claimed by this ST. This section also describes the changes that were made with respect to the PP.

Section 8, Rationale, provides rationale to demonstrate that each section of the ST is traceable to previous sections. It provides rationale that the security objectives satisfy the threats and policies, that the security functional and assurance functional requirements satisfy the objectives, and that the TOE summary specification satisfies the security requirements. This section also presents rationale for any dependencies that are not satisfied, a rationale for the Strength of Function (SOF) claim, and a rationale for the explicit requirements.

Section 9, References, provides background material for further investigation by users of the ST.

Section 10, Terminology, provides definitions for specific terms used in the ST.

Section 11, Acronyms, Abbreviations, and Initializations, provides expansions for the acronyms, abbreviations, and initializations that are used in the document. Common CC terminology has been excluded from this list.

2 TARGET OF EVALUATION DESCRIPTION

2.1 SCOPE AND BOUNDARIES OF EVALUATED CONFIGURATION

2.1.1 Physical Boundary

2.1.1.1 Physical Configuration

The FortiGate-50B, 200A, 300A, 310B, 500A, 800, 1000A, 3016B, 3600, 3600A, 3810A-E4 and FortiWiFi-50B are stand-alone appliances that do not require supporting hardware. The FortiGate-5001SX, 50001FA2 and 5001A-DW are Unified Threat Management Solution modules (blades) that may be installed in the FortiGate-5050 or 5140 chassis, each of which is capable of holding multiple blades. The chassis supports the blades by providing mounting, power and cooling fans only. As network and management interfaces are part of the blade itself, each blade acts as an independent Unified Threat Management Solution.

Each member of the series of FortiGate Unified Threat Management Solutions, termed a FortiGate unit, consists of custom hardware and firmware. The FortiGate unit consists of the following major components: FortiOS FIPS-CC compliant firmware, processor, memory, FortiASIC™, and I/O interfaces. The FortiGate-3600A, FortiGate-3810-E4 and FortiGate-5001SX models offer dual processors in order to increase performance. All models share a common software platform and use a proprietary Application-Specific Integrated Circuit (FortiASIC™) to improve performance. The FortiASIC™ is a hardware device which forms part of the FIPS 140-2 validated cryptographic module used by each FortiGate unit. The FortiASIC™ performs security and content processing.

2.1.1.2 Physical Interfaces

The FortiGate units have the interfaces defined in Table 2.

Product	Interfaces				Log Storage Type and Maximum Size
	Network (Ethernet) Interfaces		Administrator Interfaces		
	No.	Speed	Local Console	Network	
FortiGate-50B	5	10/100 Base-T	RS232/RJ-45	Yes	RAM Configurable 612KB – 3.2MB
FortiGate-200A	8	10/100 Base-T	RS232/RJ-45	Yes	RAM Configurable 648KB – 25.6MB

Product	Interfaces				Log Storage Type and Maximum Size
	Network (Ethernet) Interfaces		Administrator Interfaces		
	No.	Speed	Local Console	Network	
FortiGate-300A	6	10/100 Base-T	RS232/DB-9	Yes	Hard Drive 30 GB[8]
FortiGate-310B	10	10/100/1000 Base-T	RS232/RJ-45	Yes	RAM Configurable 648KB – 51.2MB
FortiGate-500A	8	10/100 Base-T	RS232/RJ-45	Yes	Hard Drive 30 GB[8]
	2	10/100/1000 Base-T			
FortiGate-800	4	10/100/1000 Base-T	RS232/RJ-45	Yes	Hard Drive 30 GB[8]
	4	10/100 Base-T			
FortiGate-1000A	10	10/100/1000 Base-T	RS232/RJ-45	Yes	RAM Configurable 864KB – 51.2MB
FortiGate-3016B	2	10/100 Base T	RS232/RJ-45	Yes	RAM Configurable 1.7MB – 102.4MB
	16	1 GBit SFP			
	1	AMC Card Slot[9]			
FortiGate-3600	1	10/100 Base T	RS232/DB-9	Yes	Hard Drive 15 GB
	4	1000 Base SX			
	2	1000 Base-T			
	2	1 GBit SFP			
	1	AMC Card Slot[9]			
FortiGate-3600A	8	10/100 Base T	RS232/RJ-45	Yes	RAM Configurable 1.7MB – 102.4MB
	2	1 GBit SFP			
	1	AMC Card Slot[9]			

[8] The hard drives have 40 GB capacity, of which 75% is reserved for audit logs.

[9] AMC cards are like mini-blades, hot-swappable, supporting multiple connectors per card.

Product	Interfaces				Log Storage Type and Maximum Size
	Network (Ethernet) Interfaces		Administrator Interfaces		
	No.	Speed	Local Console	Network	
FortiGate-3810A-E4	8	10/100/1000 Base-T	RS232/RJ-45	Yes	RAM Configurable 1.7MB – 204.8MB
	2	1 GBit SFP			
	4	AMC Card Slot[9]			
FortiGate-5001SX	4	10/100/1000 Base T	RS232/DB-9	Yes	RAM Configurable 1.7MB – 102.4MB
	4	1 GBit SFP			
FortiGate-5001FA2	4	10/100/1000 Base T	RS232/DB-9	Yes	RAM Configurable 1.7MB – 102.4MB
	2	1 GBit SFP			
	2	1 GBit SFP (hardware accelerated)			
	2	1 GBit SFP (hardware accelerated)			
FortiGate-5001A-DW	2	1000 Base-T	RS232/RJ-45	Yes	RAM Configurable 1.7MB – 102.4MB
FortiWiFi-50B	5	10/100 Base-T	RS232/RJ-45	Yes	RAM Configurable 612KB – 3.2MB
	2	WiFi 802.11b and 802.11g			

Table 2 - FortiGate Unified Threat Management Solution Interfaces

The FortiGate units may be securely administered over the external or internal networks or locally within the secure area. Depending on the model, the FortiGate unit provides the following administration options:

- A dedicated console port is available on all models. The port is RS232 with either a DB-9 or RJ-45 connector. When connected to a terminal which supports VT100 emulation, the console port allows access to the FortiGate unit via a Command Line Interface (CLI). This Local Console CLI permits a Security Administrator to configure the FortiGate unit, monitor its operation and examine the audit logs that are created.
- On all models remote administration may be performed via any network port that has been configured by a Security Administrator to allow HTTPS (for the Network Web-Based GUI) and SSH (for the Network CLI) traffic. When connected to a Network Management Station, this port provides remote access to the Network CLI or to the Network Web-Based GUI and allows an

authorized administrator to configure the FortiGate Unit, monitor its operation and examine the audit logs that are created;

- On models equipped with a USB port the Cryptographic Administrator may perform key loading using a FortiUSB token;
- On all models, the Security Administrator may configure automatic Anti-Virus and IPS updates, from the FortiGuard Distribution Server; and
- Models FortiGate-300A, 500A, 800, 1000A, 3016B, 3600, 3600A and 3810A-E4 are equipped with a Local Control Panel. The input portion of this panel is disabled in FIPS-CC mode, but the LCD portion provides limited status information to the Administrator.

The FortiGate units are designed to be installed and used in an environment that is configured and controlled in accordance with administrator guidance that is supplied with the product.

2.1.1.3 TOE Boundary - Single-Unit Configuration

In the Single-Unit configuration, which is supported by all of the FortiGate series, the TOE consists of a single FortiGate. The FortiGate series control network access by implementing the classic firewall concepts, in which the firewall is linked to two or more networks and controls the transfer of data between the networks. The configuration supports additional networks, each of which is physically connected to one of the Network Interfaces identified in Table 2.

Figure 1 shows an example of a single FortiGate mediating information flow between two networks. One of the networks provides access to the FortiGuard Distribution Server, which permits Anti-Virus and IPS updates to be downloaded.

The Local Console, located within a Secure Area, is a terminal or general purpose computer with a standard serial interface and optional ethernet interfaces. A serial port is required to administer the TOE via the Local Console CLI.

The Network Management Station is a terminal or general purpose computer with a standard network interface which is used to remotely administer the TOE using the Network Web-Based GUI or Network CLI.

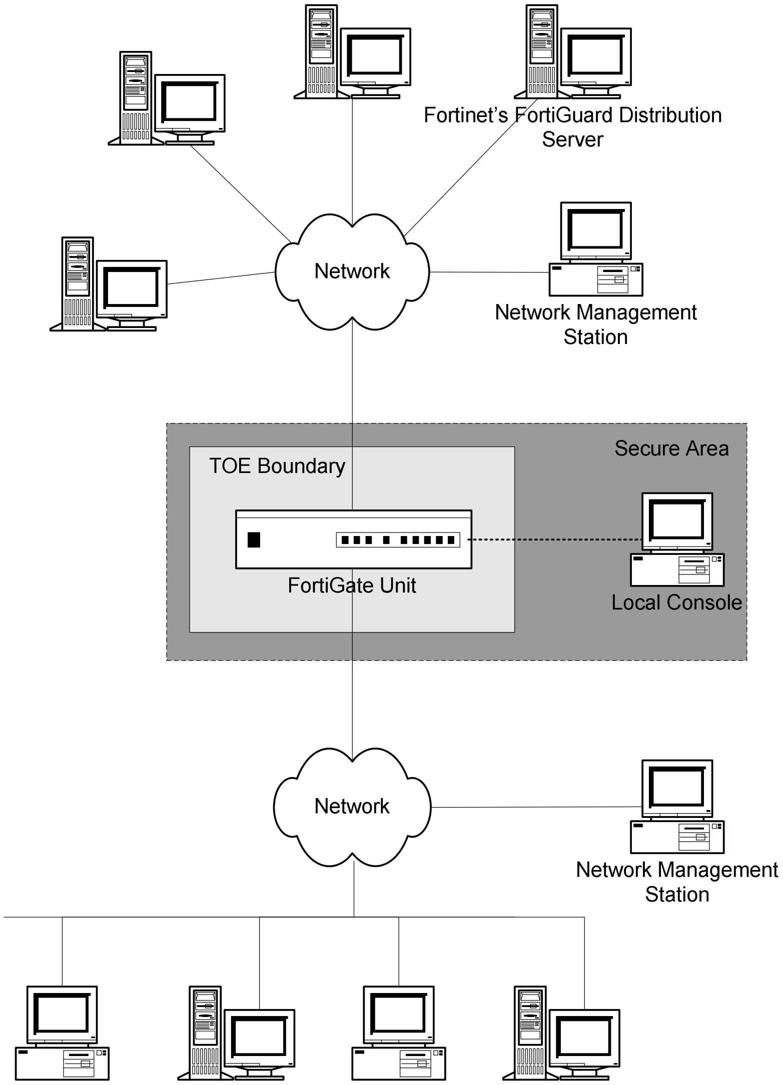
Fortinet's FortiGuard Distribution Server
Network
Network Management Station
TOE Boundary
Secure Area
FortiGate Unit
Local Console
Network
Network Management Station

Figure 1 – Single Unit FortiGate Unified Threat Management Solution Network Configuration

2.1.1.4 TOE Boundary - High-Availability Configuration

In the High-Availability (HA) configuration, which is supported by all of the FortiGate series, the TOE consists of a two or more FortiGates interconnected to form a FortiGate Cluster. The FortiGate Cluster controls network access by implementing the classic firewall concepts, in which the firewall is linked to two or more networks and controls the transfer of data between the networks. The configuration supports additional networks, each of which is physically connected to one of the Network Interfaces identified in Table 2.

Figure 2 shows three FortiGates of the same type configured in High Availability mode to form a FortiGate Cluster. A FortiGate Cluster may be configured to work in active-passive mode for failover protection or in active-active mode for failover protection and load balancing. Both active-passive mode and active-active mode are part of the evaluated configuration of the TOE. The cluster units share state and configuration information over a dedicated High Availability Link. One of the networks provides access to the FortiGuard Distribution Server, which permits Anti-Virus and IPS updates to be downloaded.

The Local Console, located within a Secure Area, is a terminal or general purpose computer with a standard serial interface and optional ethernet interfaces. A serial port is required to administer the TOE via the Local Console CLI.

The Network Management Station is a terminal or general purpose computer with a standard network interface to remotely administer the TOE using the Network Web-Based GUI or Network CLI.

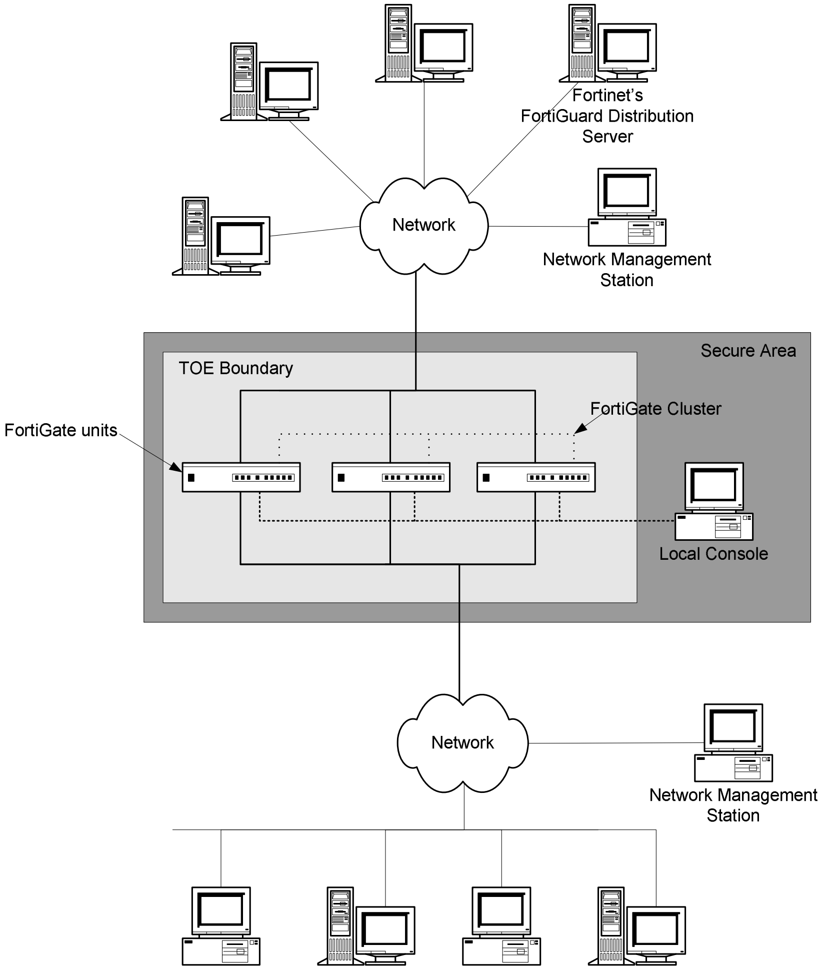

Figure 2 – High Availability FortiGate Unified Threat Management Solution Configuration

2.1.2 Logical Boundary

The logical boundary of the TOE includes all interfaces and functions within the physical boundary that are not specifically excluded in Section 2.1.3.

2.1.2.1 Logical Interfaces

Table 3 describes each of the interfaces that are included in the TOE in terms of the external entity to which it connects, the interface data that is transferred, the purpose of the interface and the protocol used for the transfer.

External Entity	Interface Data	Interface Purpose	Protocol(s)
Network Management Station	Administration Data	Allow remote administration using the CLI command interface	SSH
Network Management Station	Administration Data	Allow administration using the Web-Based GUI.	HTTPS
Certificate Server	Certificates/CRLs	Transfer certificates and certificate revocation lists to the FortiGate.	X.509
VPN Peer/Server	VPN Configuration	Configuration of VPN tunnels between the FortiGate and a remote peer or server.	IPSec/IKE
Local Console	Administration Data	Allow local administration using the CLI command interface	Serial
Local Console	Alarms	Transfer alarms to the local console.	Serial
Network User	User Data	Send and receive user data to/from the Network Users.	TCP/IP and protocols built on it.
FortiGate Cluster	High Availability Data	Exchange data to configure and synchronize the FortiGates that form a High Availability cluster.	FortiGate Clustering Protocol (FGCP)
Fortinet's FortiGuard Distribution Server	AV/Attack Updates	Transfer anti-virus and attack updates from Fortinet to the FortiGate Unit.	TCP/IP and protocols built on it.
FortiUSB	Keys	Allow the Cryptographic Administrator to load cryptographic keys.	Serial (USB)

Table 3 - FortiGate Interfaces

2.1.2.2 Functions Included in the TOE

The function of the FortiGate Series is to isolate two or more networks from each other and arbitrate the information transfers between these networks. Arbitration is based on a set of policies (rules) that are established by the Security Administrator and applied to each data packet that flows through the system. The TOE arbitrates all data that travels through it from one network to another.

The FortiGate has a FIPS-CC Mode which, when enabled by the Security Administrator, provides the capability claimed in this ST. FIPS-CC Mode provides initial default values, makes excluded features unavailable by default, and enforces the FIPS configuration requirements.

Table 4 summarizes the FortiGate features that are included in the TOE.

Feature	Description
Access Control	The FortiGate Unified Threat Management Solution provides a role-based access control capability to ensure that only authorized administrators are able to administer the FortiGate unit.
Administration (Network CLI)	The FortiGate provides management capabilities via a text-based Network CLI interface.
Administration (Local Console CLI)	The FortiGate provides management capabilities via a text-based Local Console CLI.
Administration (Network Web-Based GUI)	The FortiGate provides a Network Web-Based GUI, accessed via HTTPS, for system management and configuration.
Alarms and Alerts	The FortiGate provides audible and visible alarms that announce detected security policy violations.
Anti-Virus	The FortiGate Series provides anti-virus protection for web HyperText Transfer Protocol (HTTP), File Transfer Protocol (FTP), and email (Simple Mail Transfer Protocol (SMTP), Post-Office Protocol Version 3 (POP3), and Internet Message Access Protocol (IMAP)) content as it passes through the FortiGate unit.
Authentication	The FortiGate implements a username and password mechanism for identification and authentication.
Authentication (Firewall Policy Authentication)	The FortiGate Firewall Policy may be configured to require authentication by the user before the information flow is enabled for that user.
Certificate Management	The FortiGate provides the ability to obtain certificates and certificate revocation lists from an external certificate management server.
Cryptography	The FortiGate incorporates a FIPS 140-2 validated cryptographic module.

Feature	Description
Firewall (Information Flow Control)	The FortiGate Unified Threat Management Solution implements a stateful traffic filtering firewall. Information flow is restricted to that permitted by a policy (set of rules) defined by the Security Administrator. The default policy is restrictive (i.e., no traffic flows without Security Administrator action to configure policy).
FortiUSB	The FortiGate provides for key loading via the USB port.
High Availability (FortiGate Cluster)	The FortiGate Series provides a high availability capability between two or more identical units communicating via the FortiGate clustering protocol. Two modes of operation are supported: active-passive for failover protection and active-active for failover protection and load balancing.
ICMP	The FortiGate responds to Internet Control Message Protocol (ICMP) pings without requiring that the user be authenticated. It also passes ICMP through in accordance with policies.
Intrusion Prevention	The FortiGate uses signatures to detect and prevent attacks to the data passing through it. The intrusion prevention system (IPS) attack signatures can be updated manually or the FortiGate unit can be configured to automatically download updates. The TOE also includes local anomaly detection to protect itself from direct attacks such as denial of service (DOS) attacks.
Logging (management)	The FortiGate supports management activities for configuration of logging, retention of logs, archiving of logs, and backing up of logs.
Logging (recording)	Logging is performed and data is stored in memory, written to hard disk, or written to a FLASH memory card, depending on the model.
Protection Profile[10]	Protection profiles are used to configure anti-virus protection, and IPS.
Proxies	Firewall rules may be defined that are applicable only to users who have authenticated to the firewall in order to use a proxy service. The evaluated configuration only supports user authentication for the FTP, HTTP and Telnet protocols.
Residual Data	All residual information in any resource is over-written or otherwise destroyed such that it cannot be reused or otherwise accessed either inadvertently or deliberately.
Static Routing	Static routes are configured by defining the destination IP address and netmask of packets that the FortiGate unit is intended to intercept, and specifying a (gateway) IP address for those packets. The gateway address specifies the next-hop router to which traffic will be routed.
Self-test	The FortiGate performs self-tests of both the cryptographic and the non-cryptographic functions.

[10] The term 'Protection Profile' is also used by Fortinet and is not to be confused with the CC terminology.

Feature	Description
Time	The FortiGate maintains internal time on a system clock, settable by the Security Administrator. This clock is used when time stamps are generated.
VPN	The FortiGate supports Virtual Private Networking (VPN) using SSL or IPSec to provide a secure connection between widely separated office networks or securely link telecommuters or travellers to an office network.

Table 4 - Features Included in the TOE

2.1.3 Exclusions

The FortiGate provides more capability than is being claimed in the ST. When FIPS-CC Mode is enabled to place the TOE into the evaluated configuration, the excluded features are not enabled. With the exception of dynamic routing and the local control panel, the excluded features could be enabled by an Administrator though this would contravene the CC-specific guidance that is provided to the Administrator. When the TOE is in FIPS-CC Mode the dynamic routing function and access via the local control panel are disabled and can not be enabled without exiting FIPS-CC Mode.

Table 5 presents a summary of the features that are excluded from the TOE. These features do not contribute to any of the SFRs claimed in this ST.

Feature Excluded	Description
Administration (FortiManager)	Multiple FortiGate units may be managed by a FortiManager Server.
Administration (Local Control Panel)	The FortiGate provides a limited management interface via a LCD and associated buttons. Input via this interface is disabled in FIPS-CC Mode.
Alert Emails	In addition to alerts, the FortiGate can be configured to provide email notification.
Authentication (Active Directory)	Windows Active Directory Server may be used to authenticate users.
Authentication (RADIUS)	The FortiGate provides an option of using an external RADIUS Server for administrator authentication.
Authentication (User Group Firewall Policy Authentication)	The FortiGate Firewall Policy may be configured to require authentication by user groups before the information flow is enabled. A user group is a list of users or Radius Servers, or LDAP servers. These groups may be used in the Firewall Policy to require authentication by group rather than individually.
Authentication (LDAP)	The FortiGate provides an option of using an external LDAP Server for authentication.
Backup Configuration	The FortiGate provides a means by which the Security Administrator can back up the configuration.

Feature Excluded	Description
DHCP	The FortiGate can operate as a DHCP Server and as a DHCP relay.
Differentiated Services	The FortiGate supports differentiated services, as defined by Request for Comments (RFC) 2474 and RFC 2475.
DNS	The FortiGate can operate as a DNS server and as a DNS relay.
Dynamic Routing	Dynamic routes are configured through dynamic routing protocols that enable the FortiGate unit to automatically share information about routes with neighbouring routers and learn about routes and networks advertised by neighbouring routers.
Engine Update	The FortiGate anti-virus and IPS engines may be updated.
Firmware Update	The FortiGate firmware may be updated through a. SSL/TLS link (default method); or b. bootstrap Trivial File Transfer Protocol (TFTP) to install new firmware or replace existing configuration or firmware (disabled in FIPS-CC Mode).
Instant Messaging	The FortiGate unit is able to check Instant Messaging (IM) communications and block, rate limit, pass, and bandwidth limit the IM traffic. This capability of the TOE is excluded from the evaluation. However, a FortiGate unit is also capable of scanning IM/P2P traffic for viruses and this capability is included in the evaluation.
IPv6	Both an IPv4 and an IPv6 address may be assigned to any interface on a FortiGate unit. The interface functions as two interfaces, one for IPv4-addressed packets and another for IPv6-addressed packets. The FortiGate series support static routing, periodic router advertisements, and tunneling of IPv6-addressed traffic over an IPv4-addressed network.
Logging	The FortiGate unit is able to send log information to external servers (e.g., FortiAnalyzer, (formerly known as FortiLog) Server, ftp, Syslog Server, tftp, or WebTrends Server).
NTP Clock Setting	The FortiGate internal clock may be set through NTP.
Online Help and Documentation	The online help and documentation supplements the external administrative and user documents.
Proxies	The FortiGate supports FTP, HTTP/HTTPS, IMAP, POP3, SMTP, and Telnet proxies for firewall users. Firewall rules may be defined that are applicable only to users who have authenticated to the firewall to use one of these proxies. The evaluated configuration only supports user authentication for FTP, HTTP, and Telnet.
Replacement Messages	The Security Administrator may configure replacement messages to customize alert email and information that the FortiGate unit adds to content streams such as email messages, web pages and FTP sessions. The FortiGate unit adds replacement messages to a variety of content streams. For example, if a virus is found in an email message attachment, the attached file is removed from the email and replaced with a replacement message. The same process applies to pages blocked by web-filtering and email blocked by spam filtering.
SMTP Server	The Simple Mail Transfer Protocol (SMTP) is used to send alert emails from the FortiGate.
SNMP	The FortiGate unit is able to transfer status information to a Simple Network Management Protocol (SNMP) Manager.
Spam Filter (Email Filtering)	Email filtering can be configured to scan all IMAP and POP3 email content for unwanted senders or for unwanted content.

Feature Excluded	Description
Support to Flaw Remediation	The FortiGate unit provides a means of sending bug reports to Fortinet in aid of flaw remediation.
Traffic Shaping	The FortiGate unit can be configured to restrict traffic based on bandwidth and time. Traffic Shaping controls the bandwidth available to and sets the priority of the traffic. The FortiGate can provide a guaranteed bandwidth, maximum bandwidth, and traffic priorities.
Troubleshooting Support	The FortiGate unit provides a capability of sending troubleshooting data directly to Fortinet.
USB Disk Support	The FortiGate-500A provides support for a Universal Serial Bus (USB) disk on which firmware and configuration data may be stored.
Virtual domain	FortiGate virtual domains provide multiple logical firewalls in a single FortiGate unit, so that one FortiGate unit can provide exclusive firewall and services to multiple networks. Traffic from each network is effectively separated from every other network.
VLAN	The FortiGate supports Virtual Local Area Network (VLAN) as a sub interface attached to a physical interface port.
Web Content Filtering	Web content filtering can be configured to scan and block all HTTP content protocol streams for Uniform Resource Locators (URLs) or for web page content. If a match is found between a URL on the URL block list, or if a web page is found to contain a word or phrase in the content block list, the FortiGate blocks the web page. The blocked web page is replaced with a message that an administrator can edit using the web-based manager.
Zone	The FortiGate supports the use of a zone as a shorthand notation to form a group of related interfaces and VLAN sub interfaces.

Table 5 - Features Excluded from the TOE

2.2 SECURITY FUNCTIONAL POLICIES

This Security Target references four information flow control Security Functional Policies (SFPs):

- the UNAUTHENTICATED INFORMATION FLOW SFP;
- the AUTHENTICATED INFORMATION FLOW SFP;
- the UNAUTHENTICATED TOE SERVICES SFP; and
- the VPN SFP.

For the UNAUTHENTICATED INFORMATION FLOW SFP, the subjects under control of this policy are the TOE interfaces that connect to unauthenticated users on an internal or external network sending information through the TOE to other destinations on the internal or external network. The information flowing between subjects in the policy is traffic with attributes, defined in FDP_IFF.1.1(1), including source and destination addresses. The rules that define the SFP are found in FDP_IFF.1.2(1). FMT_MSA.3(1) requires that these rules be assigned restrictive initial values. FMT_MSA.1 ensures that the rules are subsequently managed only by the Security Administrator.

For the AUTHENTICATED INFORMATION FLOW SFP, the subjects under control of this policy are the TOE interfaces that connect to authenticated users on an internal or external network sending information through the TOE to other destinations on the internal or external network. The information flowing between subjects in the policy is traffic with attributes, defined in FDP_IFF.1.1(2), including source and destination addresses. The rules that define the SFP are found in FDP_IFF.1.2(2). FMT_MSA.3(1) requires that these rules be assigned restrictive initial values. FMT_MSA.1 ensures that the rules are subsequently managed only by the Security Administrator.

For the UNAUTHENTICATED TOE SERVICES SFP, the subjects under control of this policy are the TOE interfaces that connect unauthenticated users on an internal or external network sending information to or receiving information from the TOE. The information flowing between subjects in the policy is traffic with attributes, defined in FDP_IFF.1.1(3), including source and destination addresses. The rules that define the SFP are found in FDP_IFF.1.2(3). FMT_MSA.3(2) requires that these rules be assigned restrictive initial values. FMT_MSA.1 and FMT_MOF.1(4) ensure that the rules are subsequently managed only by the Security Administrator.

For the VPN SFP, the subjects under control of this policy are the TOE interfaces that connect authenticated VPN Remote Devices on an internal or external network sending information to or receiving information from the TOE. The information flowing between subjects in the policy is traffic with attributes, defined in FDP_IFF.1.1(4), including source and destination addresses. The rules that define the SFP are found in FDP_IFF.1.2(4). FMT_MSA.3(1) requires that these rules be assigned restrictive initial values. FMT_MSA.1 ensures that the rules are subsequently managed only by the Security Administrator.

2.3 TOE DATA

2.3.1 TSF Data

The TOE retains TSF Configuration Data, consisting of:

- Potential Violation Analysis Ruleset;
- Cryptographic Data;
- Alarm Configuration;
- Audit Configuration;
- Identification and Authentication Data (User Attributes);
- Role/Permission Data;
- Time Data;
- Self-Test Parameters;
- Information Flow Policy Ruleset, including Protection Profiles;

- TOE Services Configuration;
- TSF Data Limits On Transport-Layer Resources And Actions If Exceeded;
- TSF Data Limits On Connection-Oriented Resources And Actions If Exceeded;
- TOE Access Banners;
- Trusted Channel Definition Parameters; and
- Trusted Path Definition Parameters.

The TOE retains TSF Operational Data, consisting of:

- Audit Records;
- Alarm Data;
- Session Data;
- Trusted Channel Usage;
- Trusted Paths Usage;
- Transport-Layer Resource Usage; and
- Connection-Oriented Resource Usage.

2.3.2 User Data

The TOE mediates the following User Data, based on a defined information flow policy:

- Information Flows to/from the TOE.

The TOE responds to the following User Data, based on a defined TOE services policy:

- TOE Service Request.

2.3.3 Security Attributes

The following security attributes are defined:

- Unauthenticated Policy Attributes;
- Authenticated Policy Attributes;
- Unauthenticated TOE Services Policy Attributes; and
- VPN Policy Attributes.

2.4 SUMMARY OF TOE SECURITY FUNCTIONS

2.4.1 Identification and Authentication

All administration requires authentication by a UNIX style user identification (ID) and password mechanism. Administration may either be performed locally using the Local Console CLI or remotely using the Network Web-Based GUI or Network CLI. TOE users are required to authenticate in order to use some TOE services. Remote authentication data is protected via encryption (trusted path).

2.4.2 Administration

The TOE provides remote and local administrative interfaces that permit the administrative roles to configure and manage the TOE. In each of the two evaluated configurations (i.e., the Single-Unit Configuration, as shown in Figure 1 and a High-Availability Configuration, as shown in Figure 2), the TOE is connected to two or more networks and remote administration data flows from a Network Management Station to the TOE. In each configuration there is also a Local Console, located within a Secure Area, with an interface to the TOE.

The TOE provides three separate administrative roles: Cryptographic Administrator, Audit Administrator and Security Administrator. A user assigned to the Cryptographic Administrator role is responsible for the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the TOE. A user assigned to the Audit Administrator role is the only user permitted to delete audit data. A user assigned to the Security Administrator role is responsible for all other administrative tasks (e.g., creating the TOE security policy) not addressed by the other two administrative roles.

In this Security Target the terms Cryptographic Administrator, Audit Administrator and Security Administrator refer to an administrative user assigned to that role. For instance, Audit Administrator is an administrative user who has been assigned the audit administrator role. The terms Administrator and Administrators refer to administrative users that have been assigned one of the Administrator roles.

2.4.3 Information Flow Control

The TOE provides interfaces to a defined set of networks and mediates information flow among these networks. The two evaluated configurations are the Single-Unit Configuration, as shown in Figure 1 and a High-Availability Configuration, as shown in Figure 2. In both of these configurations the TOE is connected to two or more networks and user data flows from a connected network, through the TOE, to a connected network.

Section 5.1 ‘TOE Functional Security Requirements’ defines the minimum set of configurable security attributes required to permit or deny information flows to or through the TOE. The set of security attributes includes items such as source and destination identification, service identifiers, and user authentication. The TOE Security Administrator configures the security attributes to construct one or more access control rules as part of a security policy on the TOE. The TOE implementation consists of one or more ‘rulesets’ that are subsequently applied to one or more TOE interfaces. Packets arriving at the TOE interface are compared to the security attributes in the ‘rulesets’. When the packet attributes ‘match’ the rules security attributes, that packet or connection is approved. In addition to restricting access via the rules, the TOE must generate and maintain ‘state’ information for all approved connections mediated by the TOE. The TOE utilizes the ‘state’ information to monitor the status of an approved connection and validate incoming packets purporting to be part of an approved connection. The FDP_IFF.1.3 requirement defines the minimum sets of ‘state’ attributes required by the TOE. Additional TOE requirements such as controls on half-open connections are included to assist the Security Administrator with managing the resources utilized by maintaining ‘state’ information. The TOE is required to perform a complete reassembly of all packet fragments prior to making an access control decision on the packet.

As mentioned at the beginning of the previous paragraph, the ST defines a minimal set of security attributes required to permit information flows to or through the TOE. The same security attributes discussed above apply to controlling access to services residing on the TOE. This ST includes Internet Control Message Protocol (ICMP) as a required unauthenticated information flow to the TOE. The TOE provides the Security Administrator with the capability of enabling or disabling ICMP data to or from the TOE. When ICMP is enabled, the security attributes defined in the FDP_IFF.1.3(3) requirement, including control of the ICMP message types are available to the Security Administrator. An additional requirement for unauthenticated SMTP information flow is identified in FIA_UAU.1(2) and also meets the requirements in FDP_IFF.1.1(1). The TOE also supports authenticated information flows and the authentication requirement is identified in FIA_UAU.2. Remote administration is also supported by the TOE.

2.4.4 Trusted Channel/Path

The TOE provides encrypted communications. Trusted path refers to the encrypted connection used to authenticate an external human user with the TOE. Trusted channel refers to the encrypted connection between the TOE and an external trusted IT entity.

A trusted path communication is required for the authentication of remote administrators and users of TOE services that require authentication. A remote administrator’s communication remains encrypted throughout the remote session.

The TOE requires an encrypted trusted channel for communication with Fortinet's FortiGuard Distribution Server.

2.4.5 Encryption

Section 5.1.2 ‘Cryptographic Support’ defines the minimum set of cryptographic attributes required by the TOE. The TOE’s cryptographic module(s) are FIPS PUB 140-2 validated and meet Security Level 2 overall and Security Level 3 for the following: cryptographic module ports and interfaces, roles, services and authentication , cryptographic key management and design assurance. The TOE generates and distributes symmetric and asymmetric keys. The implementation selections for key generation and key distribution are provided in Section 5.1.2. The TOE performs data encryption/decryption using the Advanced Encryption Standard (AES) algorithm with a minimum key size of 128 bits. Additional requirements for key destruction, digital signature generation/verification, random number generation and cryptographic hashing are provided in Section 5.1.2.

2.4.6 Audit

Section 5.1.1 ‘Security Audit (FAU)’ describes the TOE’s generation of audit records, alarms and audit management. Table 7 in the FAU_GEN.1 requirement lists the set of auditable events. FAU_SEL.1 identifies the attributes that are available to the Security Administrator for configuring the events that are audited on the TOE. Each auditable event generates an audit record. Table 7 also provides a list of attributes that are included in each audit record.

In addition to generating audit records, the TOE monitors auditible events and provides a Security Administrator configurable threshold for determining a potential security violation. Once the TOE has detected a potential security violation, an alarm message is displayed at the TOE’s local console as well as each active remote administrative session. The alarm message is also displayed at any remote administrative sessions which become active before the alarm is acknowledged. The message contains the potential security violation and all audit records associated with the potential security violation. The message will be displayed at the various consoles until administrator acknowledgement of the message has occurred. Additionally, the Security Administrator can configure the TOE to generate an audible alarm to indicate a potential security violation.

As mentioned in the ‘Administration’ section above, the Audit Administrator’s role is restricted to viewing the contents of the audit records and the deletion of the audit trail. The TOE provides the Audit Administrator with a sorting and searching capability to improve audit analysis. The Security Administrator configures auditable events, backs-up audit data and manages audit data storage. The TOE provides the Security Administrator with a configurable audit trail threshold to track the storage capacity of the audit trail. As soon as the threshold is met, the TOE displays a message in the same fashion as for potential security violations, including the option of the audible alarm. If log rolling is not enabled, when the TOE reaches the audit storage capacity threshold, the TOE will enter its FIPS-CC Error Mode which prevents all auditable events except for those events resulting from actions taken by the Security and Audit Administrators to correct the audit storage problem. If log rolling is enabled and the audit log becomes full, the TOE will overwrite the oldest audit records in the audit trail.

2.4.7 Self-Protection

The TOE provides self-protection functionality to ensure continued correct operation. Self-test functions are provided to detect problems in operation and respond to problems in a defined, repeatable manner. Failure of any self-test causes the TOE to enter its FIPS-CC Error Mode. Administrator intervention is then required to return the TOE to normal operations. Additionally, the TOE protects itself by rejecting replay of communications, avoiding overload of its interfaces, managing sessions, and restricting information released on banners.

3 TOE SECURITY ENVIRONMENT

3.1 ASSUMPTIONS

The specific conditions below are assumed to exist in the TOE environment.

A.ACCESS	The TOE has access to all the IT System data it needs to perform its functions.
A.LOCATE	The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.
A.MANAGE	There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.
A.NO_TOE_BYPASS[11]	Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.
A.NOEVIL	The authorized administrators are not careless, wilfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.
A.NOTRST	The TOE can only be accessed by authorized users.
A.PHYSICAL[10]	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.PROTCT	The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.

[11] These assumptions were drawn from the FW PP MR, the TFFW PP MR and the VPN PP MR and are in addition to the assumptions drawn from the IDSS PP.

3.2 THREATS

3.2.1 Threats Addressed by the TOE

The threats discussed below are addressed by the TOE. The threat agents are either unauthorized persons or external IT entities not authorized to use the TOE itself. The threat agents are assumed to have a low attack potential and are assumed to have a moderate level of resources and access to all publicly available information about the TOE and potential methods of attacking the TOE. It is expected that the FortiGate units will be protected to the extent necessary to ensure that they remain connected to the networks they protect. The following threats are addressed by the TOE and should be read in conjunction with Section 8.1.2 TOE Security Objectives Rationale.

T.ADDRESS_MASQUERADE[12]	A user on one interface may masquerade as a user on another interface to circumvent the TOE policy.
T.ADMIN_ERROR[12]	An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.
T.ADMIN_ROGUE[12]	An administrator's intentions may become malicious resulting in user or TOE Security Function (TSF) data being compromised.
T.AUDIT_COMPROMISE[12]	A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.
T.COMDIS	An unauthorized user may attempt to disclose the data collected by the TOE by bypassing a security mechanism.
T.COMINT	An unauthorized user may attempt to compromise the integrity of the data collected by the TOE by bypassing a security mechanism.
T.CRYPTO_COMPROMISE[12]	A malicious user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromise

[12] These threats are drawn from the FW PP MR, the TFFW PP MR and the VPN PP MR and are in addition to those provided in the IDSS PP.

	the cryptographic mechanisms and the data protected by those mechanisms.
T.FLAWED_DESIGN[12]	Unintentional or intentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a malicious user or program.
T.FLAWED_IMPLEMENTATION[12]	Unintentional or intentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a malicious user or program.
T.IMPCON	The TOE may be susceptible to improper configuration by any user, causing potential intrusions to go undetected.
T.INADVE	Inadvertent activity and access may occur on an IT System which may result in the TOE being affected by unauthorised users[13].
T.INFLUX	An unauthorized user may cause malfunction of the TOE by creating an influx of data that the TOE cannot handle.
T.INTRUSION[14]	A malicious agent may attempt to attack the TOE or one of the systems connected to the TOE by passing information which is designed to damage or compromise the system which receives the information.
T.LOSSOF	An unauthorized user may attempt to remove or destroy data collected by the TOE.
T.MALICIOUS_TSF_ COMPROMISE[12]	A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).
T.MASQUERADE[12]	A user may masquerade as an authorized user or an

[13] The IDSS PP threat was modified in order to identify a threat agent and the asset being attacked.

[14] The T.INTRUSION and T.VIRUS threats are not listed in any of the PPs referenced in this ST. These threats were added in order to describe additional threats which are countered by the TOE.

	authorized IT entity to gain access to data or TOE resources.
T.MISACT	Malicious activity, such as introductions of Trojan horses and viruses, may occur on an IT System which may result in the TOE being affected by unauthorised users[12].
T.MISUSE	Unauthorized accesses and activity indicative of misuse may occur on an IT System which may result in the TOE being affected by unauthorised users[12].
T.NOHALT	An unauthorized user may attempt to compromise the continuity of the Sensor's collection functionality by halting execution of the TOE.
T.POOR_TEST[12]	Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities.
T.PRIVIL	An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.
T.REPLAY[12]	A user may gain inappropriate access to the TOE by replaying authentication information, or may cause the TOE to be inappropriately configured by replaying TSF data or security attributes (captured as it was transmitted during the course of legitimate use).
T.RESIDUAL_DATA[12]	A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.
T.RESOURCE_EXHAUSTION[12]	A malicious process or user may block others from TOE system resources (e.g., connection state tables) via a resource exhaustion denial of service attack.
T.SPOOFING[12]	An entity may mis-represent itself as the TOE to obtain authentication data.

T.UNATTENDED_SESSION[12]	A user may gain unauthorized access to an unattended session.
T.UNAUTHORIZED_ACCESS[12]	A user may gain access to services (either on the TOE or by sending data through the TOE) for which they are not authorized according to the TOE security policy.
T.UNAUTHORIZED_PEER[12]	An unauthorized IT entity may attempt to establish a security association with the TOE.
T.UNIDENTIFIED_ACTIONS[12]	The administrator may fail to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.
T.UNKNOWN_STATE[12]	When the TOE is initially started or restarted after a failure, design flaws, or improper configurations may cause the security state of the TOE to be unknown.
T.VIRUS[12]	A malicious agent may attempt to pass a virus through or to the TOE.

3.3 ORGANIZATIONAL SECURITY POLICIES

The TOE must address the organizational security policies described below.

P.ACCACT	Users of the TOE shall be accountable for their actions within the IDS.
P.ACCESS	All data collected by the TOE shall only be used for authorized purposes.
P.ACCESS_BANNER[15]	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.

[15] These policies are drawn from the FW PP MR, the TFFW PP MR and the VPN PP MR and are in addition to those provided drawn from the IDSS PP.

P.ACCOUNTABILITY[15]	The authorized users of the TOE shall be held accountable for their actions within the TOE.
P.ADMIN_ACCESS[15]	Administrators shall be able to administer the TOE both locally and remotely through protected communications channels.
P.CRYPTOGRAPHIC_FUNCTIONS[15]	The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.
P.CRYPTOGRAPHY_VALIDATED[15]	Where the TOE requires FIPS-approved security functions, only National Institute of Standards and Technology (NIST) FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key distribution, and random number generation services).
P.DETECT	All events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets must be collected.
P.INTEGRITY[15]	The TOE shall support the Internet Engineering Task Force (IETF) *Internet Protocol Security Encapsulating Security Payload* (IPSec ESP) as specified in RFC 2406. Sensitive information transmitted to a peer TOE shall apply integrity mechanisms as specified in *Use of HMAC-SHA-1-96 within ESP and AH* (RFC 2404).
P.INTGTY	Data collected by the TOE shall be protected from modification.
P.MANAGE	The TOE shall be manageable only by authorized users.

P.PROTCT	The TOE shall be protected from unauthorized accesses and disruptions of collection activities.
P.VULNERABILITY_ANALYSIS_TEST[15]	The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a low attack potential.

4 SECURITY OBJECTIVES

This chapter describes the security objectives for the TOE and the TOE's operating environment. The security objectives are divided between TOE Security Objectives (i.e., security objectives addressed directly by the TOE) and Security Objectives for the Operating Environment (i.e., security objectives addressed by the IT domain or by non-technical or procedural means). The mapping of security objectives to assumptions, threats and organizational security policies along with the rationale for this mapping is found in Section 8.

4.1 TOE SECURITY OBJECTIVES

This section defines the security objectives that are to be addressed by the TOE.

O.ACCESS	The TOE must allow authorized users to access only appropriate TOE functions and data.
O.ADMIN_ROLE[16]	The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.
O.AUDIT_GENERATION[16]	The TOE will provide the capability to detect and create records of security-relevant events associated with users.
O.AUDIT_PROTECTION[16]	The TOE will provide the capability to protect audit information.
O.AUDIT_REVIEW[16]	The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.
O.AUDITS	The TOE must record audit records for data accesses and use of the Sensor functions.
O.CHANGE_MANAGEMENT[16]	The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.

[16] These objectives are drawn from the FW PP MR, the TFFW PP MR and the VPN PP MR and are in addition to those drawn from the IDSS PP.

O.CORRECT_TSF_OPERATION[16]	The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.
O.CRYPTOGRAPHIC_ FUNCTIONS[16]	The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.
O.CRYPTOGRAPHY_ VALIDATED[16]	The TOE shall use NIST FIPS 140-2 validated cryptomodules for cryptographic services implementing FIPS-approved security functions and random number generation services used by cryptographic functions.
O.DISPLAY_BANNER[16]	The TOE will display an advisory warning regarding use of the TOE.
O.EADMIN	The TOE must include a set of functions that allow effective management of its functions and data.
O.EXPORT	When the TOE makes its Sensor data available to other IDS components, the TOE will ensure the confidentiality of the Sensor data.
O.IDACTS	The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.
O.IDAUTH	The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data.
O.INTEGR	The TOE must ensure the integrity of all audit and Sensor data.

O.INTEGRITY[16]	The TOE must be able to protect the integrity of data transmitted to a peer TOE via encryption and provide IPSec authentication for such data. Upon receipt of data from a peer TOE, the TOE must be able to decrypt the data and verify that the received data accurately represents the data that was originally transmitted.
O.INTRUSION[17]	The TOE will detect and prevent intrusion attacks which are contained within an information flow which arrives at any of the TOE network interfaces.
O.MAINT_MODE[16]	The TOE shall provide a mode from which recovery or initial startup procedures can be performed.
O.MANAGE[16]	The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.
O.MEDIATE[16]	The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.
O.OFLOWS	The TOE must appropriately handle potential audit and Sensor data storage overflows.
O.PEER_AUTHENTICATION[16]	The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.
O.PROTCT	The TOE must protect itself from unauthorized modifications and access to its functions and data.
O.REPLAY_DETECTION[16]	The TOE will provide a means to detect and reject the replay of TSF data and security attributes.

[17] The O.INTRUSION, O.SECURE_UPDATES AND O.VIRUS are not listed in any of the PPs referenced by this ST. They were added to the ST in order to describe capabilities of the TOE which are beyond those required by the referenced PPs.

O.RESIDUAL_INFORMATION[16]	The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.
O.RESOURCE_SHARING[16]	The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; Transmission Control Protocol (TCP) connections to the TOE).
O.ROBUST_ADMIN_GUIDANCE[16]	The TOE will provide administrators with the necessary information for secure delivery and management.
O.ROBUST_TOE_ACCESS[16]	The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate
O.SECURE_UPDATES[17]	The TOE shall provide a secure mechanism for the receipt of virus and intrusion signature updates.
O.SELF_PROTECTION[16]	The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure. The TSF will provide a High Availability configuration which allows for continued operation of the TOE in the event of a single unit failure.[18]
O.SOUND_DESIGN[16]	The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.
O.SOUND_IMPLEMENTATION[16]	The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.
O.THOROUGH_FUNCTIONAL_ TESTING[16]	The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.

[18] The text of the O.SELF_PROTECTION objective was modified to include the HA capabilities of the TOE.

O.TIME_STAMPS[16]	The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.
O.TRUSTED_PATH[16]	The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.
O.VIRUS[17]	The TOE will detect and block viruses contained within an information flow which arrives at any of the TOE network interfaces.
O.VULNERABILITY_ANALYSIS_TEST[16]	The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.

4.2 SECURITY OBJECTIVES FOR THE OPERATING ENVIRONMENT

This section defines the security objectives that are to be addressed by the IT domain or by non-technical or procedural means.

OE.CREDEN	Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.
OE.CRYPTANALYTIC[16]	Cryptographic methods used in the IT environment shall be interoperable with the TOE, should be FIPS 140-2 validated and should be resistant to cryptanalytic attacks (i.e., will be of adequate strength to protect unclassified Mission Support, Administrative, or Mission Critical data).
OE.INSTAL	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.
OE.INTROP	The TOE is interoperable with the IT System it monitors and other IDS components within its IDS.
OE.NO_TOE_BYPASS[16]	Information cannot flow between external and internal

	networks located in different enclaves without passing through the TOE.
OE.PERSON	Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the Sensor.
OE.PHYCAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.
OE.PHYSICAL[16]	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.

5 IT SECURITY REQUIREMENTS

This section provides security functional and assurance requirements that must be satisfied by the TOE. These requirements consist of components from the CC Part 2 and Part 3, National Information Assurance Partnership (NIAP) interpreted requirements, and explicit requirements.

5.1 TOE SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for the TOE are summarized in Table 6. These requirements consist of components derived from the IDSS PP, the FW PP MR, the TFFW PP MR, the VPN PP MR and Part 2 of the CC. The source of each component is identified in the table. Requirements which have been refined in this document are shown in Table 6 using bold text. Readers should note that in many cases the three MR PPs include requirements which are NIAP refinements of standard CC requirements. In cases where the NIAP refinements have been incorporated into Version 2.3 of the CC, this ST has used the CC requirements. These cases are noted in Table 6 using footnotes.

Component	**Description**	**Source**
FAU_ARP.1	**Security alarms**	FW PP MR, TFFW PP MR, VPN PP MR
FAU_ARP_ACK_EXP.1	**Security alarm acknowledgement**	FW PP MR, TFFW PP MR, VPN PP MR
FAU_GEN.1[19]	**Audit data generation**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAU_GEN.2[20]	**User identity association**	FW PP MR, TFFW PP MR, VPN PP MR
FAU_SAA.1[21]	**Potential violation analysis**	FW PP MR, TFFW PP MR, VPN PP MR
FAU_SAR.1	**Audit review**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP

[19] FAU_GEN.1-NIAP-0410 in MR PPs.

[20] FAU_GEN.2-NIAP-0410 in MR PPs.

[21] FAU_SAA.1-NIAP-0407 in MR PPs.

Component	Description	Source
		MR
FAU_SAR.2	**Restricted audit review**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAU_SAR.3	**Selectable audit review**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAU_SEL.1[22]	**Selective audit**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAU_STG.2[23]	**Guarantees of audit data availability**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAU_STG.3	**Action in case of possible audit data loss**	FW PP MR, TFFW PP MR, VPN PP MR
FAU_STG.4[24]	**Prevention of audit data loss**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FAV_ACT_EXP.1	Anit Virus Actions	Explicit requirement added to specify Anti Virus capabilities of the TOE
FCS_BCM_EXP.1	**Baseline cryptographic module**	FW PP MR, TFFW PP MR, VPN PP MR
FCS_CKM.1(1)[25]	**Cryptographic Key Management (key generation)**	FW PP MR, TFFW PP MR (ST contains additional requirements)

[22] FAU_SEL.1-NIAP-0407 in MR PPs.

[23] FAU_STG.1-NIAP-0423 Protected Audit Trail Storage in the MR PPs.

[24] FAU_STG.NIAP-0414-1-NIAP-0429 Site-Configurable Prevention of Audit Loss in MR PPs.

[25] FCS_CKM.1 in MR PPs.

Component	**Description**	**Source**
FCS_CKM.1(2)[26]	Cryptographic Key Management (Key Establishment for symmetric keys)	FW PP MR, TFFW PP MR (ST contains additional requirements)
FCS_CKM.1(3)[27]	Cryptographic Key Management (Key Entry for Digital Signature/Verification Private Keys)	FW PP MR, TFFW PP MR
FCS_CKM.1(4)[28]	Cryptographic Key Management (Key Validation and Packaging)	VPN PP MR
FCS_CKM.1(5)[29]	Cryptographic Key Management (Internet Key Exchange)	VPN PP MR
FCS_CKM.2(1)[30]	Cryptographic Key Management (Key Handling and Storage)	VPN PP MR
FCS_CKM.2(2)	Cryptographic Key Management (Key Distribution)	VPN PP MR
FCS_CKM.4	**Cryptographic key destruction**	FW PP MR, TFFW PP MR, VPN PP MR
FCS_COP.1(1)[31]	Cryptographic operation (Encryption/Decryption AES)	FW PP MR, TFFW PP MR
FCS_COP.1(2)[32]	Cryptographic operation (Digital Signature Generation/Verification	FW PP MR, TFFW PP MR

[26] FCS_CKM_SYM_EXP.1 in MR PPs.

[27] FCS_CKM_ASYM_EXP.1 in MR PPs.

[28] FCS_CKM_(EXP).1 Cryptographic Key Validation and Packaging in VPN PP MR.

[29] FCS_IKE_(EXP).1 Internet Key Exchange in VPN PP MR.

[30] FCS_CKM_(EXP).2 Cryptographic Key Handling and Storage in VPN PP MR.

[31] FCS_COP_EXP.2 in FW PP MR and TFFW PP MR.

[32] FCS_COP_EXP.3 in FW PP MR and TFFW PP MR.

Component	**Description**	**Source**
FCS_COP.1(3)[33]	Cryptographic operation (Cryptographic Hash function)	FW PP MR, TFFW PP MR
FCS_COP.1(4)[34]	Cryptographic operation (Random number generation)	FW PP MR, TFFW PP MR
FDP_IFC.1(1)	**Subset information flow control (unauthenticated policy)**	FW PP MR, TFFW PP MR
FDP_IFC.1(2)	**Subset information flow control (authenticated policy)**	FW PP MR
FDP_IFC.1(3)[35]	Subset information flow control (unauthenticated TOE services policy)	FW PP MR, TFFW PP MR, VPN PP MR
FDP_IFC.1(4)[36]	**Subset information flow control (VPN policy)**	VPN PP MR
FDP_IFF.1(1)[37]	**Simple security attributes (unauthenticated policy)**	FW PP MR, TFFW PP MR
FDP_IFF.1(2)[38]	**Simple security attributes (authenticated policy)**	FW PP MR
FDP_IFF.1(3)[39]	**Simple security attributes (unauthenticated TOE services policy)**	FW PP MR, TFFW PP MR, VPN PP MR

[33] FCS_COP_EXP.6 in FW PP MR and TFFW PP MR.

[34] FCS_COP_EXP.5 in FW PP MR and TFFW PP MR.

[35] FDP_IFC.1(2) in TFFW PP MR and VPN PP MR.

[36] FDP_IFC.1(1) in VPN PP MR.

[37] FDP_IFF.1-NIAP-0417(1) in FW PP MR and TFFW PP MR.

[38] FDP_IFF.1-NIAP-0417(2) in FW PP MR.

[39] FDP_IFF.1-NIAP-0417(3) in FW PP MR and FDP_IFF.1-NIAP-0417(2) in TFFW PP MR and VPN PP MR.

Component	Description	Source
FDP_IFF.1(4)[40]	**Simple security attributes (VPN Policy)**	VPN PP MR
FDP_RIP.2	Full residual information protection	FW PP MR, TFFW PP MR, VPN PP MR
FIA_AFL.1[41]	**Authentication failure handling**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FIA_ATD.1(1)[42]	**User attribute definition (administrators)**	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FIA_ATD.1(2)	**User attribute definition (authorized proxy user)**	CC Part 2
FIA_ATD.1(3)	**User attribute definition (VPN Remote Devices)**	CC Part 2
FIA_UAU.1(1)	Timing of authentication (for TOE services)	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FIA_UAU.1(2)	Timing of authentication (for information flow through TOE)	FW PP MR
FIA_UAU.2[43]	**User authentication before any action**	FW PP MR, TFFW PP MR, VPN PP MR
FIA_UAU.5[44]	**Multiple authentication mechanisms**	FW PP MR, TFFW PP MR, VPN PP MR

[40] FDP_IFF.1-NIAP-0417(1) in VPN PP MR.

[41] FIA_AFL.1-NIAP-0425 in MR PPs.

[42] In the MR PPs, FIA_ATD.1 was not iterated. Iterations (1), (2) and (3) have been used to ensure that the attribute requirements for the users required by the MR PPs were clear.

[43] FIA_UAU_EXP.2 in MR PPs.

[44] FIA_UAU_EXP.5 in MR PPs.

Component	Description	Source
FIA_UID.2[45]	User identification before any action	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FIA_USB.1	User-Subject Binding	FW PP MR, TFFW PP MR, VPN PP MR
FIP_ACT_EXP.1	Intrusion Prevention Actions	Explicit requirement added to specify intrusion prevention capabilities of the TOE
FMT_MOF.1(1)	Management of security functions behavior (TSF non-cryptographic self-test)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(2)	Management of security functions behavior (cryptographic self-test)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(3)	Management of security functions behavior (audit and alarms)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(4)	Management of security functions behavior (audit and alarms)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(5)	Management of security functions behavior (audit and alarms)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(6)	Management of security functions behavior (available TOE-services for unauthenticated users)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(7)	Management of security functions behavior (quota mechanism)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MOF.1(8)	Management of security functions behavior (cryptographic self-test frequency)	CC Part 2

[45] The IDSS PP requires FIA_UID.1. The ST is claiming FIA_UID.2 which is hierarchal to FIA_UID.1.

Component	Description	Source
FMT_MOF.1(9)	Management of security functions behavior (audit storage exhaustion)	CC Part 2
FMT_MOF.1(10)	Management of security functions behavior (session termination)	CC Part 2
FMT_MOF.1(11)	Management of security functions behavior (alarm acknowledgement	CC Part 2
FMT_MOF.1(12)	Management of security functions behavior (self-tests)	CC Part 2
FMT_MOF.1(13)	**Management of security functions behavior (IDS sensor)**	IDSS PP
FMT_MSA.1	Management of security attributes	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MSA.2	Secure security attributes	CC Part 2
FMT_MSA.3(1)[46]	**Static attribute initialization (ruleset)**	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MSA.3(2)[47]	**Static attribute initialization (services)**	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MTD.1(1)	Management of TSF data (audit data)	CC Part 2
FMT_MTD.1(2)	Management of TSF data (cryptographic TSF data)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MTD.1(3)	Management of TSF data (time TSF data)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MTD.1(4)	Management of TSF data (information flow policy ruleset)	FW PP MR, TFFW PP MR

[46] FMT_MSA.3-NIAP-0409(1) in MR PPs.

[47] FMT_MSA.3-NIAP-0409(2) in MR PPs.

Component	**Description**	**Source**
FMT_MTD.1(5)	Management of TSF data (user accounts)	CC Part 2
FMT_MTD.1(6)	Management of TSF data (TOE banner)	CC Part 2
FMT_MTD.1(7)	Management of TSF data (AV and IPS signatures)	CC Part 2
FMT_MTD.1(8)	Management of TSF data (VPN policy ruleset)	VPN PP MR
FMT_MTD.1(9)	**Management of TSF data (IDS sensor data)**	IDSS PP
FMT_MTD.2(1)	Management of limits on TSF data (transport-layer quotas)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_MTD.2(2)	Management of limits on TSF data (controlled connection-oriented quotas)	FW PP MR, TFFW PP MR, VPN PP MR
FMT_REV.1	**Revocation**	FW PP MR, TFFW PP MR, VPN PP MR
FMT_SMR.2	**Restrictions on security roles**	IDSS PP[48], FW PP MR, TFFW PP MR, VPN PP MR
FPT_AMT.1	**Abstract Machine Testing**	CC Part 2
FPT_FLS.1	Failure with preservation of secure state	CC Part 2
FPT_ITA.1	Inter-TSF availability within a defined availability metric	IDSS PP
FPT_ITC.1	Inter-TSF confidentiality during transmission	IDSS PP

[48] The IDSS PP requires FMT_SMR.1. The ST is claiming FMT_SMR.2 which is hierarchal to FMT_SMR.1.

Component	**Description**	**Source**
FPT_ITI.1	**Inter-TSF detection of modification**	IDSS PP
FPT_RCV.1	Manual Recovery	FW PP MR, TFFW PP MR, VPN PP MR
FPT_RPL.1	Replay detection	FW PP MR, TFFW PP MR, VPN PP MR
FPT_RVM.1	Non-bypassability of the TSP	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FPT_SEP.2	**SFP domain separation**	IDSS PP[49], FW PP MR, TFFW PP MR, VPN PP MR
FPT_STM.1	Reliable time stamps	IDSS PP, FW PP MR, TFFW PP MR, VPN PP MR
FPT_TST.1(1)[50]	**TSF testing (with cryptographic integrity verification)**	FW PP MR, TFFW PP MR, VPN PP MR
FPT_TST.1(2)[51]	**TSF Testing (Cryptographic self-test)**	FW PP MR, TFFW PP MR, VPN PP MR
FRU_FLT.1	Degraded fault tolerance	CC Part 2
FRU_RSA.1(1)	**Maximum quotas (transport-layer quotas)**	FW PP MR, TFFW PP MR, VPN PP MR
FRU_RSA.1(2)	**Maximum quotas (controlled connection-oriented quotas)**	FW PP MR, TFFW PP MR, VPN PP MR
FTA_SSL.1	**TSF-initiated session locking**	FW PP MR, TFFW PP MR,

[49] The IDSS PP requires FPT_SEP.1. The ST is claiming FPT_SEP.2 which is hierarchal to FPT_SEP.1.

[50] FPT_TST_EXP.4 in MR PPs.

[51] FPT_TST_EXP.5 in MR PPs.

Component	Description	Source
		VPN PP MR
FTA_SSL.2	**User-initiated locking**	FW PP MR, TFFW PP MR, VPN PP MR
FTA_SSL.3	**TSF-initiated termination**	FW PP MR, TFFW PP MR, VPN PP MR
FTA_TAB.1	**Default TOE access banners**	FW PP MR, TFFW PP MR, VPN PP MR
FTA_TSE.1	**TOE session establishment**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_ITC.1(1)	**Inter-TSF trusted channel (Prevention of Disclosure)**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_ITC.1(2)	**Inter-TSF trusted channel (Detection of Modification)**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_TRP.1(1)	**Trusted path (Prevention of Disclosure)**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_TRP.1(2)	**Trusted path (Detection of Modification)**	FW PP MR, TFFW PP MR, VPN PP MR
IDS_COL_EXP.1[52]	**Sensor data collection**	IDSS PP
IDS_RDR_EXP.1[52]	Restricted data review	IDSS PP
IDS_STG_EXP.1[52]	Guarantee of sensor data availability	IDSS PP
IDS_STG_EXP.2[52]	**Prevention of sensor data loss**	IDSS PP
FTP_ITC.1(3) (ENV)	**Inter-TSF trusted channel (Prevention of Disclosure)**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_ITC.1(4)	**Inter-TSF trusted channel**	FW PP MR, TFFW PP MR,

[52] '_EXP' was added to the label in order to make it clear that the requirement was explicit.

Component	Description	Source
(ENV)	**(Detection of Modification)**	VPN PP MR
FTP_TRP.1(3) (ENV)	**Trusted path (Prevention of Disclosure)**	FW PP MR, TFFW PP MR, VPN PP MR
FTP_TRP.1(4) (ENV)	**Trusted path (Detection of Modification)**	FW PP MR, TFFW PP MR, VPN PP MR

Table 6 - Security Functional Requirements

5.1.1 Security Audit (FAU)

FAU_ARP.1 Security alarms

FAU_ARP.1.1 - **Refinement:** The TSF shall [immediately display an alarm message identifying the potential security violation, at the option of the Security Administrator generate an audible alarm, and make accessible the audit record contents associated with the auditable event(s) that generated the alarm, at the:

- Local Console,
- Network Web-Based GUI, and Network CLI sessions that exist, and;
- Local Console, Network Web-Based GUI, and Network CLI sessions that are initiated before the alarm has been acknowledged;]

upon detection of a potential security violation.

Application Note: The TOE displays the alarm message (and sounds the audible alarm if so configured) at the Local Console, regardless of whether or not one of the Administrators is logged in at the Local Console.

FAU_ARP_ACK_EXP.1 - Security alarm acknowledgement

FAU_ARP_ACK_EXP.1.1 – The TSF shall display the alarm message identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) until it has been acknowledged. An audible alarm will sound until acknowledged by an administrator.

FAU_ARP_ACK_EXP.1.2 – **Refinement:** The TSF shall display an acknowledgement message identifying a reference to the potential security violation, a notice that it has been acknowledged, the time of the acknowledgement and the user identifier that acknowledged the alarm, at the:

- Local Console, and

- Network Web-Based GUI, and Network CLI sessions that received the alarm if they still exist.

Application Note: The TOE displays the acknowledgement message at the Local Console, regardless or whether or not one of the Administrators is logged in at the Local Console.

FAU_GEN.1 Audit data generation

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

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

b) All auditable events for the [basic] level of audit; and

c) [*All auditable events* ***listed in Table 7****, which is a complete list, including those required by the basic level of audit and the IDS-specific events required by the IDSS PP.*].

FAU_GEN.1.2 - **Refinement:** 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 ST, [*information specified in column three of Table 7 below*].

Requirement	Auditable Events	Additional Audit Record Contents
FAU_ARP.1	Potential security violation was detected	Identification of what caused the generation of the alarm
FAU_ARP_ACK_EXP.1	Alarm acknowledgement Application Note: May be combined with the auditable event **record** for FAU_ARP.1.	The identity of the Administrator that acknowledged the alarm.
FAU_GEN.1	Start-up and shutdown of audit	

Requirement	Auditable Events	Additional Audit Record Contents
	functions	
FAU_GEN.1	Access to the sensor	
FAU_GEN.1	Access to the TOE Sensor data	Object IDS, Requested access
FAU_GEN.2	None	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms	The identity of the Security Administrator performing the function
FAU_SAR.1	Reading of information from the audit records (Opening the audit trail)	The identity of the Administrator performing the function
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	The identity of the Administrator attempting the function
FAU_SAR.3	None	
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	The identity of the Security Administrator performing the function
FAU_STG.2	None	
FAU_STG.3	Actions taken due to exceeding the audit threshold	The identity of the Security Administrator performing the function
FAU_STG.4	Actions taken due to the audit storage failure.	The identity of the Security Administrator performing the function
FAV_ACT_EXP.1	Action taken due to the detection of a virus	
FCS_BCM_EXP.1	None	

Requirement	Auditable Events	Additional Audit Record Contents
FCS_CKM.1(1)	Generation and loading of key. Failure of the activity	
FCS_CKM.1(2)	Failure of the activity	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_CKM.1(3)	Failure of the activity	
FCS_CKM.1(4)	None	
FCS_CKM.1(5)	Generation and loading of key pair for digital signatures. Changes to the pre-shared key used for authentication All modifications to the key lifetimes. Failure of the authentication in Phase 1. Failure to negotiate a security association in Phase 2.	If failure occurs, record an English description for the failure.
FCS_CKM.2(1)	None	
FCS_CKM.2(2)	None	
FCS_CKM.4	None	
FCS_COP.1(1)	Failure of cryptographic operation	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_COP.1(2)	Failure of cryptographic	Type of cryptographic operation

Requirement	Auditable Events	Additional Audit Record Contents
	operation	Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_COP.1(3)	Failure of cryptographic operation	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_COP.1(4)	Failure of cryptographic operation	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FDP_IFC.1(1)	None	
FDP_IFC.1(2)	None	
FDP_IFC.1(3)	None	
FDP_IFC.1(4)	None	
FDP_IFF.1(1)	Decisions to permit/deny information flows Failure to reassemble fragmented packets	Presumed identity of source subject Identity of destination subject Transport layer protocol, if applicable Source subject service identifier, if applicable Destination subject service identifier, if applicable Identity of the firewall interface on which the TOE received the packet Identity of the rule that allowed or

Requirement	Auditable Events	Additional Audit Record Contents
		disallowed the packet flow Reason why fragmented packets could not be reassembled (i.e., invalid fragment identifier, invalid offset, invalid fragment data length)
FDP_IFF.1(2)	Decisions to permit/deny information flows Failure to reassemble fragmented packets	Presumed identity of source subject Identity of destination subject Transport layer protocol, if applicable Source subject service identifier, if applicable Destination subject service identifier, if applicable Identity of the firewall interface on which the TOE received the packet Identity of the rule that allowed or disallowed the packet flow Reason why fragmented packets could not be reassembled (i.e., invalid fragment identifier, invalid offset, invalid fragment data length)
FDP_IFF.1(3)	Decisions to permit/deny information flows between a subject and the TOE	Presumed identity of source subject Identity of destination subject Transport layer protocol, if applicable

Requirement	Auditable Events	Additional Audit Record Contents
		Source subject service identifier, if applicable Destination subject service identifier, if applicable Identity of the firewall interface associated on which the TOE received the packet Identity of the rule that allowed or disallowed the packet flow, if applicable[53]
FDP_IFF.1(4)	Decisions to permit/deny information flows Operation applied to each information flow permitted	Presumed identity of source subject Identity of destination subject Transport layer protocol, if applicable Source subject service identifier, if applicable Destination subject service identifier, if applicable Identity of the firewall interface on which the TOE received the packet For denied information flows, the reason for denial.
FDP_RIP.2	None	

[53] The TOE may not use a rule in a ruleset to allow/disallow TOE services (e.g., configuration parameter could be used instead) and if this is the case, it is not required that a rule be identified.

Requirement	Auditable Events	Additional Audit Record Contents
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts The actions (e.g. disabling of an account) taken The subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of an account)	Identity of the unsuccessfully authenticated user Claimed identity of the unsuccessfully authenticated user and the identity of the Security Administrator performing the function.
FIA_ATD.1(1)	None	
FIA_ATD.1(2)	None	
FIA_ATD.1(3)	None	
FIA_UAU.1(1)	All use of the authentication mechanism	User identity, location
FIA_UAU.1(2)	All use of the authentication mechanism	User identity, location
FIA_UAU.2	All use of authentication mechanisms	Claimed identity of the user using the authentication mechanism
FIA_UAU.5	All use of the local authentication mechanism All use of other authentication mechanisms	Claimed identity of the user attempting to authenticate
FIA_UID.2	All use of the user identification mechanism used for authorized users (that is, those that authenticate to the TOE)	Claimed identity of the user using the identification mechanism, location
FIA_USB.1	Success and failure of binding of user security attributes to a subject	The identity of the user whose attributes are attempting to be bound

Requirement	Auditable Events	Additional Audit Record Contents
FIP_ACT_EXP.1	Action taken due to the detection of an intrusion attack	
FMT_MOF.1(1)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MOF.1(2)	Enabling or disabling of the key-generation self-tests	The identity of the Security Administrator performing the function
FMT_MOF.1(3)	All modifications in the behavior of the functions in the TSF	The identity of the Administrator performing the function
FMT_MOF.1(4)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MOF.1(5)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MOF.1(6)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MOF.1(7)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MOF.1(8)	All changes to the frequency of periodic execution of the cryptographic self-tests.	The identity of the Security Administrator performing the function.
FMT_MOF.1(9)	All changes of the action to be taken in the event of audit storage exhaustion	The identity of the Security Administrator who changed the TOE configuration.
FMT_MOF.1(10)	All changes to the period of inactivity which results in	The identity of the Security Administrator who changed the

Requirement	Auditable Events	Additional Audit Record Contents
	session termination	TOE configuration.
FMT_MOF.1(11)	All alarm acknowledgements	The identity of the Administrator who acknowledged the alarm.
FMT_MOF.1(12)	All on-demand execution of the self-tests.	The identity of the Administrator who invoked the self-tests.
FMT_MOF.1(13)	All modifications in the behavior of the functions in the TSF	The identity of the Security Administrator performing the function
FMT_MSA.1	All manipulation of the security attributes	The identity of the Security Administrator performing the function
FMT_MSA.2	All offered and rejected values for a security attribute.	
FMT_MSA.3(1)	None	
FMT_MSA.3(2)	None	
FMT_MTD.1(1)	All deletions of audit data	The identity of the Audit Administrator performing the function
FMT_MTD.1(2)	All key loading operations performed by the Cryptographic Administrator	The identity of the Cryptographic Administrator performing the function
FMT_MTD.1(3)	All modifications to the time and/or date used to form the time stamps by the Security Administrator	The identity of the Security Administrator performing the function
FMT_MTD.1(4)	All modifications to the information flow policy ruleset by the Security Administrator	The identity of the Security Administrator performing the function

Requirement	Auditable Events	Additional Audit Record Contents
FMT_MTD.1(5)	All creation and modifications of user accounts.	The identity of the Security Administrator who modified the account.
FMT_MTD.1(6)	All changes to the contents of the TOE banner.	The identity of the Security Administrator who modified the banner.
FMT_MTD.1(7)	All updates to the AV and IPS signatures	The identity of the Security Administrator or FortiGuard Distribution Server who performed the update.
FMT_MTD.1(8)	All modifications of the VPN Policy Rules	The identity of the Security Administrator performing the function
FMT_MTD.1(9)	None	
FMT_MTD.2(1)	All modifications of the limits Actions taken when the quota is exceeded (include the fact that the quota was exceeded)	The identity of the Security Administrator performing the function
FMT_MTD.2(2)	All modifications of the limits Actions taken when the quota is exceeded (include the fact that the quota was exceeded)	The identity of the Security Administrator performing the function
FMT_REV.1	All attempts to revoke security attributes	List of security attributes that were attempted to be revoked The identity of the Security Administrator performing the function
FMT_SMR.2	Modifications to the group of users that are part of a role Unsuccessful attempts to use a	User IDs that are associated with the modifications The identity of the Security

Requirement	Auditable Events	Additional Audit Record Contents
	role due to the given conditions on the roles.	Administrator performing the function
FPT_AMT.1	Execution of the tests of the underlying machine and the results of the tests.	
FPT_FLS.1	Failure of the TSF	
FPT_ITA.1	The absence of TSF data when required by a TOE.	
FPT_ITC.1	None	
FPT_ITI.1	a) The detection of modification of transmitted TSF data. b) The action taken upon detection of modification of transmitted TSF data.	
FPT_RCV.1	The fact that a failure or service discontinuity occurred Resumption of the regular operation	Type of failure or service discontinuity
FPT_RPL.1 (including replay of authentication data notification from the authentication server)	Notification that a replay event occurred	Identity of the user that was the subject of the reply attack
FPT_RVM.1	None	
FPT_SEP.2	None	
FPT_STM.1	Changes to the time	
FPT_TST.1(1)	Execution of this set of TSF self tests The results of the test	The identity of the Administrator performing the test, if initiated by an Administrator

Requirement	Auditable Events	Additional Audit Record Contents
FPT_TST.1(2)	Execution of this set of TSF self tests The results of the test	The identity of the Administrator performing the test, if initiated by an Administrator
FRU_FLT.1	All TOE capabilities being discontinued due to a failure.	
FRU_RSA.1(1)	None	
FRU_RSA.1(2)	None	
FTA_SSL.1	Locking of a Local Console interactive session by the session locking mechanism Any attempts at unlocking of a Local Console interactive session	The identity of the Administrator associated with the session being locked or unlocked
FTA_SSL.2	Locking of a Local Console interactive session by the session locking mechanism Any attempts at unlocking of a Local Console interactive session	The identity of the Administrator associated with the session being locked or unlocked
FTA_SSL.3	The termination of a Network Web-Based GUI, Network CLI, or authenticated proxy user, or VPN user session by the session locking mechanism	The identity of the User or Administrator associated with the session that was terminated
FTA_TAB.1	None	
FTA_TSE.1	All attempts at establishment of a user or Administrator session	The identity of the User or Administrator attempting to establish the session For unsuccessful attempts, the reason for denial of the

Requirement	Auditable Events	Additional Audit Record Contents
		establishment attempt
FTP_ITC.1(1)	All attempted uses of the trusted channel functions	Identification of the initiator and target of all trusted channels
FTP_ITC.1(2)	All attempted uses of the trusted channel functions	Identification of the initiator and target of all trusted channels
FTP_TRP.1(1)	All attempted uses of the trusted path functions	Identification of the claimed user identity
FTP_TRP.1(2)	All attempted uses of the trusted path functions	Identification of the claimed user identity
IDS_COL_EXP.1	None	
IDS_RDR_EXP.1	Access to the Sensor Data	
IDS_STG_EXP.1	Access to the Sensor Data	
IDS_STG_EXP.2	Access to the Sensor Data	

Table 7 - Auditable Events

FAU_GEN.2 User Identity Association

FAU_GEN.2.1 - **Refinement:** The TSF shall be able to associate each auditable event with the identity of the Administrator or User that caused the event.

FAU_SAA.1 Potential violation analysis

FAU_SAA.1.1 - **Refinement:** The TSF shall be able to apply a set of rules in monitoring events and based upon these rules indicate a potential violation of the TSP.

FAU_SAA.1.2 - **Refinement:** The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [

(1) Security Administrator specified number of authentication failures;

(2) *Security Administrator specified number of Information Flow policy violations by an individual presumed source network identifier (e.g., IP address) within a Security Administrator specified time period;*

(3) *Security Administrator specified number of Information Flow policy violations to an individual destination network identifier within a Security Administrator specified time period;*

(4) *Security Administrator specified number of Information Flow policy violations to an individual destination subject service identifier (e.g., TCP port) within a Security Administrator specified time period;*

(5) *Security Administrator specified Information Flow policy rule, or group of rule violations within an Security Administrator specified time period;*

(6) *Any detected replay of TSF data or security attributes;*

(7) *Any failure of the cryptomodule self-tests (FPT_TST.1(2));*

(8) *Any failure of the other TSF self-tests (FPT_TST.1(1));*

(9) *Security Administrator specified number of encryption failures;*

(10) *Security Administrator specified number of decryption failures;*

(11) *Security Administrator specified number of Phase 1 authentication failures when negotiating the Internet Key Exchange protocol; and*

(12) *Security Administrator specified number of failures occurring during Phase 2 negotiation.*]

known to indicate a potential security violation;

b) [*the following additional rules:*

(1) *Security Administrator specified number of Anti-Virus or IPS Protection Profile*[54] *violations;*

(2) *Security Administrator specified percentage of available audit storage usage; and*

(3) *Audit storage exhaustion;*

[54] The term 'Protection Profile' is also used by Fortinet and is not to be confused with the CC terminology.

known to indicate a potential security violation].

FAU_SAR.1 Audit review

FAU_SAR.1.1 The TSF shall provide [*the Administrators*] with the capability to read [*all audit data*] from the audit records.

FAU_SAR.1.2 - **Refinement:** The TSF shall provide the audit records in a manner suitable for the Administrators to interpret the information.

FAU_SAR.2 Restricted audit review

FAU_SAR.2.1 - **Refinement:** The TSF shall prohibit all users read access to the audit records, except the Administrators.

FAU_SAR.3 Selectable audit review

FAU_SAR.3.1 - **Refinement:** The TSF shall provide the ability to perform [searches and sorting] of audit data based on:

a) [*user identity;*

b) *source subject identity;*

c) *destination subject identity;*

d) *ranges of one or more: dates, times, user identities, subject service identifiers, or transport layer protocol;*

e) *type of event (i.e., rule identity for firewall events);*

f) *TOE network interfaces;*

g) *log severity level;*

h) *action;*

i) *source interface;*

j) *destination interface; and*

k) *success or failure of related event*].

FAU_SEL.1 Selective Audit

FAU_SEL.1.1 - **Refinement:** The TSF shall allow only the Security Administrator to include or exclude auditable events from the set of audited events based on the following attributes:

a) [user identity;

b) event types consisting of traffic flow or security events];

c) [

(1) network identifier;

(2) subject service identifier;

(3) success of auditable security events;

(4) failure of auditable security events;

(5) rule identity; and

(6) *event severity level*].

FAU_STG.2 Guarantees of audit data availability

FAU_STG.2.1 – **Refinement:** The TSF shall protect the stored audit records in the audit trail from deletion by any role except the Audit Administrator.

FAU_STG.2.2 – **Refinement:** The TSF shall be able to [prevent] modifications to the stored audit records in the audit trail.

FAU_STG.2.3 The TSF shall ensure that [*the Security Administrator's selection of all or the most recent*] audit records will be maintained when the following conditions occur: [audit storage exhaustion].

FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 – **Refinement:** The TSF shall [

a) *immediately alert the administrators by displaying a message at the Local Console, Network Web-Based GUI and Network CLI when an administrative session exists for each of the defined administrative roles;*

b) *at the option of the Security Administrator, immediately alert the administrators by generating an audible alarm at the Local Console, Network Web-Based GUI and Network CLI when an administrative session exists for each of the defined administrative roles; and*

c) *at the option of the Security Administrator generate an audit record*]

if the audit trail exceeds [*a Security Administrator settable percentage of storage capacity*].

FAU_STG.4 Prevention of audit data loss

FAU_STG.4.1 - **Refinement:** The TSF shall generate an alarm and in addition shall provide the Security Administrator the capability to select one or more of the following additional actions:

- [prevent auditable events, except those taken by the Security Administrator and Audit Administrator;
- overwrite the oldest stored audit records;
- generate an audible alarm]

if the audit trail is full.

FAU_STG.4.2 – **Refinement**: The TSF shall enforce the Security Administrator's selection(s) if the audit trail is full.

5.1.2 Anti Virus Actions (FAV)

FAV_ACT_EXP.1 Anti Virus Actions

FAV_ACT_EXP.1.1 – The TSF shall provide the Security Administrator the capability to select one or more of the following actions:

- block the transmission of the information flow;
- quarantine the content of the information flow..

to be taken on detection of a virus in an information flow.

FAV_ACT_EXP.1.2 – The TSF shall provide a secure mechanism to update the virus signatures used by the TSF.

Application Note: Virus signature updates consist of updates to both the virus signature database and the processing engine for the detection of virus attacks. The TOE provides specific guidance to administrators noting that in the evaluated configuration of the TOE, only the virus signature database updates may be applied to the TOE.

5.1.3 Cryptographic Support (FCS)

This section specifies the cryptographic support required in the TOE. As previously stated the cryptographic support is required for authentication mechanisms, for trusted path, trusted channel and for integrity mechanisms. The cryptographic requirements are structured to accommodate use of the FIPS 140-2 standard and the CSE and NIST Cryptographic Module Validation Program (CMVP) in meeting the requirements, and to accommodate use of multiple cryptographic modules in meeting the required cryptographic functionality.

FCS_BCM_EXP.1 Baseline Cryptographic Module

FCS_BCM_EXP.1.1 – All cryptographic functions implemented by the TOE that are FIPS-approved cryptographic functions shall be implemented in a crypto module that is FIPS PUB 140-2 validated, and perform the specified cryptographic functions in a FIPS-approved mode of operation.

FCS_BCM_EXP.1.2 – **Refinement**: All FIPS-validated cryptographic modules implemented in the TSF shall have a minimum overall Security Level 1, meet Security Level 3 for the following: cryptographic module ports and interfaces; roles, services and authentication; cryptographic key management, and design assurance, and meet FIPS PUB 140-2, Level 4 Self Tests.

FCS_CKM.1(1) Cryptographic Key Management (Key Generation)

FCS_CKM.1.1(1) – **Refinement:** The FIPS-validated cryptomodule shall generate symmetric cryptographic keys using a FIPS-Approved Random Number Generator for all key sizes that meet the following: National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-2, January 27, 2000 – Appendix 3.1 with Change Notice 1.

FCS_CKM.1.2(1) The TSF shall generate[55] asymmetric[56] cryptographic keys in accordance with a domain parameter generator and a random number generator and/or a prime number generator that meet the following: [

a) Generated key strength shall be equivalent to, or greater than, a symmetric key strength of 128 bits using conservative estimates;

[55] This requirement applies strictly to generation of asymmetric keys. Validation techniques for generated asymmetric keys are discussed in FCS_CKM.1(4).

[56] These are the keys/parameters (e.g., the public/private key pairs) underlying a public key-based key establishment scheme, not the session keys established by such schemes.

b) ANSI X9.80 (3 January 2000), Prime Number Generation, Primality Testing, and Primality Certificates using random integers with deterministic tests, or constructive generation methods;

c) Case: For domain parameters used in finite field-based key establishment schemes

- ANSI X9.42-2001, Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography[57];

Application Note: For example, "Classic" Diffie-Hellman-based schemes

d) Case: For domain parameters used in RSA-based key establishment schemes (with odd e)

- ANSI X9.31-1998 (May 1998), Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA) for the generation of the RSA parameters[58]; and

e) Case: For domain parameters used in elliptic curve-based key establishment schemes

- ANSI X9.63-200x (1 Oct 2000), Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport using Elliptic Curve Cryptography[59].

FCS_CKM.1(2) Cryptographic Key Management (Key Establishment for Symmetric Keys)

FCS_CKM.1.1(2) – The cryptomodule shall provide the following FIPS-supported security function cryptographic key establishment technique(s) for AES symmetric keys:

- Cryptographic Key Establishment using Discrete Logarithm Key Agreement that meets the following:

[57] Any pseudorandom RNG used in these schemes for generating private values shall be seeded by a nondeterministic RNG (both types of RNGs meeting RNG requirements in this ST).

[58] A pseudorandom RNG seeded by a nondeterministic RNG (both types of RNGs meeting RNG requirements in this ST) shall be used in the generation of these primes.

[59] Any pseudorandom RNG used in these schemes for generating private values shall be seeded by a nondeterministic RNG (both types of RNGs meeting RNG requirements in this ST).

a) The cryptomodule shall provide the capability to act as the initiator or responder (that is, act as Party U or Party V as defined in the standard) to agree on cryptographic keys of all sizes using the [dhEphem] key agreement scheme where domain parameter p is a prime of 3072 bits and domain parameter q is a prime of 1024 bits, and that conforms with ANSI X9.42-2001, Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography.

b) The cryptomodule shall conform to the standard using a FIPS-approved MAC function, a FIPS-approved Random Number generation function, and a FIPS-approved Hashing function.

c) The choices and options used in conforming to the key agreement scheme(s) are as follows:

- domain parameter generation: algorithm in section 7.1 of ANSI X9.42-2001;
- domain parameter validation: algorithm in Annex B.1.3 of ANSI X9.42-2001;
- private/public key generation: algorithm specified in section 7.4 of ANSI X9.42-2001;
- public key validation method used: method 1 of section 7.4 of ANSI X9.42-2001;
- key derivation method: RFC 4253;
- hash algorithm: SHA-1;
- probabilistic test: Miller-Rabin;
- random number generation method used: FIPS-186-2 Appendix 3.1 with Change Notice 1.

FCS_CKM.1(3) Cryptographic Key Management (Key Entry for Digital Signature/Verification Private Keys)

FCS_CKM.1.1(3) – The FIPS-validated cryptomodule shall provide the following cryptographic key entry technique(s) for the private key used for the asymmetric algorithm rDSA:

- Cryptographic Key Establishment using Automated Methods

- The FIPS-validated cryptomodule shall be able to accept as input cryptographic keys using key management techniques that meet the following:

    - The TSF shall provide the capability to directly attach a key device by [[*FortiUSB token*]];

    - The [TSF] shall perform key error detection scheme on keys input via electronic methods using [[*verification of certificate structure*]]; and

    - FIPS 140-2 Key Management Security Level 3, Key Entry and Output.

FCS_CKM.1(4) Cryptographic Key Management (Key Validation and Packaging)

FCS_CKM.1.1(4) – The TSF shall apply validation techniques (e.g., parity bits or checkwords) to generated symmetric keys in accordance with:

a) FIPS PUB 46-3 (Data Encryption Standard (DES)), and

b) FIPS PUB 171[60] (Key Management Using ANSI X9.17).

FCS_CKM.1.2(4) – The TSF shall apply validation techniques to generated asymmetric keys in accordance with the standards corresponding to the generation technique as called out in FCS_CKM.1.2(1).

FCS_CKM.1.3(4) – Any public key certificates generated by the TSF shall be in accordance with NSA-certified NSA-approved certificate schemes[61].

FCS_CKM.1(5) Cryptographic Key Management (Internet Key Exchange)

FCS_CKM.1.1(5) – The TSF shall provide cryptographic key establishment techniques in accordance with RFC 2409 as follows(s):

[60] For purposes of interpreting this standard, only TDEA with 168 bits of key shall be applied (DES is not acceptable for meeting this requirement. Eventual migration to AES is expected.).

[61] DoD multilevel applications require Class 5 PKI to address worst case environments, but currently this class is just a concept. In the interim, NSA-approved certificate schemes with hardware tokens for protection of private keys are approved under the added requirement that stronger protection mechanisms must be applied at the boundaries of the protected environment as stated earlier in this PP. When Class 5 certificates are fully established, they will be required.

- Phase 1, the establishment of a secure authenticated channel between the TOE and another remote VPN endpoint, shall be performed using one of the following, as configured by the security administrator:
    - Main Mode
    - Aggressive Mode
- New Group mode shall include the private group 14, 2048-bit MOD P, [no other group modes] for the Diffie-Hellman key exchange.
- Phase 2, negotiation of security services for IPSec, shall be done using Quick Mode, using SHA-1 as the pseudo-random function. Quick Mode shall generate key material that provides perfect forward secrecy.

FCS_CKM.1.2(5) – The TSF shall require the **nonce, and the x of g^xy** be randomly generated using FIPS-approved random number generator when computation is being performed.

- The recommended nonce sizes are to be between 8 and 256 bytes;
- The minimum size for the x should be 256 bits.

FCS_CKM.1.3(5) – When performing authentication using pre-shared keys, the key shall be generated using the FIPS approved random number generator specified in FCS_COP.1(4).

FCS_CKM.1.4(5) – The TSF shall compute the value of SKEYID (as defined in RFC 2409), using SHA-1 as the pseudo-random function. The TSF shall be capable of authentication using the methods for

- Signatures: SKEYID = sha(Ni_b | Nr_b, g^xy)
- Pre-shared keys: : SKEYID = sha(pre-shared-key, Ni_b |Nr_b)
- [Authentication using Public key encryption, computing SKEYID as follows: SKEYID = sha(sha(Ni_b | Nr_b), CKY-I |CKY-R)]

Application Note: Refer to RFC 2409 for an explanation of the notation and definitions of the terms.

FCS_CKM.1.5(5) – The TSF shall compute authenticated keying material as follows:

- SKEYID_d = sha(SKEYID, g^xy | CKY-I | CKY-R | 0)
- SKEYID_a = sha(SKEYID, SKEYID_d | g^xy | CKY-I | CKY-R |1)
- SKEYID_e = sha(SKEYID, SKEYID_a | g^xy | CKY-I | CKY-R | 2)

FCS_CKM.1.6(5) – To authenticate the Phase 1 exchange, the TSF shall generate HASH_I if it is the initiator, or HASH_R if it is the responder as follows:

- HASH_I = sha(SKEYID, g^xi | g^xr | CKY-I | CKY-R | SAi_b |IDii_b)
- HASH_R = sha(SKEYID, g^xr | g^xi | CKY-R | CKY-I | SAi_b |IDir_b)

Application Note: Refer to RFC 2409 for an explanation of the notation and definitions of the terms.

FCS_CKM.1.7(5) – The TSF shall be capable of authenticating IKE Phase 1 using the following methods as defined in RFC 2409, as configured by the security administrator:

a) **Authentication with digital signatures**: The TSF shall use [RSA, DSA,[]]

b) when an RSA signature is applied to HASH I or HASH R it must be first PKCS#1 encoded. The TSF shall check the HASH_I and HASH_R values sent against a computed value to detect any changes made to the proposed transform negotiated in phase one. If changes are detected the session shall be terminated and an alarm shall be generated.

c) [[*X.509 Version 3 certificates* []] X.509 V3 implementations, if implemented, shall be capable of checking for validity of the certificate path, and at option of SA, check for certificate revocation.

d) **Authentication with a pre-shared key**: The TSF shall allow authentication using a pre-shared key.

FCS_CKM.1.8(5) – The TSF shall compute the hash values for Quick Mode in the following way

HASH(1) = sha(SKEYID_a, M-ID |[*any ISAKMP payload after HASH(1) header contained in the message*)]
HASH(2) = sha(SKEYID_a, M-ID | Ni_b | [*any ISAKMP payload after HASH(2) header contained in the message*)]
HASH(3) = sha(SKEYID_a, 0 | M-ID | Ni_b | Nr_b)

Application Note: The following steps will be performed when using the HASH computation:

- *initiator computes HASH(1) and sends to responder*
- *responder validates computation of HASH(1) and computes HASH(2) and sends HASH(2) to initiator*
- *initiator validates computation of HASH(2) and computes HASH(3) and sends HASH(3) to responder*

KE is only optional when SA elects not to use perfect forward secrecy.

Verifying that a TFS implementation actually checks HASH(1), HASH(2), and HASH(3) values sent against a computed value is important in detecting changes that could have been made to proposed transform negotiated in Quick Mode (not as likely as Phase One because Quick Mode is encrypted).

The ordering of the ISAKMP payloads may differ because Quick Mode only specifies the location of the HASH and SA payload.

FCS_CKM.1.9(5) – The TSF shall compute new keying material during Quick Mode as follows:

[when using perfect forward secrecy KEYMAT = sha(SKEYID_d, g(qm)^xy | protocol | SPI | Ni_b| Nr_b),When perfect forward secrecy is not used KEYMAT = sha(SKEYID_d | protocol | SPI | Ni_b | Nr_b)]

FCS_CKM.1.10(5) – The TSF shall at a minimum, support the following ID types:

[assignment: ID_IPV4_ADDR, ID_FQDN, ID_USER_FQDN, ID_IPV4_ADDR_SUBNET, ID_IPV6_ADDR, ID_IPV6_ADDR_SUBNET, ID_IPV4_ADDR_RANGE, ID_IPV6_ADDR_RANGE, ID_DER_ASN1_DN, ID_DER_ASN1_GN, ID_KEY_ID].

FCS_CKM.2(1) Cryptographic Key Management (Key Handling and Storage)

FCS_CKM.2.1(1) – The TSF shall perform key entry and output in accordance with FIPS PUB 140-2, Level 3.

FCS_CKM.2.2(1) – The TSF shall provide a means to ensure that keys are associated with the correct entities (i.e., person, group, or process) to which the keys are assigned.

FCS_CKM.2.3(1) – The TSF shall perform a key error detection check on each transfer of key (internal, intermediate transfers).

Application Note: A parity check is an example of a key error detection check.

FCS_CKM.2.4(1) – The TSF shall encrypt or split persistent secret and private keys when not in use.

Application Note: A persistent key, such as a file encryption key, is one that must be available in the system over long periods of time. A non-persistent key, such as a key used to encrypt or decrypt a single message or a session, is one that is ephemeral in the system.

Application Note: "When not in use" shall be interpreted in the strictest sense so that persistent keys only exist in plaintext form during intervals of operational necessity. For example, a file encryption key shall exist in plaintext form only during actual encryption and/or decryption processing of a file. Once the file is decrypted or encrypted the file encryption key shall be immediately covered for protection.

FCS_CKM.2.5(1) – The TSF shall destroy non-persistent cryptographic keys after an administrator-defined period of time of inactivity.

FCS_CKM.2.6(1) – The TSF shall overwrite each intermediate storage area for plaintext key/critical cryptographic security parameter (i.e., any storage, such as memory buffers, that is included in the path of such data). This overwriting shall be executed three or more times using a different alternating data pattern each time upon the transfer of the key/critical cryptographic security parameter to another location.

Application Note: This is related to the elimination of internal, temporary copies of plaintext keys created during processing, not to the total destruction of a key from the TOE which is discussed under Key Destruction. Although verification of the zeroization of each intermediate location of a plaintext key/critical cryptographic security parameter is desired here (by checking for the final known alternating data pattern), it is not required at this time. However vendors are highly encouraged to incorporate this verification whenever possible into their implementations.

FCS_CKM.2.7(1) – The TSF shall prevent archiving of expired (private) signature keys.

Application Note: This requirement is orthogonal to typical system back-up procedures. Therefore, it does not address the problem of archiving an active (private) signature key during a system back-up and saving the key beyond its intended life span.

FCS_CKM.2(2) Cryptographic Key Management (Key Distribution)

FCS_CKM.2.1(2) – The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method: [Manual (Physical) Method and Automated (electronic) Method] that meets the following:

a) Manual (Physical) Methods:

- The TSF shall support manual distribution of symmetric key in accordance with FIPS PUB 171 (Key Management Using ANSI X9.17).
- The TSF shall support manual distribution of private asymmetric key material (certificates and/or keys) in accordance with NSA-

certified Department of Defense (DoD) Public Key Infrastructure (PKI) for public key distribution using NSA-approved certificate schemes with hardware tokens for protection of private keys that meet the following:

1) PKI Roadmap for the DoD,
2) DoD X.509 Certificate Policy,
3) PKCS #8 v1.2 (Private-Key Information Syntax Standard),
4) PKCS #12 v1.0 (Personal Information Exchange Syntax),
5) PKCS #5 v2.0 (Password-Based Encryption Standard, 25 Mar 1999 - Final), and
6) PKCS #11 v2.11 (Cryptographic Token Interface Standard).

- The TSF shall support manual distribution of public asymmetric key material (certificates and/or keys) in accordance with NSA-certified DOD PKI for public key distribution using NSA-approved certificate schemes for protection of public keys that meet the following:

1) PKI Roadmap for the DoD,
2) DoD X.509 Certificate Policy,
3) PKCS#12 v1.0 (Personal Information Exchange Syntax),

b) Automated (Electronic) Methods:

- The TSF shall automatically distribute symmetric keys in accordance with FIPS PUB 171 (Key Management Using ANSI X9.17).[62].

- The TSF shall automatically distribute public asymmetric key material (certificates and/or keys) in accordance with NSA-certified DoD PKI for public key distribution using NSA-approved certificate schemes[63] that meet the following:

[62] Until NIST identifies approved methods for automatically distributing symmetric key, FIPS PUB 171 (Key Management Using ANSI X9.17) is being used here. For purposes of interpreting FIPS PUB 171, only TDEA with 168 bits of key shall be applied. (DES is not acceptable for meeting this requirement. Eventual migration to AES is expected.) Where public key schemes are used in key transport methods, NIST Special Publication 800-56 ("Recommendation on Key Establishment Schemes"; DRAFT 2.0, January 2003) shall also be used.

[63] DoD multilevel applications require Class 5 PKI to address worst case environments, but currently this class is just a concept. In the interim, NSA-approved certificate schemes with hardware tokens for protection of private key are approved under the added requirement that stronger protection mechanisms must be applied at the boundaries of the protected environment as stated earlier in this PP. When Class 5 certificates are fully established, they will be required.

1) PKI Roadmap for the DoD,
2) DoD X.509 Certificate Policy,
3) PKCS#12 v1.0 (Personal Information Exchange Syntax),

- The TSF shall only support manual distribution of private asymmetric key material (certificates and/or keys) in accordance with NSA-certified DOD PKI for public key distribution using NSA-approved certificate schemes[64] with hardware tokens for protection of private keys that meet the following:

  1) PKI Roadmap for the DoD,
  2) DoD X.509 Certificate Policy,
  3) PKCS #8 v1.2 (Private-Key Information Syntax Standard)
  4) PKCS #12 v1.0 (Personal Information Exchange Syntax Standard)
  5) PKCS #5 v2.0 (Password-Based Encryption Standard, 25 Mar 99--Final) and,
  6) PKCS #11 v2.11 (Cryptographic Token Interface Standard).

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1 – **Refinement:** The TSF shall destroy cryptographic keys in accordance with a cryptographic key zeroization method that meets the following:

a) The Key Zeroization Requirements in FIPS PUB 140-2 Key Management Security Level 3;

b) Zeroization of all private cryptographic keys, plaintext cryptographic keys and all other critical cryptographic security parameters shall be immediate and complete;

c) The zeroization shall be executed by overwriting the key/critical cryptographic security parameter storage area; and

d) The TSF shall overwrite each intermediate storage area for private cryptographic keys, plaintext cryptographic keys, and all other critical security parameters three or more times with a different alternating pattern each time upon the transfer of the key/CSPs to another location.

Application Note: The TOE stores the cryptographic keys in flash memory. Flash RAM has a limited number of supported read/write cycles. To avoid burning out

[64] See previous footnote.

areas of the flash RAM through repeated use, the device drivers for the flash RAM used in the FortiGate products do not re-write data to the same location when updating files. Files and data locations are moved around the flash card to evenly distribute the use.

Deleting data from flash memory is immediate and complete, leaving no magnetic residue or signature. There is therefore no need to overwrite a cryptographic key stored in flash memory multiple times in order to ensure that the data can not be recovered.

For these reasons, the requirement in FCS_CKM.4.1 c) is not directly applicable. The intent of the requirement, that data be permanently destroyed, is met as described previously. Rewriting data multiple times to destroy keys/critical cryptographic security parameters does not provide additional protection when using flash based data storage.

FCS_COP.1(1) Cryptographic Operation (Encryption/Decryption AES)

FCS_COP.1.1(1) – A cryptomodule shall perform encryption and decryption using the FIPS-Approved Security Function AES algorithm operating in CBC mode(s) supporting key sizes of 128 bits, 192 bits, 256 bits.

FCS_COP.1(2) Cryptographic Operation (Digital Signature Generation/Verification)

FCS_COP.1.1(2) – A cryptomodule shall perform digital signature generation and verification using the FIPS-Approved Security Function [selection*:*

- rDSA

  a) The cryptomodule shall implement rDSA (rDSA with odd e) with a modulus size of [2048 bits or greater] in a manner that conforms to ANSI X9.31-1998, Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA).

  b) The choices and options used in conforming to the X9.31-1998 are as follows:

  - public verification exponent, e: fixed at 17;
  - hash algorithm: SHA-1;
  - random number generation method used: ANSI X9.31 Appendix A.2.4 with AES;
  - SEED value(s) for key generation: generated from RNG; and

- private signature key options: d and n derived, p and q generated.]

FCS_COP.1(3) Cryptographic Operation (Hashing Functions)

FCS_COP.1.1(3) – The TSF shall perform non-VPN Cryptographic Hashing Functions used by other cryptographic functionality of the TSF using a FIPS-approved Cryptographic Hashing Function implemented in a FIPS-approved cryptomodule running in a FIPS-approved mode.

FCS_COP.1.2(3) – The TSF shall perform *VPN* cryptographic hashing services in accordance with a NIST-approved hash implementation of the Secure Hash algorithm and message digest size of at least 256 bits that meets the following:
FIPS PUB 180-2.

Application Note: The message digest size should correspond to double the system encryption key strength.

FCS_COP.1(4) Cryptographic Operation (Random Number Generation)

FCS_COP.1.1(4) – The TSF shall perform all Random Number Generation used by the cryptographic functionality of the TSF, as well as all SFRs that require random numbers, using a FIPS-approved Random Number Generator implemented in a FIPS-approved cryptomodule running in a FIPS-approved mode.

Note: The RNG used by the TOE is that specified in FIPS 186-2 Appendix 3.1 with Change Notice 1.

FCS_COP.1.2(4) – The TSF shall defend against tampering of the random number generation (RNG)/ pseudorandom number generation (PRNG) sources.

Application Note: The RNG/PRNG should be resistant to manipulation or analysis of its sources, or any attempts to predictably influence its states. Three examples of very different approaches the TSF might pursue to address this include: a) identifying the fact that physical security must be applied to the product, b) applying checksums over the sources, or c) designing and implementing the TSF RNG with a concept similar to a keyed hash (e.g., where periodically, the initial state of the hash is changed unpredictably and each change is protected as when provided on a tamper-protected token, or in a secure area of memory.

5.1.4 User data protection (FDP)

FDP_IFC.1(1) Subset information flow control (unauthenticated policy)

FDP_IFC.1.1(1) – **Refinement:** The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] on

- [source subject: TOE interface on which information is received;
- destination subject: TOE interface to which information is destined;
- information: network packets; and
- operations: pass information, pass information via application proxy (SMTP) by opening a relay connection from the TSF on behalf of the source subject to the destination subject service identifier, and with the TSF ensuring the following conditions:

  a) the connection from the source subject is terminated at the proxy,

  b) the new relay connection is established between the proxy and destination subject, which does not use the stateful protocol attributes associated with the terminated connection in (a).]

FDP_IFC.1(2) Subset information flow control (authenticated policy)

FDP_IFC.1.1(2) – **Refinement:** The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] on

a) [source subject representing authenticated proxy user: source network identifier;

d) destination subject: TOE interface to which information is destined;

e) information: network packets; and

f) operations: pass information via application proxy (FTP, Telnet, HTTP) by opening a relay connection from the TSF on behalf of the source subject to the destination subject service identifier, and with the TSF ensuring the following conditions:

a) the connection from the source subject is terminated at the proxy,

b) the new relay connection is established between the proxy and destination subject, which does not use the stateful protocol attributes associated with the terminated connection in (a).]

FDP_IFC.1(3) Subset information flow control (unauthenticated TOE services policy)

FDP_IFC.1.1(3) - The TSF shall enforce the [UNAUTHENTICATED TOE SERVICES SFP] on

a) [source subject: TOE interface on which information is received;

b) destination subject: the TOE;

c) information: network packets; and

d) operations: accept or reject network packet].

FDP_IFC.1(4) Subset information flow control (VPN policy)

FDP_IFC.1.1(4) – **Refinement**: The TSF shall enforce the [VPN SFP] on

a) [source subject: TOE interface on which information is received;

b) destination subject: TOE interface to which information is destined.

c) information: network packets; and

d) operations:

   i) pass packets without modifying;

   ii) send IPSec encrypted and authenticated packets to a peer TOE using ESP in tunnel mode as defined in RFC 2406; and

   iii) decrypt, verify authentication and pass received packets from a peer TOE in tunnel mode using ESP].

FDP_IFF.1(1) Simple security attributes (unauthenticated policy)

FDP_IFF.1.1(1) – **Refinement**: The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] based on the following types of subject and information security attributes:

a) Source subject security attributes:

- set of source subject identifiers.

b) Destination subject security attributes:

- Set of destination subject identifiers; and
- [*schedule*].

c) Information security attributes:

- presumed identity of source subject[65];
- identity of destination subject;
- transport layer protocol;
- source subject service identifier;
- destination subject service identifier (e.g., TCP or UDP destination port number);
- [[*Schedule:*
    - *One-time schedule*
        - *Start Time*
        - *End Time*
    - *Recurring schedule*
        - *Days of week on which schedule is active*
        - *Start Time*
        - *End Time*]].
- SMTP
    1) commands (i.e., HELO, EHLO, HELP, MAIL, RCPT, DATA, QUIT, RSET, VRFY, NOOP, EXPN, TURN, SEND, SOML, SAML, SIZE);
    2) MIME Content-Types and Sub-Types:
        a. text:
            - plain
            - richtext

[65] The TOE can make no claim as to the real identity of any source subject; the TOE can only suppose that such identities are accurate. Therefore, a ‘presumed identity’ is used to identify source subjects. Note, however, that the TOE can ensure that the identity is included in the set that is associated with the interface (see FDP_IFF.1.6(1)).

- enriched

b. multipart:

- mixed
- parallel
- digest
- alternative

c. message:

- rfc822
- partial
- external-body

d. application:

- octet-stream
- postscript

e. image

f. audio

g. video.

d) Stateful packet attributes: [for IP-based network stacks:

- ➢ Connection-oriented protocols:
  - sequence number;
  - acknowledgement number;
  - Flags;
    - SYN;
    - ACK;
    - RST;

        - FIN.

    - Connectionless protocols:

        - source and destination network identifiers;
        - source and destination service identifiers].

FDP_IFF.1.2(1) **–** The TSF shall permit an information flow between a source subject and a destination subject via a controlled operation if the following rules hold:

- [the presumed identity of the source subject is in the set of source subject identifiers;
- the identity of the destination subject is in the set of source destination identifiers;
- the information security attributes match the attributes in an information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [*first match*]; and
- the selected information flow policy rule specifies that the information flow is to be permitted].

Application Note: The TOE implements its AV and IPS measures using protection profiles which may be included as a part of any firewall rule. A protection profile may be created which causes the TOE to scan packets of the following protocol types (HTTP, FTP, SMTP, POP3, IMAP and IM) for viruses before the packet is permitted to pass throught the TOE. A protection profile may also cause the TOE to scan packets for signatures which match IPS attack signatures held by the TOE.

FDP_IFF.1.3(1) - The TSF shall enforce the [following:

- fragmentation rule:
    - prior to applying the information policy ruleset, the TOE completely reassembles fragmented packets;
- stateful packet inspection rules:
    - whenever a packet is received that is not associated with an allowed established session (e.g., the SYN flag is set without the ACK flag being set), the information flow policy ruleset, as defined in FDP_IFF.1.2(1), is applied to the packet;

- o otherwise, the TSF associates a packet with an allowed established session using the stateful packet attributes].

FDP_IFF.1.4(1) - The TSF shall provide the following [the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied].

FDP_IFF.1.5(1) - The TSF shall explicitly authorize an information flow based on the following rules: [none].

FDP_IFF.1.6(1) - The TSF shall explicitly deny an information flow based on the following rules:

a) [The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE is not included in the set of source identifiers for the source subject;

b) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a broadcast identity;

c) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier;

d) The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the destination subject; and

e) The TOE shall reject SMTP traffic that contains source routing symbols (e.g., in the mailer RCPT commands)]

FDP_IFF.1(2) Simple security attributes (authenticated policy)

FDP_IFF.1.1(2) – **Refinement:** The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] based on the following types of subject and information security attributes:

a) [Source subject security attributes:

- source network identifier.

b) Destination subject security attributes:

- Set of destination subject identifiers.

c) Information security attributes:

- identity of source subject;
- identity of destination subject;
- transport layer protocol;
- destination subject service identifier (e.g. TCP destination port number):
- FTP sub-commands specified in RFC 959, and the optional commands introduced by RFC 2228;
- HTTP request methods specified in RFC 2616.

d) Stateful packet attributes: [for IP-based network stacks:

- ➢ Connection-oriented protocols:
  - ▪ sequence number;
  - ▪ acknowledgement number;
  - ▪ Flags;
    - SYN;
    - ACK;
    - RST; and
    - FIN.
- ➢ Connectionless protocols:
  - ▪ source and destination network identifiers; and
  - ▪ source and destination service identifiers].

FDP_IFF.1.2(2) – **Refinement:** The TSF shall permit an information flow between a source subject and a destination subject via a controlled operation if the following rules hold:

- [the source subject has successfully authenticated to the TOE;
- the identity of the destination subject is in the set of destination identifiers;

- the information security attributes match the attributes in a information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [*first match*]; and

- the selected information flow policy rule specifies that the information flow is to be permitted via the authenticated proxy selected by the rule].

FDP_IFF.1.3(2) – The TSF shall enforce the [following

- fragmentation rule:
  - prior to applying the information policy ruleset, the TOE completely reassembles fragmented packets;
- stateful packet inspection rules:
  - whenever a packet is received that is not associated with an allowed established session (e.g., the SYN flag is set without the ACK flag being set), the information flow policy ruleset, as defined in FDP_IFF.1.2(2), is applied to the packet;
  - otherwise, the TSF associates a packet with an allowed established session using the stateful packet attributes].

FDP_IFF.1.4(2) – **Refinement:** The TSF shall provide the following [the Security Administrator shall have the capability to view all information flows allowed by this information flow policy ruleset before the ruleset is applied].

FDP_IFF.1.5(2) – The TSF shall explicitly authorize an information flow based on the following rules: [none].

FDP_IFF.1.6(2) – The TSF shall explicitly deny an information flow based on the following rules: [none].

FDP_IFF.1(3) Simple security attributes (unauthenticated TOE services policy)

FDP_IFF.1.1(3) - The TSF shall enforce the [*UNAUTHENTICATED TOE SERVICES SFP*] based on the following types of subject and information security attributes:

a) [Source subject security attributes:

- set of source subject identifiers.

b) Destination subject security attributes:

- TOE's network identifier.

c) Information security attributes:

- presumed identity of source subject;
- identity of destination subject;
- transport layer protocol;
- source subject service identifier;
- destination subject service identifier (e.g., TCP or UDP destination port number); and
- [for an IP-based network stack: ICMP message type and code as specified in RFC 792]].

FDP_IFF.1.2(3) – The TSF shall permit an information flow between a source subject and the TOE via a controlled operation if the following rules hold:

a) [the presumed identity of the source subject is in the set of source subject identifiers;

b) the identity of the destination subject is the TOE;

c) the information security attributes match the attributes in an information flow control policy according to the following algorithm [*first match*].

FDP_IFF.1.3(3) – **Refinement**: The TSF shall enforce the [following rules:

- The TOE shall allow source subjects to access TOE services [for IP-based network stacks: ICMP] without authenticating those source subjects; and
- The TOE shall allow the list of services specified immediately above to be enabled (become available to unauthenticated users) or disabled (become unavailable to unauthenticated users)].

FDP_IFF.1.4(3) - The TSF shall provide the following [the Security Administrator shall have the capability to view all information flows allowed by this information flow control policy before the policy is applied].

FDP_IFF.1.5(3) - The TSF shall explicitly authorize an information flow based on the following rules: [none].

FDP_IFF.1.6(3) - The TSF shall explicitly deny an information flow based on the following rules: [

a) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE is not included in the set of source identifiers for the source subject;

b) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a broadcast identity;

c) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier; and

d) The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the TOE].

FDP_IFF.1(4) Simple security attributes (VPN policy)

FDP_IFF.1.1(4) – **Refinement**: The TSF shall enforce the [VPN SFP] based on the following types of subject and information security attributes: [

a) Source subject security attributes:

- set of source subject identifiers.

b) Destination subject security attributes:

- set of destination subject identifiers.

c) Information security attributes:

- presumed identity of source subject; and
- identity of destination subject.

FDP_IFF.1.2(4) – The TSF shall permit an information flow between a source subject and a destination subject via a controlled operation if the following rules hold:

a) [the presumed identity of the source subject is in the set of source subject identifiers;

b) the identity of the destination subject is in the set of source destination identifiers;

c) the information security attributes match the attributes in an information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [*first match*]; and

d) the selected information flow policy rule specifies that the information flow is to be permitted, and what specific operation from FDP_IFC.1(3) is to be applied to that information flow].

FDP_IFF.1.3(4) – The TSF shall enforce the [*no additional VPN SFP rules*]

FDP_IFF.1.4(4) – The TSF shall provide the following [*the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied*].

FDP_IFF.1.5(4) - The TSF shall explicitly authorize an information flow based on the following rules: [*none*].

FDP_IFF.1.6(4) - The TSF shall explicitly deny an information flow based on the following rules: [

a) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE is not included in the set of source identifiers for the source subject;

b) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a broadcast identity;

c) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier;

d) The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the destination subject.)].

FDP_RIP.2 Full residual information protection

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

5.1.5 Identification and authentication (FIA)

TOE security functions implemented by a probabilistic or permutational mechanism (e.g., password or hash function) are required (at EAL2 and higher) to include a strength of function claim. Strength of Function shall be demonstrated for the authentication mechanism used by the administrators to be SOF-Basic, as defined in Part 1 of the CC. Specifically, the local authentication mechanism must demonstrate adequate protection against attackers possessing a low attack potential.

Application Note: The term 'local authentication mechanism' in the paragraph above is duplicated from the FW PP MR and the TFFW PP MR. It denotes that the authentication mechanism used by the TOE is an integral part of the TOE rather than being provided by an external IT entity. The authentication mechanism is used by the TOE to authenticate users of the Local Console as well as remote users.

FIA_AFL.1 - Authentication failure handling

FIA_AFL.1.1 - **Refinement:** The TSF shall detect when [a Security Administrator-configurable integer] of unsuccessful authentication attempts occur related to [*administrators attempting to authenticate to the Network Web-Based GUI and Network CLI, attempted proxy user authentication and authentication attempts by VPN peers*].

FIA_AFL.1.2 – **Refinement:** When the defined number of unsuccessful authentication attempts related to the applicable item in FIA_AFL.1.1 has been met, the TSF shall [*at the option of the Security Administrator prevent:*

- *authentication via the Network Web-Based GUI and Network CLI for the user assumed to have exceeded the authentication attempt limit; and*
- *proxy user authentication for the user assumed to have exceeded the authentication attempt limit.*
- *VPN peer authentication for the VPN peer assumed to have exceeded the authentication attempt limit.*

until an action is taken by the Security Administrator, or until a Security Administrator defined time period has elapsed].

Application Note: The TSF monitors authentication failures at the Local Console and inititates an alarm when the authentication failure limit is exceeded (see FAU_SAA.1). However, the TSF does not prevent further authentication attempts at the Local Console when the authentication failure limit has been exceeded.

FIA_ATD.1(1) User attribute definition (administrators)

FIA_ATD.1.1(1) – **Refinement:** The TSF shall maintain the following list of security attributes belonging to an authorized Administrator:

a) [*user identifiers (role, username, password);and*

b) *three optional trusted host IP address/netmasks pairs from which the administrator can login*].

Application Note: The user identifiers listed is subparagraph a) above include the 'user identity', 'authentication data' and 'authorizations' listed in the IDSS PP.

FIA_ATD.1(2) User attribute definition (authorized proxy user)

FIA_ATD.1.1(2) – **Refinement:** The TSF shall maintain the following list of security attributes belonging to an authorized proxy user:

a) [*user identifiers (role, username, password); and*

b) *user group and applicable firewall policies*].

FIA_ATD.1(3) User attribute definition (VPN Remote Devices)

FIA_ATD.1.1(3) – **Refinement:** The TSF shall maintain the following list of security attributes belonging to an authorized VPN Remote Device:

a) [*IPSec Phase 1 parameters*

- *remote device identifier, the name that represents the remote VPN peer or client:*
- *role;*
- *connection type (i.e., static IP address, dialup user, or dynamic DNS);*
- *static IP address of the remote peer, if connection type is static IP address;*
- *domain name of the remote peer, if dynamic DNS is selected;*
- *Phase 1 mode (Main or aggressive);*
- *authentication method, either preshared key or RSA signature;*
- *preshared key if applicable;*
- *server certificate name that the FortiGate will use to authenticate itself to the remote peer or dialup client during phase 1 negotiations*

- *peer options, depending on the remote gateway and authentication method settings defined above;*
- *optional advanced settings for the phase 1 proposal;*

b) *IPSec Phase 2 parameters:*

- *tunnel name;*
- *phase 1 configurations associated with the tunnel; and*
- *tunnel key lifetime.*]

FIA_UAU.1(1) Timing of authentication (for TOE services)

FIA_UAU.1.1(1) - The TSF shall allow [*ICMP*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2(1) - The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.1(2) Timing of authentication (for information flow through the TOE)

FIA_UAU.1.1(2) – The TSF shall allow [*SMTP traffic to flow with mediation through the TOE*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2(2) – The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.2 User authentication before any action

FIA_UAU.2.1 – **Refinement:** The TSF shall require the administrators, Fortinet's FortiGuard Distribution Server, VPN Peers and users of [Telnet, FTP, HTTP] to be successfully authenticated before allowing any other TSF-mediated actions on behalf of these authorized users.

FIA_UAU.5 – Multiple authentication mechanisms

FIA_UAU.5.1 – **Refinement**: The TSF shall provide a local authentication mechanism and [*and a device level authentication mechanism based on X.509 certificates*] to perform user authentication.

Application Note: The 'device level authentication mechanism based on X.509 certificates' is used by the TOE for mutual authentication of VPN peers.

FIA_UAU.5.2 – The TSF shall authenticate any user's claimed identity according to the [*configuration set by the Security Administrator to define which authentication mechanism is to be used for each user*].

FIA_UID.2 User identification before any action

FIA_UID.2.1 - The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

FIA_USB.1 User-Subject Binding

FIA_USB.1.1 - The TSF shall associate the following user security attributes with subjects acting on behalf of that user: [*all user security attributes*].

FIA_USB.1.2 – The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users:

a) [*static user attributes, such as username, are assigned to users when they are created by the Security Administrator;*

b) *one of the assigned user attributes is a role;*

c) *the role defines subjects that may operate on behalf of the user;*

d) *when a user authenticates, "dynamic" user attributes, such as IP address, may be assigned; and*

e) *user-subject binding occurs when the user successfully invokes a subject to act on its behalf*].

FIA_USB.1.3 – The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users:

a) [*the associations between the user security attributes and the subjects acting on behalf of users are predefined by the FortiGate code and data tables, and cannot be changed;*

b) *the values of the static security attributes may be changed only by the Security Administrator, who is permitted to edit the TSF data defining the user; and*

c) *the values of the dynamic security attributes are assigned when the user session is created and changed programmatically as needed*].

Application Note: User security attributes are defined in FIA_ATD.1(1), FIA_ATD.1(2) and FIA_ATD.1(3).

5.1.6 Intrusion Prevention Actions (FIP)

FIP_ACT_EXP.1 Intrusion Prevention Actions

FIP_ACT_EXP.1.1 – The TSF shall prevent intrusion attacks directed at the TOE and shall provide the Security Administrator with the configurable capability to detect and prevent intrusion attacks contained within an information flow processed by the TOE.

FIP_ACT_EXP.1.2 – The TSF shall provide a secure mechanism to update the intrusion prevention signatures used by the TSF.

Application Note: Intrusion Prevention signature updates consist of updates to both the intusion prevention signature database and the processing engine for the detection of intrusion attacks. The TOE provides specific guidance to administrators noting that in the evaluated configuration of the TOE, only the signature database updates may be applied to the TOE.

5.1.7 Security management (FMT)

FMT_MOF.1(1) - Management of security functions behavior (TSF non-Cryptographic Self-test)

FMT_MOF.1.1(1) - The TSF shall restrict the ability to [modify the behavior of] the functions [*TSF Self-Test (FPT_TST.1(1))*] to [*the Security Administrator*].

Application Note: The phrase 'modify the behavior of' refers to the ability of the Security Administrator to specify the frequency for the periodic execution of the TSF non-cryptographic self-tests.

FMT_MOF.1(2) - Management of security functions behavior (Cryptographic Self-test)

FMT_MOF.1.1(2) - The TSF shall restrict the ability to [enable, disable] the functions [*TSF Self-Test (FPT_TST.1(2))*] to [*the Cryptographic Administrator*].

Application Note: This requirement describes the ability of the Cryptographic Administrator to determine whether or not the cryptographic self-tests are executed after the generation of each key.

FMT_MOF.1(3) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(3) - The TSF shall restrict the ability to [enable, disable, determine and modify the behavior of] the functions [*Security Audit (FAU_SAR)*] to [*an Administrator*].

Application Note: This requirement describes the ability of all administrators to read, search and sort the data in the audit trail.

FMT_MOF.1(4) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(4) - The TSF shall restrict the ability to [enable, disable, determine and modify the behavior of] the functions [

a) *Security Audit Analysis (FAU_SAA); and*

b) *Security Audit (FAU_SEL)*]

to [*the Security Administrator*].

Application Note: This requirement describes the ability of the Security Administrator to specify whether or not an auditable event is included or excluded from the audit trail (based on identified criteria as listed in FAU_SEL) as well as the ability of the Security Administrator to define rules which govern the generation of alarms to indicate a potential violation of the TSP (FAU_SAA).

FMT_MOF.1(5) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(5) - The TSF shall restrict the ability to [enable, disable] the functions [*Security Alarms (FAU_ARP)*] to [*the Security Administrator*].

Application Note: This requirement describes the ability of the Security Administrator to specify whether or not an alarm generates an audible signal.

FMT_MOF.1(6) Management of security functions behavior (available TOE-services for unauthenticated users)

FMT_MOF.1.1(6) - The TSF shall restrict the ability to [enable, disable] the functions [*for an IP-based network stack: ICMP*] to [*the Security Administrator*].

Application Note: This requirement describes the ability of the Security Administrator to specify whether or not the TOE will respond to ICMP requests from unauthenticated users.

FMT_MOF.1(7) Management of security functions behavior (quota mechanism)

FMT_MOF.1.1(7) - The TSF shall restrict the ability to [determine the behavior of] the functions [

a) *Controlled connection-oriented resource allocation (FRU_RSA.1(2));*

b) *an administrator-specified network identifier;*

c) *set of administrator-specified network identifiers;*

d) *administrator-specified period of time*] to [*the Security Administrator*].

> *Application Note: This requirement describes the ability of the Security Administrator to specify the parameters which apply to quotas for connection oriented resources; namely the network identifiers and the time period over which the quotas apply. The actual specification of the quota is covered by FMT_MTD.2(2).*

FMT_MOF.1(8) Management of security functions behavior (cryptographic self-test frequency)

FMT_MOF.1.1(8) – The TSF shall restrict the ability to [modify the behaviour of] the functions [*cryptographic self-tests (FPT_TST.1(2)*] to [*the Security Administrator*].

> *Application Note: The Security Administrator is responsible for setting the frequency for the periodic execution of the cryptographic self-tests. The frequency may not be less than once per day.*

FMT_MOF.1(9) Management of security functions behavior (audit storage exhaustion)

FMT_MOF.1.1(9) – The TSF shall restrict the ability to [modify the behaviour of] the functions [*action taken by the TOE in the event of audit storage exhaustion*] to [*the Security Administrator*].

> *Application Note: For this requirement, the phrase 'modify the behavior of' refers to the ability of the Security Administrator to specify the action to be taken in the event of audit storage exhaustion. Audit Storage exhaustion is defined as the percentage of available audit storage usage which generates an alarm as described by FAU_SAA.1.*

FMT_MOF.1(10) Management of security functions behavior (session termination)

FMT_MOF.1.1(10) – The TSF shall restrict the ability to [modify the behaviour of] the functions [*session termination (FTA_SSL.1, FTA_SSL.3*] to [*the Security Administrator*].

> *Application Note: For this requirement, the phrase 'modify the behavior of' refers to the ability of the Security Administrator to specify a period of inactivity after which the inactive session of an administrator or an authenticated proxy user is terminated by the TOE.*

FMT_MOF.1(11) Management of security functions behavior (alarm acknowledgement)

FMT_MOF.1.1(11) – The TSF shall restrict the ability to [modify the behaviour of] the functions [alarms] to [*an Administrator*].

> *Application Note: For this requirement, the phrase 'modify the behavior of' refers to the ability of all Administrators to acknowledge alarms.*

FMT_MOF.1(12) Management of security functions behavior (self-tests)

FMT_MOF.1.1(12) – The TSF shall restrict the ability to [modify the behaviour of] the functions [*cryptographic and non-cryptographic self-tests*] to [*an Administrator*].

> *Application Note: For this requirement, the phrase 'modify the behavior of' refers to the ability of all Administrators to manually execute the cryptographic and non-cryptographic self-tests.*

FMT_MOF.1(13) Management of security functions behavior (IDS sensor)

FMT_MOF.1.1(13) – **Refinement:** The TSF shall restrict the ability to [modify the behavior of] the functions [*Sensor data collection and review (IDS_COL_EXP.1)*] to [*an Administrators*].

> *Application Note: For this requirement, the phrase 'modify the behavior of' refers to the ability of all Administrators to manage the IDS functions of the TOE.*

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 – The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP, AUTHENTICATED INFORMATION FLOW SFP, UNAUTHENTICATED TOE SERVICES SFP, VPN SFP*] to restrict the ability to [[*manipulate*]] the security attributes [*referenced in the indicated polices*] to [*the Security Administrator*].

> *Application Note: The term "manipulate" is used to indicate that the security attributes specified in the iterations of FDP_IFF.1.1 may be used to create additional "attributes" that can be used in specifying information flow policy rules (for example, a set of network identifiers that can be used as a "group").*

FMT_MSA.2 Secure security attributes

FMT_MSA.2.1 – The TSF shall ensure that only secure values are accepted for security attributes.

FMT_MSA.3(1) Static attribute initialization (ruleset)

FMT_MSA.3.1(1) – **Refinement**: The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP, AUTHENTICATED INFORMATION FLOW SFP and, VPN SFP*] to provide [restrictive] default values for the information flow policy ruleset that is used to enforce the SFP.

FMT_MSA.3.2(1) - The TSF shall allow the [*Security Administrator*] to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3(2) Static attribute initialization (services)

FMT_MSA.3.1(2) – **Refinement**: The TSF shall enforce the [*UNAUTHENTICATED TOE SERVICES SFP*] to provide [restrictive] default values for the set of TOE services available to unauthenticated users.

FMT_MSA.3.2(2) - The TSF shall allow the [*Security Administrator*] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1(1) Management of TSF data (deletion of audit data)

FMT_MTD.1.1(1) – The TSF shall restrict the ability to [delete] the [*audit data*] to [*the Audit Administrator*].

FMT_MTD.1(2) Management of TSF data (cryptographic TSF data)

FMT_MTD.1.1(2) – The TSF shall restrict the ability to [modify] the [*cryptographic security data*] to [*the Cryptographic Administrator*].

Application Note: This requirement describes the ability of the Cryptographic Administrator to load keys using a FortiUSB token.

FMT_MTD.1(3) Management of TSF data (time TSF data)

FMT_MTD.1.1(3) – The TSF shall restrict the ability to [[*set*]] the [*time and date used to form the time stamps in FPT_STM.1*] to [*the Security Administrator*].

FMT_MTD.1(4) Management of TSF data (Information flow policy ruleset)

FMT_MTD.1.1(4) – The TSF shall restrict the ability to [query, modify, delete, [*create*]] the [*information flow policy rules*] to [*the Security Administrator*].

FMT_MTD.1(5) Management of TSF data (user accounts)

FMT_MTD.1.1(5) – The TSF shall restrict the ability to [modify, [*create*]] the [*user accounts*] to [*the Security Administrator*].

FMT_MTD.1(6) Management of TSF data (TOE banner)

FMT_MTD.1.1(6) – The TSF shall restrict the ability to [modify] the [*TOE banner*] to [the *Security Administrator*].

FMT_MTD.1(7) Management of TSF data (AV and IPS signatures)

FMT_MTD.1.1(7) – The TSF shall restrict the ability to [modify] the [*AV and IPS signatures*] to [*the Security Administrator and the Fortinet FortiGuard Distribution Server*].

FMT_MTD.1(8) Management of TSF data (VPN policy ruleset)

FMT_MTD.1.1(8) - The TSF shall restrict the ability to [query, modify, delete, [*create*]] the [*VPN Policy rules*] to [*the Security Administrator*].

FMT_MTD.1(9) Management of TSF data (IDS sensor data)

FMT_MTD.1.1(9) - **Refinement:** The TSF shall restrict the ability to query Sensor data to [*the Security Administrator*].

FMT_MTD.2(1) Management of limits on TSF data (transport-layer quotas)

FMT_MTD.2.1(1) -The TSF shall restrict the specification of the limits for [*quotas on transport-layer connections*] to [*the Security Administrator*].

FMT_MTD.2.2(1) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [*take Security Administrator-specified action, one of clear-session, drop, drop-session, pass, pass-session, reset, reset-client, or reset-server*].

FMT_MTD.2(2) Management of limits on TSF data (controlled connection-oriented quotas)

FMT_MTD.2.1(2) -The TSF shall restrict the specification of the limits for [*quotas on controlled connection-oriented resources*] to [*the Security Administrator*].

FMT_MTD.2.2(2) -The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [*take Security Administrator-specified action, one of clear-session, drop, drop-session, pass, pass-session, reset, reset-client, or reset-server*].

FMT_REV.1 Revocation

FMT_REV.1.1 – **Refinement**: The TSF shall restrict the ability to revoke security attributes associated with the [users, information flow policy ruleset, services available to unauthenticated users] within the TSC to [*the Security Administrator*].

> *Application Note: The selection "information flow policy ruleset" is an object. The selection "services available to unauthenticated users" is a subject.*

FMT_REV.1.2 – **Refinement**: The TSF shall immediately enforce the: [

a) *revocation of a user's role (Security Administrator, Cryptographic Administrator, Audit Administrator);*

b) *revocation of a user's ability to use an authenticated proxy;*

c) *changes to the information flow policy ruleset when applied;*

d) *disabling of a service available to unauthenticated users; and*

e) *changes to the set of security associations with peer TOEs*].

FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 - **Refinement:** The TSF shall maintain the roles: [

a) Security Administrator, who will also perform the functions allocated to the Sensor Administrator in the IDSS PP;

b) Cryptographic Administrator (i.e., users authorized to perform cryptographic initialization and management functions);

c) Audit Administrator;

d) Authenticated Proxy User; and

e) VPN User].

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

FMT_SMR.2.3 – **Refinement**: The TSF shall ensure that the conditions [

a) *all administrator roles shall be able to administer the TOE via the Local Console;*

b) *all administrator roles shall be able to administer the TOE via the Network Web-Based GUI and Network CLI;*

c) *all administrator roles are distinct; that is, there shall be no overlap of operations performed by each role, with the following exceptions:*

- *all administrators can review the audit trail; and*
- *all administrators can invoke the self-tests*]

are satisfied.

5.1.8 Protection of the TOE Security Functions (FPT)

FPT_AMT.1 Abstract Machine Testing

FPT_AMT.1.1 – **Refinement**: The TSF shall run a suite of tests [during initial start-up, periodically during normal operation as specified by the Security Administrator, at the request of an authorized user] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 – The TSF shall preserve a secure state when the following types of failures occur: [*failure of a unit in a FortiGate cluster is detected*].

Application Note: The FPT_FLS.1 requirement is only implemented in the High Availability configuration of the TOE. The FPT_FLS.1 requirement is not specified in either the FW PP MR or the TFFW PP MR.

FPT_ITA.1 Inter-TSF availability within a defined availability metric

FPT_ITA.1.1 The TSF shall ensure the availability of [*audit and Sensor data*] provided to a remote trusted IT product within [*one minute of receipt of request for the data*] given the following conditions [*audit or Sensor data is available for transmission*].

FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1 The TSF shall protect all TSF data transmitted from the TSF to a remote trusted IT product from unauthorised disclosure during transmission.

FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1 - The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: [*SHA-1*].

FPT_ITI.1.2 – **Refinement:** The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform [*drop and log the packet*] if modifications are detected by the receiving device.

FPT_RCV.1 Manual Recovery

FPT_RCV.1.1 - After [*a failure or service discontinuity*] the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

Application Note: In the terminology used by the TOE, the maintenance mode of this requirement is termed the "FIPS-CC Error Mode".

FPT_RPL.1 Replay detection

FPT_RPL.1.1 – The TSF shall detect replay for the following entities: [*TSF data and security attributes*].

FPT_RPL.1.2 – The TSF shall perform [

- *reject data; and*
- *audit event*]

when replay is detected.

FPT_RVM.1 Non-bypassability of the TSP

FPT_RVM.1.1 - The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

FPT_SEP.2 SFP domain separation

FPT_SEP.2.1 – The unisolated portion of the TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.2.2 – The TSF shall enforce separation between the security domains of subjects in the TSC.

FPT_SEP.2.3 – **Refinement**: The TSF shall maintain the part of the TSF related to [*cryptography*] in an address space for its own execution that protects it from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to the cryptographic functionality.

FPT_STM.1 Reliable time stamps

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

FPT_TST.1(1) TSF testing (with cryptographic integrity verification)

FPT_TST.1.1(1) – **Refinement**: The TSF shall run a suite of self-tests [during initial start-up, periodically during normal operation as specified by the Security Administrator and at the request of an Administrator] to demonstrate the correct operation of [*the hardware portions of the TSF*].

FPT_TST.1.2(1) – **Refinement**: The TSF shall provide an Administrator with the capability to use a TSF-provided cryptographic function to verify the integrity of [*all TSF data except the following: audit data, IDS sensor data*].

FPT_TST.1.3(1) – **Refinement**: The TSF shall provide an Administrator with the capability to use a TSF-provided cryptographic function to verify the integrity of stored TSF executable code.

FPT_TST.1(2) TSF testing (Cryptographic self-test)

FPT_TST.1.1(2) – **Refinement**: The TSF shall run the suite of self-tests provided by the FIPS 140-2 cryptographic module [during initial start-up <u>(power on)</u>, at the request of an administrator, periodically during normal operation, at the conditions [

a) *The periodic execution shall be at a Security Administrator-specified interval not less than at least once a day;*

b) *The TSF shall be able to run the suite of self-tests provided by the FIPS 140-2 cryptographic module immediately after the generation of a key;*

c) *invocation of self-test shall be restricted to an administrator*]]

to demonstrate the correct operation of [[*the cryptographic components of the TSF*]].

FPT_TST.1.2(2) – **Refinement**: The TSF shall provide an administrator with the capability to use a TSF-provided cryptographic function to verify the integrity of [*all TSF data except the following: audit data, IDS sensor data*].

FPT_TST.1.3(2) – **Refinement**: The TSF shall provide an administrator with the capability to use a TSF-provided cryptographic function to verify the integrity of stored TSF executable code.

Application Note: The FPT_TST.1.2(2) and FPT_TST.1.3(2) components are redundant with FPT_TST.1.2(1) and FPT_TST.1.3(1) respectively, and are included only to complete the definition of the function.

5.1.9 Fault tolerance (FRU_FLT)

FRU_FLT.1 Degraded fault tolerance

FRU_FLT.1.1 – The TSF shall ensure the operation of [*TCP load balancing for packets belonging to stateful sessions and configuration synchronization*] when the following failures occur: [*failure of a unit in a FortiGate cluster is detected*].

Application Note: The FRU_FLT.1 requirement is only implemented in the High Availability configuration of the TOE. The FRU_FLT.1 requirement is not specified in either the FW PP MR or the TFFW PP MR.

5.1.10 Resource allocation (FRU_RSA)

FRU_RSA.1(1) - Maximum quotas (transport-layer quotas)

FRU_RSA.1.1(1) – **Refinement:** The TSF shall enforce maximum quotas of the following resources: [*transport-layer representation*] that [a source subject identifier] can use [over a specified period of tim*e*].

FRU_RSA.1(2) - Maximum quotas (controlled connection-oriented quotas)

FRU_RSA.1.1(2) – **Refinement:** The TSF shall enforce Security Administrator-specified maximum quotas of the following resources: [*TCP Session, which is a controlled connection-oriented resource*] that [users associated with an Security Administrator-specified network identifier and a set of administrator-specified network identifiers] can use [over a Security Administrator-specified period of time].

5.1.11 TOE Access (FTA)

FTA_SSL.1 TSF-initiated session locking

FTA_SSL.1.1 – **Refinement:** The TSF shall lock a Local Console interactive session after [*a Security Administrator-specified time period of inactivity*] by:

a) clearing or overwriting display devices, making the current contents unreadable;

b) disabling any activity of the user’s data access/display devices other than unlocking the session.

FTA_SSL.1.2 – The TSF shall require the following events to occur prior to unlocking the session: [*the user must re-identify and re-authenticate*].

Application Note: In order to prevent the problems associated with a locked local console session and no alternate administrative access to the TOE short of a cold reboot, the TOE implements this requirement by terminating the administrators session after a Security Administrator-specified period of inactivity. To re-establish the session, the administrator is required to re-identify and re-authenticate. This *implementation is more restrictive than the requirement in the MR PPs which only requires re-authentication in order to unlock the session.*

FTA_SSL.2 User-initiated locking

FTA_SSL.2.1 – **Refinement:** The TSF shall allow user-initiated termination of the user's own Local Console interactive session by:

a) clearing or overwriting display devices, making the current contents unreadable;

b) disabling any activity of the user's data access/display devices other than re-establishing the session.

Application Note: The TOE meets the intent of session locking by imposing a more stringent session termination requirement.

FTA_SSL.2.2 – **Refinement:** The TSF shall require the following events to occur prior to re-establishing the Local Console interactive session: [*the user must re-identify and re-authenticate*].

Application Note: The TOE implements this requirement by requiring that an administrator terminate his Local Console interactive session. Then in order to re-establish the session, the administrator is required to both re-identify and re-authenticate, thus making the TOE's implementation more restrictive than required by the MR PPs.

FTA_SSL.3 TSF-initiated termination

FTA_SSL.3.1 - **Refinement:** The TSF shall terminate an authenticated Proxy User, VPN User, Network Web-Based GUI, or Network CLI session after a [*Security Administrator-configurable time interval of session inactivity*].

FTA_TAB.1 Default TOE access banners

FTA_TAB.1.1 – **Refinement**: Before establishing a user session that requires authentication or before establishing an administrative session, the TSF shall display only a Security Administrator-specified advisory notice and consent warning message regarding unauthorized use of the TOE.

FTA_TSE.1 TOE session establishment

FTA_TSE.1.1 - **Refinement:** The TSF shall be able to deny establishment of an authorized Proxy User session, VPN User session, Network Web-Based GUI session, and Network CLI session based on [*interface and IP address, time, and day*].

5.1.12 Trusted Path/Channels (FTP)

FTP_ITC.1(1) Inter-TSF trusted channel (Prevention of Disclosure)

FTP_ITC.1.1(1) – **Refinement**: The TSF shall use encryption to provide a trusted communication channel between itself and Fortinet's FortiGuard Distribution Server that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure.

FTP_ITC.1.2(1) - **Refinement:** The TSF shall permit [the TSF, or Fortinet's FortiGuard Distribution Server] to initiate communication via the trusted channel.

FTP_ITC.1.3(1) – **Refinement**: The TSF shall initiate communication via the trusted channel for [*all authentication functions, and High Availability Cluster communication*].

FTP_ITC.1(2) Inter-TSF trusted channel (Detection of Modification)

FTP_ITC.1.1(2) – **Refinement**: The TSF shall use a cryptographic signature to provide a trusted communication channel between itself and Fortinet's FortiGuard Distribution Server that is logically distinct from other communication channels and provides assured identification of its end points and detection of the modification of data.

FTP_ITC.1.2(2) – **Refinement:** The TSF shall permit [the TSF, or Fortinet's FortiGuard Distribution Server] to initiate communication via the trusted channel.

FTP_ITC.1.3(2) – **Refinement**: The TSF shall initiate communication via the trusted channel for [*all authentication functions and High Availability Cluster communication*].

FTP_TRP.1(1) Trusted path (Prevention of Disclosure)

FTP_TRP.1.1(1) - **Refinement:** The TSF shall provide an encrypted communication path between itself and administrators using the Network Web-Based GUI and Network CLI, VPN Users, and authenticated proxy 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.

FTP_TRP.1.2(1) - **Refinement:** The TSF shall permit proxy users, VPN Users and administrators using the Network Web-Based GUI and Network CLI to initiate communication via the trusted path.

FTP_TRP.1.3(1) – **Refinement:** The TSF shall require the use of the trusted path for proxy user, VPN User, and administrator authentication and all remote administration action.

FTP_TRP.1(2) Trusted path (Detection of Modification)

FTP_TRP.1.1(2) - **Refinement:** The TSF shall use a cryptographic signature to provide a communication path between itself and administrators using the Network Web-Based GUI and Network CLI, VPN Users and authenticated proxy users that is logically distinct from other communication paths and provides assured identification of its end points and detection of the modification of data.

FTP_TRP.1.2(2) - **Refinement:** The TSF shall permit proxy users, VPN Users and administrators using the Network Web-Based GUI and Network CLI to initiate communication via the trusted path.

FTP_TRP.1.3(2) **– Refinement:** The TSF shall require the use of the trusted path for proxy user, VPN User, and administrator authentication and all remote administration actions.

5.1.13 Intrusion Detection System Explicit Requirements

IDS_COL_EXP.1 Sensor Data Collection

IDS_COL_EXP.1.1 **– Refinement**: The Sensor shall be able to collect the following events from the targeted IT System resource(s):

a) [network traffic].

IDS_COL_EXP.1.2 - At a minimum, the Sensor shall collect the following information:

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

b) The additional information specified in the Details column of Table 8 - Sensor Events.

Component	Event	Details
IDS_COL_EXP.1	Network traffic	Protocol, source address, destination address

Table 8 - Sensor Events

IDS_RDR_EXP.1 Restricted Data Review

IDS_RDR_EXP.1.1 The Sensor shall provide [*Administrators*] with the capability to read [*all entries*] from the Sensor data.

IDS_RDR_EXP.1.2 The Sensor shall provide the Sensor data in a manner suitable for the user to interpret the information.

IDS_RDR_EXP.1.3 The Sensor shall prohibit all users read access to the Sensor data, except those users that have been granted explicit read-access.

IDS_STG_EXP.1 Guarantee of Sensor Data Availability

IDS_STG_EXP.1.1 The Sensor shall protect the stored Sensor data from unauthorised deletion.

IDS_STG_EXP.1.2 The Sensor shall protect the stored Sensor data from modification.

IDS_STG_EXP.1.3 The Sensor shall ensure that [*the Security Administrator's selection of all or the most recent*] Sensor data will be maintained when the following conditions occur: [Sensor data storage exhaustion].

IDS_STG_EXP.2 Prevention of Sensor data loss

IDS_STG_EXP.2.1 - **Refinement:** The Sensor shall provide the Security Administrator the capability to select one of the following actions: [

a) prevent events that would cause Sensor data recording, except those events taken caused by the authorised user with special rights; or

b) overwrite the oldest stored Sensor data]

and send an alarm if the storage capacity has been reached.

5.1.14 Strength of Function Requirement

FortiGate Unified Threat Management Solutions provide a level of protection that is appropriate against threat agents whose attack potential is low, in IT environments that require that information flows be controlled and restricted among network nodes where the FortiGate unit can be appropriately protected from physical attacks. The FortiGate unit's management console must be controlled to restrict access to only authorized administrators. It is expected that the FortiGate units will be protected to the extent necessary to ensure that they remain connected to the networks they protect. The minimum strength of function, SOF-Basic, is consistent with those requirements.

The password rules will ensure that the implementation has the required strength.

5.2 SECURITY REQUIREMENTS FOR THE IT ENVIRONMENT

This Security Target provides functional requirements for the IT Environment. The IT environment includes authorized IT entities (e.g., a certificate authority server) and any IT entities that are used by administrators to remotely administer the TOE. These requirements consist of functional components from Part 2 of the CC.

FTP_ITC.1(3)(ENV) Inter-TSF trusted channel (Prevention of Disclosure)

FTP_ITC.1.1(3)(ENV) – The IT Environment shall provide a trusted communication channel between itself and the TSF that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure.

FTP_ITC.1.2(3)(ENV) – The IT Environment shall permit [the TSF, or the IT Environment] to initiate communication via the trusted channel.

FTP_ITC.1.3(3)(ENV) – The TSF shall initiate communication via the trusted channel for [*FortiGuard Distribution Server authentication and communication with the FortiGuard Distribution Server*].

FTP_ITC.1(4)(ENV) Inter-TSF trusted channel (Detection of Modification)

FTP_ITC.1.1(4)(ENV) – The IT Environment shall provide an encrypted communication channel between itself and the TSF that is logically distinct from other communication channels and provides assured identification of its end points and detection of the modification of data.

FTP_ITC.1.2(4)(ENV) – The IT Environment shall permit [the TSF, or the IT Environment] to initiate communication via the trusted channel.

FTP_ITC.1.3(4)(ENV) – The TSF shall initiate communication via the trusted channel for [*FortiGuard Distribution Server authentication and communication with the FortiGuard Distribution Server*].

FTP_TRP.1(3)(ENV) Trusted path (Prevention of Disclosure)

FTP_TRP.1.1(3)(ENV) - The IT Environment shall provide an encrypted communication path between itself and [selection: the TSF] that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure.

FTP_TRP.1.2(3)(ENV) - The IT Environment shall permit [Network Web-Based GUI administrators, Network CLI administrators, and Network Users of the TSF] to initiate communication to the TSF via the trusted path.

FTP_TRP.1.3(3)(ENV) – The IT Environment shall initiate the use of the trusted path for [Network User authentication, administrator authentication and all administrative use of the Network Web-Based GUI and Network CLI].

FTP_TRP.1(4)(ENV) Trusted path (Detection of Modification)

FTP_TRP.1.1(4)(ENV) - The IT Environment shall provide an encrypted communication path between itself and [selection: *the TSF*] that is logically distinct from other communication paths and provides assured identification of its end points and detection of the modification of data.

FTP_TRP.1.2(4)(ENV) - The IT Environment shall permit [Network Web-Based GUI administrators, Network CLI administrators, and Network Users of the TSF] to initiate communication to the TSF via the trusted path.

FTP_TRP.1.3(4)(ENV) – The IT Environment shall initiate the use of the trusted path for [Proxy User authentication, administrator authentication and administrative use of the Network Web-Based GUI and Network CLI]**.**

5.3 TOE SECURITY ASSURANCE REQUIREMENTS

The TOE assurance requirements for this ST consist of the requirements corresponding to the EAL4 level of assurance, as defined in the CC Part 3, augmented by the inclusion of Systematic Flaw Remediation (ALC_FLR.3).

The assurance requirements are summarized in the Table 9 below.

Assurance Class	**Assurance Components**	
	Identifier	**Name**
Configuration management	ACM_AUT.1	Partial CM automation
	ACM_CAP.4	Generation support and acceptance procedures
	ACM_SCP.2	Problem tracking CM coverage
Delivery and operation	ADO_DEL.2	Detection of modification
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.2	Fully defined external interfaces
	ADV_HLD.2	Security enforcing high-level design
	ADV_IMP.1	Subset of the implementation of the TSF
	ADV_LLD.1	Descriptive low-level design
	ADV_RCR.1	Informal correspondence demonstration
	ADV_SPM.1	Informal TOE security policy model
Guidance documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance
Life cycle support	ALC_DVS.1	Identification of security measures
	ALC_FLR.3	Systematic flaw remediation

Assurance Class	Assurance Components	
	Identifier	Name
Life cycle Support	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.1	Well-defined development tools
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: high-level design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability assessment	AVA_MSU.2	Validation of analysis
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.2	Independent vulnerability analysis

Table 9 - Assurance Requirements

6 TOE SUMMARY SPECIFICATION

This section provides a description of the security functions and assurance measures of the TOE that meet the TOE security requirements defined in Section 5. The functions and functional requirements are cross-referenced in Table 20 - Mapping of Security Functions to Security Functional Requirements from CC Part 2. The assurance measures and assurance requirements are cross-referenced in Table 23.

6.1 TOE SECURITY FUNCTIONS

6.1.1 Overview

The TOE security functions that were introduced in Section 2.4 are further elaborated in this section. The major functions (e.g., audit) are decomposed to more clearly define their functionality.

6.1.2 Identification and Authentication

F.I&A	In order to protect the TOE data and services, the TOE requires identification and authentication for all administrative access and network user access to specific services. Identification and authentication is always enforced on the serial interface (local console). On the network interfaces identification and authentication is enforced for all administrator access, specific services, and VPN users. The identification and authentication mechanism is a username and password combination. The accounts are created by the Security Administrator over the serial or network interfaces. The TOE also requires identification and authentication for high availability units in a cluster. Each unit has a unique identifier (username) and a shared password. The USB interface does not directly require identification and authentication since the Cryptographic Administrator must be authenticated to load keys from the USB token. However the TOE will on;y recognize FortiUSB tokens, restricted by the vendor ID of the token.

6.1.3 Administration

F.ADMIN	Administrative access to the TOE is restricted to authorised administrators and is controlled through a set of pre-defined roles (Security Administrator, Audit Administrator and Crypto

Administrator). The roles permit specific types of administrative activities to be performed.

All Administrators can read audit log data, acknowledge alarms and execute the self-tests. In addition the Audit Administrator can delete audit records and the Crypto Administrator can modify the cryptographic security data. The Security Administrator can not delete audit records or modify cryptographic security data but can perform all other TOE administration functions.

The TOE allows both local and remote administration. Local administration is performed using the Local Console. Remote administration is performed using the Network Web-Based GUI or Network CLI interfaces.

The TOE immediately enforces the revocation of an administrative role.

6.1.4 Information Flow Control

F.IFC

The TOE operates in accordance with four information flow security functional policies.

The UNAUTHENTICATED INFORMATION FLOW SFP allows unauthenticated users to pass information through the TOE, with firewall mediation according to the firewall rules defined by the Security Administrator.

The AUTHENTICATED INFORMATION FLOW SFP allows authenticated users to pass information through the TOE, with firewall mediation according to the firewall rules defined by the Security Administrator.

The UNAUTHENTICATED TOE SERVICES POLICY SFP allows unauthenticated users to use TOE services by sending packets to the TOE and receiving responses back from it.

The VPN SFP allows authenticated users to send and receive information protected by trusted paths and channels to/from the TOE.

The security functional policies are implemented as firewall rules. The rules that implement the SFPs have restrictive default values and by default no information is allowed to flow, and TOE services are not available to unauthenticated users. Regardless of firewall rules, packets which include specific parameters as specified by the security functional requirements which define the security functional policies are never permitted to pass through the TOE. Modification of the rules is restricted to the Security Administrator and the Security Administrator can also specify alternative initial values to override the default values. The TOE allows the Security Administrator to view all information flows allowed by the information flow policy rules before the rules are applied.

The TOE mediates all information flows which pass through it. For information to pass through the TOE, it must match one of the Security Administrator specified firewall rules which permit the information flow.

The TOE ensures that all information flows provided to the TOE by external entities for transfer to other entities are subjected to the defined firewall rules and conform to them before they are allowed to proceed toward the destination entity.

The TSF immediately enforces revocation of a user's permission to use the information flow and also immediately enforces changes to the information flow policy rules when applied. The TOE also immediately enforces the disabling of a service which was available to an unauthenticated user.

The TOE ensures that no information from previously processed information flows is transferred to subsequent information flows. This applies both to information that is input to the TOE from an external source and to information (e.g., padding bits) that might be added by the TOE during processing of the information from the external source.

The TOE follows a sequence of ordered steps in order to decide whether or not a requested information flow is allowed to proceed.

The very first processing step performed by the FortiGate on incoming information is an inspection for IPS anomalies which target the TOE directly. Examples of IPS anomalies include syn floods, ping of death, source routing and port scans.

If the incoming information flow is not blocked by the inspection for IPS anomalies, it is next processed against the firewall policy rules and authentication requirements.

If the incoming information flow is allowed by the firewall policy rules (using the first match algorithm) and if any required authentication has been completed successfully, the incoming information flow may be subject to additional restrictions based on any Protection Profile which is associated with the firewall policy rule which allowed the information flow.

Protection Profiles are used to define additional information flow restrictions which may be based on any or all of the following types of information:

- Scheduling
- SMTP commands
- SMTP MIME types
- FTP subcommands
- HTTP request methods
- Virus signatures
- IPS signature matching

Only the Security Administrator may create, modify or delete a Protection Profile. Additionally, only the Security Administrator may associate a protection profile with a firewall policy rule.

The specific steps used by the TOE to process incoming information flows and enforce its security policy are summarized below:

1) Local IPS Anomaly protection (kernel level)

2) Firewall flow control policy enforcement

    a. First matched policy must explictly allow traffic to flow.

3) Authenticated flow control policies

    a. If configured for flow-control policy, successful authentication required for traffic to flow

4) Protection Profile services (if explicitly enabled)

    a. Scheduling

        i. If scheduling is enabled, time period must be explicitly allowed

    b. SMTP Commands

        i. All SMTP commands permitted unless explicitly denied

c. MIME Types

   i. All MIME types permitted unless explicitly denied

d. FTP Sub-Commands

   i. All FTP sub-commands permitted unless explicitly denied

e. HTTP Request Methods

   i. All HTTP request methods permitted unless explicitly denied

f. Virus protection

   i. If content is matched against an AV signature, the configured action is performed.

g. IPS Signature matching

   i. If the nature of the connection or content is matched against an IPS signature, the configured action is performed.

It must be noted that traffic is only passed to the next enforcement method if previous enforcement methods explicitly allow the traffic.

After all security policy enforcement is performed and no further security scrutiny is required, the packet data is forwarded to the network host as determined by the configuration of the egress interface and/or static route.

6.1.5 Trusted Channel/Path

F.TRSTCOMM

The TOE provides trusted paths and trusted channels, protected by encryption to guard against disclosure and protected by cryptographic signature to detect modifications. The trusted paths and trusted channels are logically distinct from other communication paths and provide assured identification of their end points.

The trusted paths are used to protect remote Administrator authentication, all remote administrator actions, Proxy User authentication, VPN user authentication, and all VPN user actions. Remote administration sessions apply to the Network Web-Based GUI and Network CLI.

The Network CLI uses SSH version 2 and only supports the use of the following FIPS PUB 140-2 approved algorithms to encrypt all authentication and communications data:

- 3DES
- AES
- HMAC-SHA1

Only administrator accounts stored in the local authentication database are permitted to authenticated (i.e. root authentication and proxy user accounts cannot be used).

By default, SSH connections to the TOE are disabled and must be explicitly enabled before an administrator can use the Network CLI interface.

The TOE supports the use of fingerprints as defined in RFC 4251, in that it provides "[a method] for verifying the correctness of host keys, e.g., a hexadecimal fingerprint derived from the SHA-1 hash [FIPS-180-2] of the public key." When a Network CLI connection is first established, the TOE transmits a 2048-bit RSA public key to the connecting client which can be used to validate the identity of the TOE. Each FortiGate unit is delivered with a factory installed 2048-bit RSA public/private key pair. However the Cryptographic Administrator may use a FortiUSB token to replace this key pair with another key pair which he has generated or obtained from an alternate source. An administrator attempting to establish a Network CLI connection with the TOE can choose to allow or disconnect the connection based on the aforementioned fingerprint. If the administrator chooses to continue, the identity of the TOE is considered to be valid and the TOE prompts the connecting client for user and password credentials.

The Network Web-Based GUI uses the HTTPS protocol for secure administrator communications. With respect to the TOE implementation of HTTPS, TLS version 1.0 (RFC 2246) is used to encrypt and authenticate administration sessions between the remote browser and TOE. The TOE supports ciphersuites; TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (RFC 2246) and TLS_DHE_RSA_WITH_AES_128_CBC_SHA (RFC 3268).

These ciphersuites mean that the keying material is determined when the session is established through a Diffie-Hellman (DH) exchange which consists of:

- Server sends 2048-bit RSA public certificate
- Server generates, signs (RSA PKCS#1) and sends DH parameters and DH public value
- Client generates and sends DH public value. The keying material is then used to encrypt/decrypt (AES128 and 3DES) and authenticate (HMAC-SHA1) the data exchange.

By default, HTTPS connections to the TOE are disabled and must be explicitly enabled before an Administrator may use the Network Web-Based GUI.

When a connection is first established, the server presents the 2048-bit RSA certificate to the connecting web browser. The administrator can examine the certificate to validate the identity of the TOE and then choose to continue with the connection if the certificate conforms to the expected values. Only after the certificate has been explicity accepted as valid will the administrator be presented with the login page, where the user and password credentials can be submitted for authentication. As with the Network CLI, only local administrator account credentials can be used to successfully authenticate to the TOE via the Network Web-Based GUI.

The trusted channels provide communication between the TOE and the FortiGuard Distribution Server that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure. The FortiGuard Distribution Server is used to obtain updates to the IPS (attack) signatures and virus definitions.

The TOE must be explicitly configured to obtain AV and IPS signature updates from the FortiGuard Distribution Server. At this time, a UDP port must be specified. This UDP port is used by the FortiGuard Distribution Network to advise the TOE that signature updates are available for download. No secure channel is established at this time.

When the TOE becomes aware that an update is available it will (if so configured) initiate a trusted channel connection to the FortiGuard Distribution Server using the factory-loaded 2048-bit RSA certificate which is issued by the Fortinet CA. This certificate cannot be modified by any TOE administrator.

Alternatively, the AV and IPS signature updates can be downloaded manually by the TOE administrator or on a schedule (hourly/daily/weekly). The trusted channel described in the previous paragraph is also used for these manual/schedules updates.

As noted in Section 5 under FAV_ACT_EXP.1 and FIP_ACT_EXP.1, AV and IPS signature updates consist of updates to both the signatures data files and the AV and IPS processing engines. The TOE provides specific guidance to administrators which notes that in the evaluated configuration of the TOE, only updates to the signatures data files may be applied.

6.1.6 Encryption

F.CRYPTO

The TOE uses FIPS-approved cryptography that has been implemented in FIPS 140-2 validated cryptographic modules. The FIPS-validated cryptographic modules implemented in the TSF meet Security Level 2 overall and meet Security Level 3 for the following: cryptographic module ports and interfaces; roles, services and authentication; cryptographic key management, and design assurance. The proprietary FortiASIC™ chip is a hardware component which forms part of the validated cryptographic modules used in the TOE. Cryptographic key destruction by the TOE meets the key zeroization requirements of Key Management Security Level 3 from FIPS PUB 140-2. The TOE only stores keys in memory, either in RAM or Flash memory. Keys are destroyed by overwriting the key storage area with an alternating pattern at least once.

The TSF provides a cryptographic function that an Administrator may use to verify the integrity of all TSF data except the audit data and to verify the integrity of the TSF executable code. These self-tests are executed on initial start-up or at the request of an Administrator.

The TOE provides a USB interface which may be used by the Cryptographic Administrator to load private keys for the rDSA asymmetric algorithm from a FortiUSB token.

The 2048-bit RSA certificate used by the Network Web-Based GUI can be replaced by certificates trusted by the crypto administrator. These keys/certificates are to be placed on the FortiUSB token and the load operation can be executed via a Network CLI or Network Web-Based GUI administrator session.

6.1.7 Audit

F.AUDIT

The TOE creates audit records for administrative events, potential TSP violations and information flow decisions. The TOE records the identity of the Administrator or User who caused the event for which the audit record is created. The TOE applies timestamps to auditable events as they occur.

Upon detecting a potential TSP violation, the TOE immediately displays an alarm message identifying the potential TSP violation and, at the option of the Security Administrator, generates an audible alarm and makes accessible the audit record contents associated with the auditable event(s) that generated the alarm. The TOE displays alarm messages and sounds the audible alarm until the alarm has been acknowledged.

The administrator can review, search and sort the audit records. The audit records are stored locally; using memory, a hard disk or a FLASH memory card depending on the model. The storage devices used by each model for audit record storage are identified in Table 2.

The Security Administrator specifies whether the TOE prevents the loss of audit records or provides log rolling capabilities. If log rolling is not enabled, reaching 95% of the audit storage capacity results in the TOE entering an error mode which shuts down the network interfaces and therefore prevents the occurrence of auditable events (except those taken by an authorized administrator to clear the error mode). When the TOE is in the error mode, only administrative access is allowed and this access is restricted to the Security Administrator and Audit Administrator. The 95% audit log threshold limit allows the TOE to record the actions taken by Security Administrator or Audit Administrator to clear the error mode. When log rolling is enabled the oldest audit records are overwritten.

If the TOE is operating as part of an Active-Active HA cluster, the HA master logs all administrative events for the cluster. The HA master also logs all potential TSP violations and information flow decisions that it processes. HA slaves log all potential TSP violations and information flow decisions that they process. The administrator can access slave audit records through the master HA unit.

If the audit log of any node in a cluster becomes full, that node takes the action specified for the master node. If this action is to shut down the TOE interfaces the following will result:

- If the audit log of a slave node becomes full (active-active cluster), the slave node drops out of the cluster;
- If the audit log of a master node becomes full (active-active cluster), the master node has failed and one of the slave nodes will become the new master node; and
- If the audit log of the master node (active-passive cluster) becomes full, the master node has failed and the backup node will take over as the master node.

6.1.8 Self-Protection

F.PROTECT

The TOE ensures that no information flows from one network interface to another without passing through the TOE and being subject to the firewall rules.

The TOE maintains an isolated security domain for its own execution. FortiOS is the only application that is on the TOE and no other applications can be loaded onto the TOE. Administrators and users do not have access to the operating system or the file system (there are no root/system level users). The TOE stores all security and configuration data in segregated configuration files. The TOE only provides identification, authentication and information flow services to non-administrative users.

The TOE ensures that no residual data from previous packets passing through the TOE is reused in any way. Any residual information in any resource is over-written or otherwise destroyed so that it cannot be reused or otherwise accessed either inadvertently or deliberately.

The TOE runs a suite of self-tests during initial start-up, periodically during normal operation as specified by the Security Administrator, and at the request of an administrator to demonstrate the correct operation of the hardware portions of the TSF. The TOE also runs the suite of self-tests provided by the FIPS 140-2 cryptographic module during initial start-up, at the request of an administrator, and periodically at a Security Administrator-specified interval not less than once a day, to demonstrate the correct operation of the cryptographic components of the TSF.

Failure of the self-tests cause the TOE to enter a mode where the ability to return the TOE to a secure state is provided.

Time is provided by the TSF and can only be changed by the Security Administrator. Changes to the time are audited.

Before establishing a user session that requires authentication or before establishing an administrative session, the TOE displays a Security Administrator-specified advisory notice and consent warning message regarding unauthorized use of the TOE.

The TOE protects itself by rejecting replay of communications, avoiding overload of its interfaces, managing sessions, and restricting information released on banners.

The TOE terminates Authenticated User, administrative sessions, and VPN sessions after a Security Administrator-configurable time interval of inactivity.

The HA feature provides failover protection capability which includes configuration synchronization. FortiGates which form part of a HA cluster exchange configuration information using a proprietary protocol (FGCP). Before any information is exchanged members of a HA cluster authenticate using information built into the FortiGate at the time of manufacture. Configuration information is exchanged every time the configuration of the master node in a HA cluster is updated. In this way, the slave or passive nodes in a cluster are prepared to assume the role of master node should the master node fail. Section 6.1.7 (F.AUDIT) describes how audit information is protected by the TOE's HA capabilities.

F.IPS

The TOE provides an Intrusion Protection System that examines network traffic arriving on its interfaces for evidence of intrusion attempts. If such evidence is found, the TOE records the event in a sensor log. The sensor log is made available only to authorised administrators, and is provided in a manner suitable for the administrators to interpret the information.

The TOE protects the stored sensor data from modification and from unauthorised deletion. The TOE allows the Security Administrator to specify the action to be taken if the storage allocated for sensor data is full, either stop generating sensor data, or overwrite the oldest sensor data. An alarm is sent if the storage capacity has been reached.

The Sensor data is made available to remote trusted IT products within one minute of receipt of request for the data, provided the data is available for transmission. The TOE uses encryption to ensure that data transmitted from the TSF to a remote trusted IT product is protected from unauthorised disclosure during transmission. The TOE detects modification of TSF data transmitted between the TSF and a remote trusted IT product. The TOE will retransmit the data if the remote trusted IT product detects modifications and requests a re-transmission.

6.2 ASSURANCE MEASURES

A description of each of the TOE assurance measures follows.

M.ID The TOE incorporates a unique version identifier that can be displayed to the user.

M.CMSYS The TOE was developed and is maintained using a documented CM system, with automated generation support, to ensure that only authorised changes are made to the TOE configuration items and implemented in the evaluated version of the TOE and to support the generation of the TOE. The organization, operation and usage of the CM system are described in a CM plan, which describes the method used to uniquely identify the configuration items, describes the automated tools and their usage in the system, and identifies CM records that are to be retained as evidence that the CM system is operating in accordance with the plan and that all configuration items have been and are being effectively maintained under the CM system. A list that uniquely identifies and describes all configuration items that comprise the TOE, all TOE documentation, all configuration items required to create the TOE (i.e., implementation representation), security flaws and the evaluation evidence required by the assurance components of the ST, is maintained. The procedures used to accept modified or newly created configuration items as part of the TOE are documented in an acceptance plan.

M.GETTOE The developer uses a documented and controlled process and procedures for shipping a packaged TOE, identified by serial number, to a customer. The delivery documentation describes all procedures and technical measures that are necessary to maintain security and detect modifications or any discrepancy between the developer's master copy and the version received at the user site. The documentation describes how the procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

M.SETUP Documented procedures describe all the steps necessary for the secure installation, generation, and start-up of the TOE. Application of these procedures to the TOE results in a secure configuration.

M.SPEC The development documentation consists of a functional specification, a high level TOE design, and a low level TOE design.

The informal, internally consistent, functional specification describes the TSF and the purpose and method of use of all external TSF external interfaces, providing complete details of all effects, exceptions and error messages. The functional specification completely represents the TSF and includes rationale that the TSF is completely represented.

The informal, internally consistent high-level design describes the structure of the TSF in terms of TSP-enforcing and other subsystems, and, for each subsystem, describes the security functionality that it provides. The high-level design identifies all underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software. The high-level design identifies all interfaces to the subsystems of the TSF and identifies which of these interfaces are externally visible. The high-level design describes the purpose and method of use all interfaces to the subsystems of the TSF, and provides details of effects, exceptions and error messages, as appropriate.

The informal, internally consistent, low-level design describes the TSF in terms of TSP-enforcing and other modules, describes the purpose of each module, defines the interrelationships between the modules in terms of security functionality provided and dependencies on other modules, and describes how each TSP-enforcing function is provided. The low-level design identifies all interfaces to the modules of the TSF, identifies which of these interfaces are externally visible, and describes the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.

M.IMPREP An internally consistent implementation representation unambiguously defines the TSF to a level of detail such that the TSF can be generated without further design decisions.

M.TRACE Correspondence mappings demonstrate that the security functionality detailed in the TOE functional specification is upwards traceable to this ST, downwards traceable to the high level design, low level design, implementation representation, and is traceable to the TSP model. For each adjacent pair of provided TSF representations, a correspondence analysis demonstrates that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

M.TOESPM The informal TOE security policy model describes the rules and characteristics of all policies of the TSP that can be modeled. The rationale included with the model demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled. Correspondence between the functional specification and the TSP model shows that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

M.DOCS Documentation is provided in the form of operational guidance for the administrator and for the user.

The administrator guidance describes the administrative functions and interfaces available to the administrator of the TOE, describes how to administer the TOE in a secure manner, and contains warnings about functions and privileges that should be controlled in a secure processing environment. The administrator guidance describes all assumptions regarding user behaviour that are relevant to secure operation of the TOE, describes all security parameters under the control of the administrator, indicating secure values as appropriate, and describes each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF. The administrator guidance is consistent with all other documentation supplied for evaluation, and describes all security requirements for the IT environment that are relevant to the administrator. Procedurally, the administrator is required to choose a password with the following characteristics:

- One (or more) of the characters should be capitalized
- One (or more) of the characters should be numeric
- One (or more) of the characters should be non alpha-numeric (e.g. punctuation mark)

The user guidance describes the functions and interfaces available to the non-administrative users of the TOE, describes the use of user-accessible security functions provided by the TOE, and contains warnings about user-accessible functions and privileges that should be controlled in a secure processing environment. The user guidance clearly presents all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment. The user guidance is consistent with all other documentation supplied for evaluation, and describes all security requirements for the IT environment that are relevant to the user. Flaw remediation guidance is provided to describe how TOE users report to the developer any suspected security flaws in the TOE. The flaw remediation guidance also describes a means by which TOE users may register with the developer, to be eligible to receive security flaw reports and corrections. The flaw remediation guidance identifies the specific points of contact for all reports and enquiries about security issues involving the TOE.

M.DEVSEC The development security documentation describes all the physical, procedural, personnel and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment and provides evidence that these security measures are followed during the development and maintenance of the TOE.

M.FLAWREM Flaw remediation procedures, addressed to TOE developers, establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to these flaws. The flaw remediation procedures documentation describes the procedures used to track all reported security flaws in each release of the TOE. The flaw remediation procedure requires that a description of the nature and effect of each flaw be provided, as well as the status of finding a correction to that flaw. The flaw remediation procedure requires that corrective actions be identified for each of the security flaws and the flaw remediation procedures documentation describes the methods used to provide flaw information, corrections, and guidance on corrective actions to TOE users. The flaw remediation procedures documentation describes a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE. The procedures for processing reported security flaws ensures that any reported flaws are corrected and the correction issued to TOE users. The procedures for processing reported security flaws provide safeguards that any corrections to these security flaws do not introduce any new flaws. The flaw remediation procedures include a procedure requiring timely responses for the automatic distribution of security flaw reports and the associated corrections to registered users who might be affected by the security flaw.

M.LIFECYCLE A life-cycle model has been established for use in the development and maintenance of the TOE. Life-cycle definition documentation has been produced that describes this life-cycle model. The life-cycle model provides for the necessary control over the development and maintenance of the TOE.

M.DEVTOOLS The development tools being used for the TOE have been identified and the selected implementation-dependent options of the development tools have been documented. All development tools used for implementation are well-defined. The documentation of the development tools unambiguously defines the meaning of all statements and of all implementation-dependent options used in the implementation.

M.TESTCOV An analysis of the test coverage demonstrates the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification. This analysis demonstrates that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

M.TESTDPT An analysis of the depth of testing demonstrates that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

M.DEVTEST A suitably configured TOE is tested by the developer in a controlled environment to confirm that the TSF operates as specified, and that the TOE is protected from a representative set of well-known attacks. The developer-provided test documentation consists of test plans, test procedure descriptions, expected test results and actual test results. The test plans identify the security functions to be tested and describe the goal of the tests to be performed. The test procedure descriptions identify the tests to be performed and describe the scenarios for testing each security function. These scenarios include any ordering dependencies on the results of other tests. The expected test results show the anticipated outputs from a successful execution of the tests. The test results from the developer execution of the tests demonstrate that each tested security function behaved as specified.

M.INDTEST Independent tests, which are conducted on a suitable TOE, with the aid of a set of resources equivalent to those that were used in the developer's functional testing of the TSF, confirm that the TOE operates as specified.

M.VALIDANAL The guidance documentation identifies all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation, lists all assumptions about the intended environment, and lists all requirements for external security measures (including external procedural, physical and personnel controls). This guidance documentation is complete, clear, consistent and reasonable. The fact that the guidance documentation provides sufficient information to permit the TOE to be configured and used securely using only the supplied guidance documentation, and allows all insecure states to be detected is confirmed by independent evaluation and performance of the procedures using only the supplied guidance. The developer-provided analysis of the guidance documentation demonstrates that the guidance documentation is complete, and that guidance is provided for secure operation in all modes of operation of the TOE.

M.SOFASS A strength of TOE security function analysis is performed and documented for F.I&A, which is the only mechanism identified in the ST as having a strength of TOE security function claim. This analysis shows that F.I&A meets or exceeds the specific strength of function metric defined in the ST.

M.VULANAL The TOE design is examined to ensure that the security functions adequately address perceived threats in the security environment. Threats include deliberate attempts to disable, bypass, and brute-force attack the TSF. A documented vulnerability analysis of the TOE deliverables is conducted in order to search for ways in which a user can violate the TSP, and the disposition of identified vulnerabilities is documented, showing, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE. The vulnerability analysis documentation justifies that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks performed by an attacker possessing a low attack potential.

7 PROTECTION PROFILE CLAIMS

This section provides the IDSS PP conformance claim statements.

7.1 IDSS PP REFERENCE

The TOE conforms to the following IDSS PP:

- Intrusion Detection System Sensor Protection Profile (IDSS PP), Prepared for National Security Agency by Science Applications International Corporation, Version 1.2, April 27, 2005.

7.2 IDSS PP TAILORING

The following tailoring was applied to the IDSS PP to produce this ST:

- In response to consumer demand, the assurance package was upgraded from EAL2 to EAL4, augmented by ALC_FLR.3;
- The A.NO_TOE_BYPASS and A.PHYSICAL assumptions were drawn from the FW PP MR, the TFFW PP MR and the VPN PP MR. The A.NO_GENERAL_PURPOSE assumption found in the FW PP MR, the TFFW PP MR and the VPN PP MR was omitted as the TOE consists of proprietary hardware and software and thus it is not possible load general purpose computing software onto the TOE. All other assumptions are drawn from the IDSS PP. Readers should note that the A.PHYSICAL and A.PROTCT assumptions are very similar. Both were retained in the ST to reflect their differing origins.
- The threat statements for the T.INADVE, T.MISACT and T.MISUSE threats (from the IDSS PP) were expanded to clearly identify the threat agent.
- There are minor differences in the wording of the threat statements in the three MR PPs. The wording from the VPN PP MR was used in this ST as it is the most recent. Also the VPN PP MR includes one threat T.UNAUTHORIZED_PEER which is not included in the FW PP MR or the TFFW PP MR.
- There are minor differences in the wording of organizational security policies in the three MR PPs. The wording from the VPN PP MR was used in this ST as it is the most recent. Also, the VPN PP MR includes one policy P.INTEGRITY which is not included in the FW PP MR or the TFFW PP MR. In the MR PPs, the P.VULNERABILITY_ANALYSIS_TEST security policy states that the TOE must undergo independent testing to demonstrate that the TOE is resistant to an attacker possessing a medium attack potential. As the threat environment for this evaluation is based upon attackers with low attack potential, this policy

statement was modified to be consistent with the claimed environment and strength of function level.

- There are minor differences in the wording of the security objectives in the three MR PPs. The wording from the VPN PP MR was used in this ST as it is the most recent. Also, the VPN PP MR includes two security objectives; O.INTEGRITY and O.PEER_AUTHENTICATION, which are not found in the FW PP MR or the TFFW PP MR. In the MR PPs, the O.VULNERABILITY_ANALYSIS_TEST security objective states that the TOE will undergo appropriate independent testing to demonstrate that the TOE is resistant to attackers with medium attack potential. As the security environment for this evaluation is based on attackers with low attack potential, this security objective was modified accordingly.
- The O.DOCUMENT_KEY_LEAKAGE objective from the FW PP MR, the TFFW PP MR and the VPN PP MR was omitted from the ST. This objective is related to AVA_VLA.3 security assurance requirement (included in the MR PPs) which is not included in the ST.
- The names of the security objectives for the environment were changed from the "O.XXX" notation in the IDSS PP to "OE.XXX" notation to provide a clearer distinction from the TOE security objectives, which are labeled "O.XXX".
- The OE.NO_GENERAL_PURPOSE objective found in all three MR PPs was omitted from the ST. The TOE consists of proprietary hardware and software and it is not possible to load general purpose software onto the TOE.
- FAU_ARP.1.1

  i) The requirement was rewritten to clearly distinguish between the alarm method (on screen message and optionally audible) and the recipients of the alarm.

  ii) The term 'remote administrator' was changed to 'Local Console, Network Web-Based GUI, and Network CLI' in order to be specific.

  iii) The [*assignment: other methods*] was omitted since the TOE does not implement other methods of indicating an alarm.
- FAU_ARP_ACK_EXP.1.2

  i) The term 'remote administrator' was changed to 'Network Web-Based GUI, and Network CLI' in order to be specific.

  ii) The words 'if they still exist' was added for the remote administrator sessions (Network Web-Based GUI and Network CLI) since a session could be terminated by the TSF (FTA_SSL.1) or an administrator could log out before the alarm was acknowledged.

- FAU_GEN.1 – A NIAP refinement of this requirement (FAU_GEN.1-NIAP-0410) is included in all three MR PPs. This ST has used the requirement statement from the IDSS PP as a starting point and made refinements (as noted below) so that the requirement is compliant with all four PPs.
- FAU_GEN.1.1 – Subparagraph c) from the IDSS PP was reworded to include the requirements from all four PPs.
- FAU_GEN.1.2 – The qualification '(if applicable)' was added to the identification of a subject identity since the subject identity may not always be known. Subparagraph b) was reworded to correctly identify Table 7 as the source for identifying additional audit record contents.
- The IDSS PP-specific requirements for audit data generation (FAU_GEN.1) were merged with other audit requirements into one comprehensive table;
- FAU_GEN.1, Table 7 - Auditable Events

  i) FIA_AFL.1 - The wording from the MR PPs has been used since it is more explicit than that in the IDSS PP. The word 'identify' was changed to 'claimed identity'.
  ii) FMT_MOF.1(1) - the administrator as explicitly defined as the Security Administrator
  iii) FMT_MOF.1(2) - the administrator was explicitly defined as the Security Administrator
  iv) FMT_MSA.1 - the administrator was explicitly defined as the Security Administrator
  v) FMT_MTD.1(2) - the administrator was explicitly defined as the Cryptographic Administrator
  vi) FMT_MTD.1(3) - the administrator was explicitly defined as the Security Administrator
  vii) FMT_MTD.2(1) - the administrator was explicitly defined as the Security Administrator
  viii) FPT_FLS.1 - This requirement was added from the CC.
  ix) FPT_RPL.1 - A typographical error was corrected, 'reply' was changed to 'replay'.
  x) FRU_FLT.1 - This requirement was added from the CC.
  xi) FTA_SSL.1 - To make the distinction between local and remote sessions, 'interactive session' was changed to 'Local Console interactive session'. For clarity 'user' was changed to 'Administrator' since this function only applies to an Administrator.
  xii) FTA_SSL.2 - To make the distinction between local and remote sessions, 'interactive session' was changed to 'Local Console interactive session'. For clarity 'user' was changed to 'Administrator' since this function only applies to an Administrator.
  xiii) FTA_SSL.3 - The term 'remote session' was changed to 'authenticated Proxy User, VPN User, Network Web-Based GUI, Network CLI' session

'. For clarity 'user' was changed to 'User or Administrator' to make the requirement clearer.

xiv) FTA_TSE.1 - For clarity 'user' was changed to 'User or Administrator' to make the requirement clearer.

- FAU_GEN.2.1 - For clarity 'user' was changed to 'Administrator or User'.
- FAU_SAA.1.1 - 'the audited events' was changed to 'events'.
- FAU_SAA.1.2 - The following changes were made to the requirements provided in the FW PP MR and TFFW PP MR:

  i) For clarity the phrase 'administrator specified time period' in subparagraphs a)(2), a)(3), a)(4) and a)(5) was changed to 'Security Administrator specified time period'.

  ii) The completed assignment operation in subparagraph b(1) includes Fortinet Protection Profiles since they include additional TOE functionality not provided by the Information Flow policy violations covered in subparagraphs a)(2), a)(3), a)(4) and a)(5).

- FAU_SAR.1.2 – The word 'user' was changed to 'Administrators' since audit review is restricted to administrators.
- FAU_SAR.2.1 – The requirement was simplified since only administrators have access to the audit data.
- FAU_SAR.3.1 – The IDSS PP only requires the ability to sort audit data based on a small number of criteria. The TOE is capable of both searching and sorting the audit data based on a wider selection of criteria which allows it to conform to both the IDSS PP requirement and the requirements of the MR PPs.
- FAU_SEL.1.1 – The requirement from the IDSS PP was refined to specify that only the Security Administrator is able to include or exclude auditable events. Additional auditable event attributes were specified. These changes allow the ST to conform to the IDSS PP requirement and the requirements of the MR PPs.
- FAU_STG.2.1 – The requirement from the IDSS PP was refined to clearly indicate that the Audit Administrator is the only role authorized to delete records from the audit trail. This refined allows the ST to comply with the FAU_STG.1.1-NIAP-0423 Protected Audit Trail Storage requirement from the MR PPs.
- FAU_STG.2.2 – The IDSS PP requires that a compliant TOE ‘detect’ audit trail modifications. The requirement has been refined to be more restrictive by replacing the word ‘detect’ with ‘prevent’. This refinement allows the ST to conform with the IDSS PP and the MR PPs.
- FAU_STG.3 – The following changes were made with respect to the requirement from the FW PP MR, the TFFW PP MR and the VPN PP MR:

i) This requirement was rewritten into a bulleted list format rather than a paragraph format and in the process, the extraneous open bracket included in the PPs was omitted.

ii) The phrase 'and at the remote administrative console' was changed to 'Network Web-Based GUI and Network CLI' in order to be specific.

iii) The phrase ‘generate an audible alarm,’ was changed to ‘immediately alert the administrators by generating an audible alarm at the Local Console, Network Web-Based GUI, and Network CLI when an administrative session exists for each of the defined administrative roles; and’ in order to more accurately describe the audible alarm function.

- FAU_STG.4.1 – The requirement from the IDSS PP was rewritten to into a bulleted list format in order to clearly distinguish the mandatory and optional components of the requirement.
- FAU_STG.4.2 – This component was added to the requirement as a refinement. While it is beyond the requirements of the IDSS PP, it is requirement for conformance with the MR PPs.
- FCS_BCM_EXP.1.2 – The VPN PP MR words this requirement differently than the FW PP MR and the TFFW PP MR. The later wording has been used, with the additional words ‘and meet FIPS PUB 140-2, Level 4 Self Tests’ added as a refinement so that the ST conforms with all three MR PPs.
- FCS_CKM.1 – The FW PP MR and TFFW PP MR require that the CC author specify the standard which is used by the TOE to generate random numbers for symmetric key generation. The standard used by the TOE (which defines a FIPS-Approved random number generation algorithm) has been specified in this requirement.
- FCS_CKM.4

i) The MR PPs specify overwriting of cryptographic keys three times. The TOE overwrites cryptographic keys stored in flash memory once only. A rationale for this change has been provided in the application note associated with the FCS_CKM.4.1 requirement.

ii) The wording of this requirement in the VPN PP MR differs from the wording used in the FW PP MR and the TFFW PP MR. The later wording was used as the basis for the requirement as it specifies overwriting requirements for intermediate storage areas which are omitted from the VPN PP MR.

iii) The word 'and' was moved from the end of subparagraph b) to the end of subparagraph c).

- FDP_IFC.1.1(1) – The FW PP MR includes a selection operation in the last bullet point for this requirement. The purpose of the selection operation is to allow the ST author to include any other application proxies (in addition to SMTP) which do not require authentication. Since the TOE does not provide any other application proxies which do not require authentication, the selection operation has been omitted from the ST.

- FDP_IFC.1.1(2) – For this requirement, the FW PP MR includes (in subparagraph d) a selection operation which allows the ST author to specify additional application proxies which require authentication. Since the TOE does not require authentication for any additional application proxies, the selection was omitted.
- FDP_IFC.1.1(4) – For this requirement, the VPN PP MR includes (in subparagraph d) an assignment operation which allows the ST author to specify additional operations to be performed on network packets which are subject to the VPN policy. Since the ST does not specify any additional operations, the assignment operation was omitted.
- FDP_IFF.1.1(1)

  i) The FW PP MR and TFFW PP MR include a selection operation as the second bullet point for subparagraph a). The purpose of the selection operation is to allow the ST author to specify additional source subject security attributes. Since the TOE does not define additional source subject security attributes, this selection operation has been omitted.

  ii) For clarity, the information security attributes for schedules (which are included using the selection operation in the PP) were moved ahead of the SMTP attributes in subparagraph c).

  iii) To correct a PP format issue, the stateful packet attributes subparagraph is labeled as subparagraph d) rather than appearing as a bullet under subparagraph c).

  iv) The FW PP MR and TFFW PP MR include selection operations for both connection-oriented and connectionless protocols which allow the specification of additional security attributes. Since the TOE does not define any additional security attributes for these protocols, the selection operations were omitted.

- FDP_IFF.1.1(2)

  i) The FW PP MR includes a selection operation as the second bullet point for subparagraph a). The purpose of the selection operation is to allow the ST author to specify additional source subject security attributes. Since the TOE does not define additional source subject security attributes, this selection operation has been omitted.

  ii) The FW PP MR include a selections operation as the second bullet point for subparagraph b). The purpose of the selection operation is to allow the ST author to specify additional destination subject security attributes. Since the

TOE does not define additional destination subject security attributes, this selection operation has been omitted.

iii) The FW PP MR includes two selection operations as the last two bullet points for subparagraph c). The first selection allows the ST author to include any sub-commands associated with any additional application proxies defined in FDP_IFC.1(2). However, since there are no additional application proxies defined in FDP_IFC.1(2), the selection operation was omitted. The second PP selection allows the ST author to include any additional information security attributes used by the TOE. Since this TOE does not use any additional information security attributed, this selection was also omitted.

iv) To correct a FW PP MR format issue, the stateful packet attributes subparagraph is labeled as subparagraph d) rather than appearing as a bullet under subparagraph c).

v) The FW PP MR includes selection operations for both connection-oriented and connectionless protocols which allow the specification of additional security attributes. Since the TOE does not define any additional security attributes for these protocols, the selection operations were omitted.

- FDP_IFF.1.2(2) – The word 'administrator' was replaced by 'Security Administrator' to make it clear that the rules in the information policy flow ruleset are defined by the Security Administrator.
- FDP_IFF.1.4(2) – The layout of the requirement has been modified from that used in the FW PP MR since the PP uses a bulleted list format which includes only one bullet point. The modified layout is consistent with the layout used by the PP for FDP_IFF.1.4(1) and FDP_IFF.1.4(3).
- FDP_IFF.1.1(3)

i) The MR PPs include selection operations in the last bullet point for subparagraphs a) and b) which allows the ST author to specify additional subject security attributes for source and destination subjects. However, since the TOE does use additional subject security attributes in order to enforce the UNAUTHENTICATED TOE SERVICES SFP, these selection operations have been omitted.

ii) The MR PPs include a selection operation in the last bullet point for subparagraph c) which allows the ST author to specify additional information security attributes for services identified in FIA_UAU.1(1). However, since this ST does not identify any additional services in FIA_UAU.1(1) the selection operation has been omitted.

- FDP_IFF.1.3(3) – The MR PPs include a selection operation in the first bullet point for this requirement which allows the ST author to list other unauthenticated network services provided by the TOE. However, since the

TOE does not provide any other unauthenticated network services (except ICMP) the selection operation has been omitted.

- FDP_IFF.1.1(4) – The VPN PP MR includes selection operations in the last bullet point for subparagraphs a) and b) which allows the ST author to specify additional subject security attributes for source and destination subjects. However, since the TOE does use additional subject security attributes in order to enforce the VPN SFP, these selection operations have been omitted.
- FIA_AFL.1 – The IDSS PP uses the FIA_AFL.1 requirement to describe authentication failure handling for external IT products attempting to authenticate to the TOE. The MR PPs impose additional requirements for authentication failure handling. In order to comply with all of the PPs, the wording from the MR PPs was used as a basis and then refined to specifically describe the authentication failure handling capabilities of the TOE. It should be noted that although the TOE communicates with Fortinet's FortiGuard Distribution Server and FortiAnalyzer (as trusted IT entities), the TOE authenticates to these external entities. For this reason these external trusted IT entities are not listed in the FIA_AFL.1 requirement.
- FIA_AFL.1.1 –The following changes were made with respect to wording of this requirement from the MR PPs:

  i) The phrase 'administrators attempting to authenticate remotely' was replaced by 'administrators attempting to authenticate to the Network Web-Based GUI and Network CLI' in order to be more specific.

  ii) The phrase 'authenticated proxy users' (which is used in the FW PP MR) was replaced by 'attempted proxy user authentication' which is more precise.

  iii) The phrase 'authorized IT entities' was replaced by 'authentication attempts by VPN peers' as these are the only authorized IT entity which will authenticate to the TOE.
- FIA_AFL.1.2 – The following changes were made with respect to wording from the MR PPs:

  i) The requirement was reworded to make it clear that it applies to a proxy user attempting to authenticate rather than applying to an authenticated proxy user.

  ii) A bullet point list format was used to make the requirement clearer.

  iii) The PPs do not clearly indicate that the authentication limit applies to each item and/or assumed user individually. For instance, failures by a given proxy user should not lock out all remote administrators. To make this distinction clear in the ST, the words 'for the user assumed to have exceeded the authentication attempt limit' was added for proxy user authentication and remote administrator authentication. Similarly the words ‘for the VPN peer assumed to have exceeded the authentication attempt limit’ were added for VPN peer authentication.

- FIA_ATD.1 - The requirement was iterated and the user was made explicit (administrators, authorized proxy user, and VPN remote devices). The IDSS PP lists the user’s security attributes as user identity, authentication data, and authorizations. The ST author has chosen to be more specific and use username / password rather than user identity / authentication data and has used role rather than authorizations.
- FIA_UAU.1(2) - The FW PP MR includes a selection operation which allows the ST author to specify additional unauthenticated proxy services. Since the TOE does not provide any additional unauthenticated proxy services (beyond SMTP) the selection was omitted.
- FIA_UAU.2.1

  i) The FW PP MR includes a selection operation which allows an ST author to specify additional proxy services (beyond Telnet, FTP and HTTP) which require authentication. Since the TOE does not require authentication for any additional proxy services, this selection was omitted.

  ii) The authorized IT entity (Fortinet's FortiGuard Distribution Server) was explicitly stated.

  iii) The phrase ‘VPN Peers’ was included in the list of users which require authentication before any action.
- FIA_UAU.5 – The explicit requirement used by the three MR PPs was replaced by the standard CC requirement. The requirement was refined by replacing the word ‘support’ with the word ‘perform’ to conform to the wording used by the MR PPs.
- FIA_UID.1 was replaced by FIA_UID.2. Since FIA_UID.2 is hierarchical to FIA_UID.1, the IDSS PP requirement is met;
- FMT_MOF.1 – The IDSS PP requirement was iterated due to the inclusion of the CC Part 2, FW PP MR TFFW PP MR and VPN PP MR requirements. The IDSS PP requirement was moved to FMT_MOF.1(13).
- FMT_MOF.1(1) to FMT_MOF.1(7) - Additional iterations of the FMT_MOF.1 requirement have been added from the FW PP MR, TFFW PP MR and VPN PP MR.
- FMT_MOF.1(8) - An additional iteration of the FMT_MOF.1 requirement was added to address the management of the cryptographic self-tests.
- FMT_MOF.1(9) - An additional iteration of the FMT_MOF.1 requirement was added to address the management of actions to be taken in the event of audit storage exhaustion.

- FMT_MOF.1(10) - An additional iteration of the FMT_MOF.1 requirement was added to address the management of the session termination function.
- FMT_MOF.1(11) - An additional iteration of the FMT_MOF.1 requirement was added to address the management of the alarm acknowledgement function.
- FMT_MOF.1(12) - An additional iteration of the FMT_MOF.1 requirement was added to address the management of the on-demand execution of the cryptographic and non-cryptographic self-tests.
- FMT_MOF.1(13) - The review requirement was specified. The words ‘authorised Sensor administrators’ were replaced by the phrase ‘an Administrator’ in order to conform with the terminology used throughout the ST. The TOE permits all the Administrators (Security, Audit and Crypto) to act as administrators of the IDS Sensor.
- FMT_MSA.3(1) – The phrase ‘security attributes’ was replaced by the phrase ‘the information flow policy ruleset’ in order to conform with the FW PP MR, TFFW PP MR and the VPN PP MR. The information flow policy ruleset comprises the security attributes which define the security functional policies listed by this requirement. In addition the word ‘are’ was changed to ‘is’ to remain grammatically correct.
- FMT_MSA.3(2) – The phrase ‘security attributes that are used to enforce the SFP’ was replaced by the phrase ‘the set of TOE services available to unauthenticated users’ in order to comply with the FW PP MR, TFFW PP MR and VPN PP MR. The set of TOE services available to unauthenticated users describes the security attributes applicable to the enforcement of the UNAUTHENTICATED TOE SERVICES SFP.
- FMT_MTD.1 – The FW PP MR, TFFW PP MR and VPN PP MR each list a number of iterations of the FMT_MTD.1 requirement. The first iteration in all these PPs is not an actual requirement statement, but rather a placeholder which is intended for the ST author to include additional TSF data management requirements not covered by the PPs. In this ST, this iteration of the requirement (along with the fifth, sixth and seventh iterations are used to include TSF data management requirements which are specific to the TOE). The second, third and fourth iterations are requirements imposed by the FW PP MR, the TFFW PP MR and the VPN PP MR. The eighth iteration is a requirement levied only by the VPN PP MR, while the nineth and final iteration is a requirement levied only by the IDSS PP MR.
- FMT_MTD.1(1) – This iteration of the requirement was added to describe the management of audit data.
- FMT_MTD.1(5) – This iteration of the requirement was added to describe the management of user account data.

- FMT_MTD.1(6) – This iteration of the requirement was added to describe the management of TOE banner data.
- FMT_MTD.1(7) – This iteration of the requirement was added to describe the management of AV and IPS signature data by the TOE.
- FMT_MTD.1(9) - This requirement was refined. The TOE does not allow sensor data to be added. The IDSS PP wording also includes audit data and TOE data which (in this ST) are addressed by the other iterations of FMT_MTD.1.
- FMT_REV.1.1 - The FW PP MR includes a selection operation which allows the ST author to specify additional resources for which the revocation of security attributes may be restricted. Since the TOE does not provide any additional resources for which security attributes may be revoked, the selection was omitted.
- FMT_REV.1.2 - The FW PP MR includes a selection operation which allows the ST author to specify the immediate enforcement of security attribute revocation for additional resources defined in FMT_REV.1.1. Since the ST does not list any additional resources in FMT_REV.1.1, the selection has been omitted.
- FMT_SMR.1 was replaced by FMT_SMR.2. Since FMT_SMR.2 is hierarchical to FMT_SMR.1, the IDSS PP requirement is met;
- FMT_SMR.2.1

  i) The role of Sensor Administrator, which is required by FMT_SMR.1 in the IDSS PP, was merged with the Security Administrator, which is identified in FMT_SMR.2.

  ii) The VPN User role has been added.
- FMT_SMR.2.3

  i) In FMT_SMR.2.1, the FW PP MR and TWFW PP MR allow the ST author to define additional roles. Then in the first three bullet points of FMT_SMR.2.3, the PPs make reference to ‘all roles’ when it is clear that the phrase ‘all administrator roles’ was intended. Therefore the phrase ‘all roles’ has been replaced by the phrase ‘all administrator roles’.

  ii) The phrases ‘locally’ and ‘remotely’ where replaced with ‘via the Local Console’ and ‘via the Network Web-Based GUI and Network CLI’ respectively, in order to be specific about the location of administrative sessions.
- FPT_AMT.1 – The phrase ‘periodically during normal operation’ was replaced with ‘periodically during normal operation as specified by the Security Administrator’ in order to clearly specify that only the Security Administrator can specifiy the periodicity with which the self tests are executed.

- FPT_ITI.1.2 – The requirement was made more explicit by specifying that modifications are to be detected by the receiving device.
- FPT_SEP.1 was replaced by FPT_SEP.2. Since FPT_SEP.2 is hierarchical to FPT_SEP.1, the IDSS PP requirement is met.
- FPT_SEP.2.3 – The FW PP MR, TFFW PP MR and VPN PP MR have refined this requirement so that it describes protection afforded by the TSF to its cryptographic functionality, rather than to security functional policies as included in the standard CC Part 2 requirement. This refinement has been retained in this ST.
- FPT_TST.1(1) – This requirement has been refined to satisfy the explicit requirement (FPT_TST_EXP.4) used by the FW PP MR, TFFW PP MR and VPN PP MR.
- FPT_TST.1(2) – This requirement has been refined to satisfy the explicit requirement (FPT_TST_EXP.5) used by the FW PP MR, TFFW PP MR and VPN PP MR.
- FRU_RSA.1(1) – In the selection operation which identifies the user or subject to which the quota applies, the selection 'subject' was replaced with 'a sourc subject identifier' to clearly identify the subject using the terminology of the ST.
- FRU_RSA.1(2) – The ST has retained the refinements introduced by the FW PP MR, TFFW PP MR and VPN PP MR.
- FTA_SSL.1 – The phrase 'an interactive session' was replaced with 'a Local Console interactive session' in order to use terminology specific to the TOE.
- FTA_SSL.2 – This requirement was refined in order to describe (for Local Console sessions) the more restrictive requirement of session termination rather than session locking.
- FTA_SSL.3.1 – The words 'an interactive session' were replaced with 'an authenticated Proxy User, VPN User, Network Web-Based GUI or Network CLI session' in order to explicitly define the remote sessions which will be terminated.
- FTA_TAB.1.1 – This requirement was refined in order to comply with the FW PP MR, TFFW PP MR and VPN PP MR.
- FTA_TSE.1.1 – This requirement was refined in order to comply with the FW PP MR, TFFW PP MR and VPN PP MR.
- FTP_ITC.1.1(1) - With respect to the FW PP MR/TFFW PP MR, the phrase 'authorized IT entities' was replaced with 'Fortinet's FortiGuard Distribution Server' as this is the only authorized IT entity.

- FTP_ITC.1.2(1) - With respect to the FW PP MR/TFFW PP MR, the phrase ‘authorized IT entities’ was replaced with ‘Fortinet’s FortiGuard Distribution Server’ as this is the only authorized IT entity.
- FTP_ITC.1.1(2) - With respect to the FW PP MR/TFFW PP MR, the phrase ‘authorized IT entities’ was replaced with ‘Fortinet’s FortiGuard Distribution Server’ as this is the only authorized IT entity.
- FTP_ITC.1.2(2) - With respect to the FW PP MR/TFFW PP MR, the phrase ‘authorized IT entities’ was replaced with ‘Fortinet’s FortiGuard Distribution Server’ as this is the only authorized IT entity.
- FTP_TRP.1.1(1) - With respect to the FW PP MR/TFFW PP MR, the term ‘remote administrators’ was replaced with the phrase ‘administrators using the Network Web-Based GUI and Network CLI’ in order to be more specific. VPN Users as added.
- FTP_TRP.1.2(1) - With respect to the FW PP MR/TFFW PP MR, the term ‘remote users’ was replaced with the phrase ‘proxy users and administrators using the Network Web-Based GUI and Network CLI’ in order to make it clear that the users include proxy users and administrators. VPN Users as added.
- FTP_TRP.1.3(1) - With respect to the FW PP MR/TFFW PP MR:

  i) For clarity 'user authentication' was changed to 'proxy user and administrator authentication'.

  ii) The PPs include a selection operation which allows an ST author to specify additional services for which a trusted path is required. Since the TOE does not provide any additional services which require a trusted path, the selection was omitted.
- FTP_TRP.1.1(2) - With respect to the FW PP MR/TFFW PP MR, the term ‘remote administrators’ was replaced with the phrase ‘administrators using the Network Web-Based GUI and Network CLI’ in order to be more specific.
- FTP_TRP.1.2(2) - With respect to the FW PP MR/TFFW PP MR, the term ‘remote users’ was replaced with the phrase ‘proxy users and administrators using the Network Web-Based GUI and Network CLI’ in order to make it clear that the users include proxy users and administrators.
- FTP_TRP.1.3(2) - With respect to the FW PP MR/TFFW PP MR:

  i) For clarity 'user authentication' was changed to 'proxy user and administrator authentication'.

  ii) The PPs include a selection operation which allows an ST author to specify additional services for which a trusted path is required. Since the TOE does not provide any additional services which require a trusted path, the selection was omitted.

- FTP_TRP.1.3(2) - VPN Users was added.
- IDS_COL_EXP.1.1 – This requirement from the IDSS PP has been refined leaving out the assignment operation which allows the ST author to detail other specifically defined auditable events. Since the TOE only creates IDS audit records for network traffic (as specified in the previous selection operation) the assignment operation from the PP has been omitted.
- IDG_STG_EXP.2 – This requirement from the IDSS PP has been refined to reflect the capabilities of the TOE with respect to the protection of IDS Sensor data, when the storage capacity for that data has been exhausted.
- FTP_ITC.1.3(1)(ENV) - With respect to the FW PP MR/TFFW PP MR, the phrase 'all authentication functions' was replaced by the phrase 'FortiGuard Distribution Server authentication' as the FortiGuard Distribution Server is the only authorized IT entity for the TOE.
- FTP_ITC.1.3(2)(ENV) - With respect to the FW PP MR/TFFW PP MR, the phrase 'all authentication functions' was replaced by the phrase 'FortiGuard Distribution Server authentication' as the FortiGuard Distribution Server is the only authorized IT entity for the TOE.
- FTP_TRP.1.2(1)(ENV) - With respect to the FW PP MR/TFFW PP MR, the term 'remote users' was replaced with 'Network Web-Based GUI administrators, Network CLI administrators and Network Users' in order to more precisely identify the users of the trusted path.
- FTP_TRP.1.3(1)(ENV) - With respect to the FW PP MR/TFFW PP MR:

  i) The phrase 'user authentication, all remote administration actions' was replaced with 'Network User authentication, administrator authentication and all administrative use of the Network Web-Based GUI and Network CLI' in order to more clearly specify the services provided by the environment which require use of the trusted path.

  ii) The PPs include a selection operation which allows the ST author to list other services for which the trusted path is required. Since the TOE does not require the environment to provide any additional services via the trusted path, the selection was omitted.
- FTP_TRP.1.2(2)(ENV) - With respect to the FW PP MR/TFFW PP MR, the term 'remote users' was replaced with 'Network Web-Based GUI administrators, Network CLI administrators and Network Users' in order to more precisely identify the users of the trusted path.
- FTP_TRP.1.3(2)(ENV) - With respect to the FW PP MR/TFFW PP MR:

  i) The phrase 'user authentication, all remote administration actions' was replaced with 'Network User authentication, administrator authentication and all administrative use of the Network Web-Based GUI and Network

CLI' in order to more clearly specify the services provided by the environment which require use of the trusted path.

ii) The PPs include a selection operation which allows the ST author to list other services for which the trusted path is required. Since the TOE does not require the environment to provide any additional services via the trusted path, the selection was omitted.

- IDS_STG_EXP.2.1 was reworded to make the requirement clear;
- In order to provide additional guidance on the intended use and the operating environment, additional assumptions and threats were added. The assumptions are listed in Sections 3.1 and 3.2 and the additional ones have been identified;
- Objectives were added to counter the added threats. The objectives are listed in Sections 4.1 and 4.2 and the additional ones have been identified;
- Additional SFRs were added. The SFRs are identified in Table 6 - Security Functional Requirements.

8 RATIONALE

This section describes the rationale for the Security Objectives and Security Functional Requirements as defined in Section 4 and Section 5, respectively. Additionally, this section describes the rationale for satisfying all of the dependencies and the rationale for the strength of function (SOF) claim.

8.1 RATIONALE FOR SECURITY OBJECTIVES

8.1.1 Overview

Table 10, Table 11, and Table 12 present a bi-directional mapping of Assumptions, Threats, and Organizational Policies to Security Objectives for the TOE and for the Environment. Three tables are used in order to present the information in a readable format. In order to allow the reader to ensure that the mapping is complete, each table includes all assumptions, threats and policies. Consequently all rows in a given table do not map to an objective. The tables show that each of the assumptions, threats and organizational policies is addressed by at least one security objective, and that each security objective addresses at least one of the assumptions, threats, or organizational policies. This overview is followed by detailed descriptions and rationale for the mapping to TOE Security Objectives and to the Security Objectives for the Environment.

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH
A.ACCESS															
A.LOCATE															
A.MANAGE															
A.NO_TOE_BYPASS															
A.NOEVIL															
A.NOTRST															
A.PHYSICAL															
A.PROTCT															
T.ADDRESS_MASQUERADE															
T.ADMIN_ERROR		X													
T.ADMIN_ROGUE		X													

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH
T.AUDIT_COMPROMISE				X											
T.COMDIS	X												X		X
T.COMINT	X														X
T.CRYPTO_COMPROMISE															
T.FLAWED_DESIGN						X									
T.FLAWED_IMPLEMENTATION							X								
T.IMPCON	X											X			X
T.INADVE						X								X	
T.INFLUX															
T.INTRUSION															
T.LOSSOF	X														X
T.MALICIOUS_TSF_COMPROMISE											X				
T.MASQUERADE															
T.MISACT						X								X	
T.MISUSE						X								X	
T.NOHALT	X													X	X
T.POOR_TEST								X							
T.PRIVIL	X														X
T.REPLAY															
T.RESIDUAL_DATA															
T.RESOURCE_EXHAUSTION															
T.SPOOFING															
T.UNATTENDED_SESSION															
T.UNAUTHORIZED_ACCESS															
T.UNAUTHORIZED_PEER															
T.UNIDENTIFIED_ACTIONS					X										
T.UNKNOWN_STATE								X							
T.VIRUS															
P.ACCACT						X									X
P.ACCESS	X														X
P.ACCESS_BANNER											X				
P.ACCOUNTABILITY			X												

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH
P.ADMIN_ACCESS		X													
P.CRYPTOGRAPHIC_FUNCTIONS									X						
P.CRYPTOGRAPHY_VALIDATED									X	X					
P.DETECT						X								X	
P.INTEGRITY															
P.INTGTY															
P.MANAGE	X											X			X
P.PROTCT															
P.VULNERABILITY_ANALYSIS_TEST															

Table 10- Mapping of Security Assumptions, Threats, and Policies to Objectives

(Part 1 of 3)

	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SECURE_UPDATES	O.SELF_PROTECTION
A.ACCESS																
A.LOCATE																
A.MANAGE																
A.NO_TOE_BYPASS																
A.NOEVIL																
A.NOTRST																
A.PHYSICAL																
A.PROTCT																
T.ADDRESS_MASQUERADE						X										
T.ADMIN_ERROR					X								X			
T.ADMIN_ROGUE																
T.AUDIT_COMPROMISE											X					X
T.COMDIS									X							
T.COMINT	X								X							
T.CRYPTO_COMPROMISE											X					X
T.FLAWED_DESIGN																X
T.FLAWED_IMPLEMENTATION																
T.IMPCON																
T.INADVE																
T.INFLUX							X									
T.INTRUSION			X												X	
T.LOSSOF	X								X							
T.MALICIOUS_TSF_COMPROMISE					X						X					X
T.MASQUERADE														X		
T.MISACT																
T.MISUSE																
T.NOHALT																
T.POOR_TEST																
T.PRIVIL									X							
T.REPLAY										X						
T.RESIDUAL_DATA											X					

	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SECURE_UPDATES	O.SELF_PROTECTION
T.RESOURCE_EXHAUSTION												X				
T.SPOOFING																
T.UNATTENDED_SESSION														X		
T.UNAUTHORIZED_ACCESS						X										
T.UNAUTHORIZED_PEER								X								
T.UNIDENTIFIED_ACTIONS																
T.UNKNOWN_STATE				X									X			
T.VIRUS															X	
P.ACCACT																
P.ACCESS									X							
P.ACCESS_BANNER																
P.ACCOUNTABILITY														X		
P.ADMIN_ACCESS																
P.CRYPTOGRAPHIC_FUNCTIONS																
P.CRYPTOGRAPHY_VALIDATED											X					
P.DETECT																
P.INTEGRITY		X														
P.INTGTY	X															
P.MANAGE									X							
P.PROTCT							X									
P.VULNERABILITY_ANALYSIS_TEST																

Table 11 - Mapping of Security Assumptions, Threats, and Policies to Objectives

(Part 2 of 3)

	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST	OE.CREDEN	OE.CRYPTANALYTIC	OE.INSTAL	OE.INTROP	OE.NO_TOE_BYPASS	OE.PERSON	OE.PHYCAL	OE.PHYSICAL
A.ACCESS											X				
A.LOCATE														X	
A.MANAGE													X		
A.NO_TOE_BYPASS												X			
A.NOEVIL								X		X				X	
A.NOTRST								X						X	
A.PHYSICAL															X
A.PROTCT														X	
T.ADDRESS_MASQUERADE															
T.ADMIN_ERROR															
T.ADMIN_ROGUE															
T.AUDIT_COMPROMISE															
T.COMDIS															
T.COMINT															
T.CRYPTO_COMPROMISE									X						
T.FLAWED_DESIGN					X										
T.FLAWED_IMPLEMENTATION		X	X				X								
T.IMPCON										X					
T.INADVE															
T.INFLUX															
T.INTRUSION															
T.LOSSOF															
T.MALICIOUS_TSF_COMPROMISE					X										
T.MASQUERADE					X										
T.MISACT															
T.MISUSE															
T.NOHALT															
T.POOR_TEST			X				X								
T.PRIVIL															
T.REPLAY															
T.RESIDUAL_DATA															

	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST	OE.CREDEN	OE.CRYPTANALYTIC	OE.INSTAL	OE.INTROP	OE.NO_TOE_BYPASS	OE.PERSON	OE.PHYCAL	OE.PHYSICAL
T.RESOURCE_EXHAUSTION															
T.SPOOFING					X										
T.UNATTENDED_SESSION															
T.UNAUTHORIZED_ACCESS															
T.UNAUTHORIZED_PEER															
T.UNIDENTIFIED_ACTIONS															
T.UNKNOWN_STATE	X														
T.VIRUS						X									
P.ACCACT															
P.ACCESS															
P.ACCESS_BANNER															
P.ACCOUNTABILITY				X											
P.ADMIN_ACCESS					X										
P.CRYPTOGRAPHIC_FUNCTIONS															
P.CRYPTOGRAPHY_VALIDATED															
P.DETECT															
P.INTEGRITY															
P.INTGTY															
P.MANAGE								X		X			X		
P.PROTCT														X	
P.VULNERABILITY_ANALYSIS_TEST							X								

Table 12 - Mapping of Security Assumptions, Threats, and Policies to Objectives

(Part 3 of 3)

8.1.2 TOE Security Objectives Rationale

Table 13 provides detailed descriptions and rationale for the mapping from Security Objectives to Threats and Policies.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.ADDRESS_MASQUERADE A user on one interface may masquerade as a user on another interface to circumvent the TOE policy.	O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	O.MEDIATE (FDP_IFC.1(1), FDP_IFF.1(1), FDP_IFC.1(2), FDP_IFF.1(2) , FDP_IFC.1(3), FDP_IFF.1(3)) counters this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. The rules in each of the policies ensure that the network identifier in a network packet is in the set of network identifiers associated with a TOE's network interface. Therefore, if a user supplied a network identifier in a packet that was associated with a TOE network interface other than the one the user supplied the packet on, the packet would not be allowed to flow through the TOE, or access TOE services. This would, for example, prevent a user from sending a packet from the Internet claiming to be on a machine on the protected enclave.
T.ADMIN_ERROR An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.	O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management. O.ADMIN_ROLE The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely. O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	O.ROBUST_ADMIN_GUIDANCE helps to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner and to provide the administrator with instructions to ensure the TOE was not corrupted during the delivery process. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is unsecure. O.ADMIN_ROLE plays a role in mitigating this threat by limiting the functions an administrator can perform in a given role. For example, the Audit Administrator could not make a configuration mistake that would impact the information flow policy. O.MANAGE contributes to mitigating this threat by providing administrators the capability to view configuration settings. For example, if the Security Administrator made a mistake when configuring the ruleset, providing them the capability to view the rules affords them the ability to review the rules and discover any mistakes that might have been made.
T.ADMIN_ROGUE An administrator's intentions may become malicious resulting in user or TSF data being compromised.	O.ADMIN_ROLE The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.	O.ADMIN_ROLE mitigates this threat by restricting the functions available to an administrator. This is somewhat different than the part this objective plays in countering T.ADMIN_ERROR, in that this presumes that separate individuals will be assigned separate roles. If the Audit Administrator's intentions become malicious they would not be able to render the TOE unable to enforce its information flow policies. On the other hand, if the Security Administrator becomes malicious they could affect the information flow policy, but the Audit Administrator may be able to detect those actions.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.AUDIT_COMPROMISE A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information. O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	O.AUDIT_PROTECTION contributes to mitigating this threat by controlling access to the audit trail. No one is allowed to modify audit records, the Audit Administrator is the only one allowed to delete the audit trail. The TOE has the capability to prevent auditable actions from occurring if the audit trail is full. O.RESIDUAL_INFORMATION prevents a user not authorized to read the audit trail from access to audit information that might otherwise be persistent in a TOE resource (e.g., memory). By ensuring the TOE prevents residual information in a resource, audit information will not become available to any user or process except those explicitly authorized for that data. O.SELF_PROTECTION contributes to countering this threat by ensuring that the TSF can protect itself from users. If the TSF could not maintain and control its domain of execution, it could not be trusted to control access to the resources under its control, which includes the audit trail. Likewise, ensuring that the functions that protect the audit trail are always invoked is also critical to the mitigation of this threat.
T.COMDIS An unauthorized user may attempt to disclose the data collected by the TOE by bypassing a security mechanism.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.EXPORT When the TOE makes its Sensor data available to other IDS components, the TOE will ensure the confidentiality of the Sensor data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.EXPORT objective ensures that confidentiality of TOE data will be maintained. The O.PROTCT objective addresses this threat by providing TOE self-protection.
T.COMINT An unauthorized user may attempt to compromise the integrity of the data collected by the TOE by bypassing a security mechanism.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.INTEGR objective ensures no TOE data will be modified. The O.PROTCT objective addresses this threat by providing TOE self-protection.

Threat/Policy	Objectives Addressing the Threat	Rationale
	O.INTEGR The TOE must ensure the integrity of all audit and Sensor data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	
T.CRYPTO_COMPROMISE A malicious user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromise the cryptographic mechanisms and the data protected by those mechanisms.	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	O.RESIDUAL_INFORMATION mitigates the possibility of malicious users or processes from gaining inappropriate access to cryptographic data, including keys. This objective ensures that the cryptographic data does not reside in a resource that has been used by the cryptographic module and then reallocated to another process. O.SELF_PROTECTION contributes to countering this threat by ensuring that the TSF can protect itself from users. If the TSF could not maintain and control its domain of execution, it could not be trusted to control access to the resources under its control, which includes the cryptographic data and executable code.
T.FLAWED_DESIGN Unintentional or intentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a malicious user or program.	O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development. O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented. O.VULNERABILITY_ANALYSIS_TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.	O.CHANGE_MANAGEMENT plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. This includes controlling physical access to the TOE's development area, and having an automated configuration management system that ensures changes made to the TOE go through an approval process and only those persons that are authorized can make changes to the TOE's design and its documentation. O.SOUND_DESIGN counters this threat, to a degree, by requiring that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered. O.VULNERABILITY_ANALYSIS_TEST ensures that the design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design.
T.FLAWED_IMPLEMENTATION Unintentional or intentional errors in implementation of the TOE design may occur, leading to flaws that may be	O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.	O.CHANGE_MANAGEMENT plays a role in mitigating this threat in the same way that the flawed design threat is mitigated. By controlling who has access to the TOE's implementation representation and ensuring that changes to the implementation are analyzed and made in a controlled manner, the threat

Threat/Policy	Objectives Addressing the Threat	Rationale
exploited by a malicious user or program.	O.SOUND_IMPLEMENTATION The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented. O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements. O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.	of intentional or unintentional errors being introduced into the implementation are reduced. In addition to documenting the design so that implementers have a thorough understanding of the design, O.SOUND_IMPLEMENTATION requires that the developer's tools and techniques for implementing the design are documented. Having accurate and complete documentation, and having the appropriate tools and procedures in the development process helps reduce the likelihood of unintentional errors being introduced into the implementation. Although the previous three objectives help minimize the introduction of errors into the implementation, O.THOROUGH_FUNCTIONAL_ TESTING increases the likelihood that any errors that do exist in the implementation (with respect to the functional specification, high level, and low-level design) will be discovered through testing. O.VULNERABILITY_ANALYSIS_TEST helps reduce errors in the implementation that may not be discovered during functional testing. Ambiguous design documentation, and the fact that exhaustive testing of the external interfaces is not required may leave bugs in the implementation undiscovered in functional testing. Having an independent party perform a vulnerability analysis and conduct testing outside the scope of functional testing increases the likelihood of finding errors.
T.IMPCON The TOE may be susceptible to improper configuration by any user causing potential intrusions to go undetected.	O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data. O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.EADMIN objective ensures the TOE has all the necessary administrator functions to manage the product. The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. These objects are supported by the OE.INSTAL objective, which states the authorized administrators will configure the TOE properly.
T.INADVE Inadvertent activity and access may occur on an IT System which may result in the TOE being affected by unauthorised users[66].	O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions. O.IDACTS	The O.AUDITS and O.IDACTS objectives address this threat by requiring collection of audit and Sensor data.

[66] The IDSS PP threat was modified in order to identify a threat agent and the asset being attacked.

Threat/Policy	Objectives Addressing the Threat	Rationale
	The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	
T.INFLUX An unauthorized user may cause malfunction of the TOE by creating an influx of data that the TOE cannot handle.	O.OFLOWS The TOE must appropriately handle potential audit and Sensor data storage overflows .	The O.OFLOWS objective counters this threat by requiring the TOE handle data storage overflows.
T.INTRUSION A malicious agent may attempt to attack the TOE or one of the systems connected to the TOE by passing information which is designed to damage or compromise the system which receives the information.	O.INTRUSION The TOE will detect and prevent intrusion attacks which are contained within an information flow which arrives at any of the TOE network interfaces. O.SECURE_UPDATES The TOE shall provide a secure mechanism for the receipt of virus and intrusion signature updates..	The O.INTRUSION objective ensures that the TOE detects and prevents intrusion attacks which are directed at the TOE or any of the systems connected to the TOE. The O.SECURE_UPDATES objective ensures that the TOE becomes aware of newly discovered intrusion attack methods. Together these objectives mitigate the threat posed by intrusion attack techniques.
T.LOSSOF An unauthorized user may attempt to remove or destroy data collected by the TOE.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.INTEGR The TOE must ensure the integrity of all audit and Sensor data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.INTEGR objective ensures no TOE data will be deleted. The O.PROTCT objective addresses this threat by providing TOE self-protection.
T.MALICIOUS_TSF_COMPROMISE A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE. O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use. O.RESIDUAL_INFORMATION	O.DISPLAY_BANNER helps mitigate this threat by providing the Security Administrator the ability to remove product information (e.g., product name, version number) from a banner that is displayed to users. Having product information about the TOE provides an attacker with information that may increase their ability to compromise the TOE. O.MANAGE is necessary because an access control policy is not specified to control access to TSF data. This objective is used to dictate who is able to view and modify TSF data, as well as the behavior of TSF functions.

Threat/Policy	Objectives Addressing the Threat	Rationale
	The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure. O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.RESIDUAL_INFORMATION is necessary to mitigate this threat, because even if the security mechanisms do not allow a user to explicitly view TSF data, if TSF data were to inappropriately reside in a resource that was made available to a user, that user would be able to inappropriately view the TSF data. O.SELF_PROTECTION requires that the TSF be able to protect itself from tampering and that the security mechanisms in the TSF cannot be bypassed. Without this objective, there could be no assurance that users could not view or modify TSF data or TSF executables. O.TRUSTED_PATH plays a role in addressing this threat by ensuring that a trusted communication path exists between the TOE and authorized users (i.e., remote administrators, authorized IT entities). This ensures the transmitted data cannot be compromised or disclosed (e.g., encrypted) during the duration of the trusted path. The protection offered by this objective is limited to TSF data and security attributes (i.e., the data communication between peer TOEs via a VPN is protected by the VPN policy stated in FDP_IFC.1(3) and FDP_IFF.1(3) and FTP_ITC does not apply to VPN communications)
T.MASQUERADE A user may masquerade as an authorized user or an authorized IT entity to gain access to data or TOE resources.	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate. O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.ROBUST_TOE_ACCESS mitigates this threat by controlling the logical access to the TOE and its resources. By constraining how and when authorized users can access the TOE, and by mandating the type and strength of the authentication mechanism this objective helps mitigate the possibility of a user attempting to login and masquerade as an authorized user. In addition, this objective provides the administrator the means to control the number of failed login attempts a user can generate before an account is locked out, further reducing the possibility of a user gaining unauthorized access to the TOE. O.TRUSTED_PATH ensures that the communication path end points between the TOE and authorized users (remote administrators, authorized IT entities) are defined. This mechanism allows the TOE to be assured that it is communicating with an authorized user. This also ensures that the transmitted data cannot be disclosed (e.g., encrypted). The protection offered by this objective is limited to TSF data and security attributes.
T.MISACT Malicious activity, such as introductions of Trojan horses and viruses, may occur on an IT System which may result in the	O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions.	The O.AUDITS and O.IDACTS objectives address this threat by requiring collection of audit and Sensor data.

Threat/Policy	Objectives Addressing the Threat	Rationale
TOE being affected by unauthorised users[67].	O.IDACTS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	
T.MISUSE Unauthorized accesses and activity indicative of misuse may occur on an IT System which may result in the TOE being affected by unauthorised users[68].	O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions. O.IDACTS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.AUDITS and O.IDACTS objectives address this threat by requiring collection of audit and Sensor data.
T.NOHALT An unauthorized user may attempt to compromise the continuity of the TOE's collection functionality by halting execution of the TOE.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.IDACTS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.IDACTS objective addresses this threat by requiring the TOE to collect all events, including those attempts to halt the TOE.
T.POOR_TEST Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered.	O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements. O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.	Design analysis determines that TOE's documented design satisfies the security functional requirements. In order to ensure the TOE's design is correctly realized in its implementation, the appropriate level of functional testing of the TOE's security mechanisms must be performed during the evaluation of the TOE. O.THOROUGH_FUNCTIONAL_ TESTING ensures that adequate functional testing is performed to ensure the TSF satisfies the security functional requirements and demonstrates that the TOE's security mechanisms operate as documented. While functional testing serves an important purpose, it does not ensure the TSFI cannot be used in unintended ways to circumvent the TOE's security

[67] The IDSS PP threat was modified in order to identify a threat agent and the asset being attacked.

[68] The IDSS PP threat was modified in order to identify a threat agent and the asset being attacked.

Threat/Policy	Objectives Addressing the Threat	Rationale
	O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.	policies. O.VULNERABILITY_ANALYSIS_TEST addresses this concern by requiring a vulnerability analysis be performed in conjunction with testing that goes beyond functional testing. This objective provides a measure of confidence that the TOE does not contain security flaws that may not be identified through functional testing. While these testing activities are a necessary activity for successful completion of an evaluation, this testing activity does not address the concern that the TOE continues to operate correctly and enforce its security policies once it has been fielded. Some level of testing must be available to end users to ensure the TOE's security mechanisms continue to operate correctly once the TOE is fielded O.CORRECT_TSF_OPERATION ensures that once the TOE is installed at a customer's location, the capability exists that the integrity of the TSF (hardware and software) can be demonstrated, and thus providing end users the confidence that the TOE's security policies continue to be enforced.
T.PRIVIL An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.PROTCT objective addresses this threat by providing TOE self-protection.
T.REPLAY A user may gain inappropriate access to the TOE by replaying authentication information, or may cause the TOE to be inappropriately configured by replaying TSF data or security attributes (captured as it was transmitted during the course of legitimate use).	O.REPLAY_DETECTION The TOE will provide a means to detect and reject the replay of TSF data and security attributes.	O.REPLAY_DETECTION prevents a user from replaying TSF data and security attributes (e.g., TSF data or security attributes transmitted between a remote administrator, an authorized IT entity and the TOE) that could leave the TOE in a configuration that the administrative staff did not intend (e.g., an administrator modifies the auditable events to be recorded and a user captures that traffic. At a later date the administrator determines that the new set of auditable events is not sufficient and again modifies the events to be audited. The user then replays the earlier audit event configuration.).

Threat/Policy	Objectives Addressing the Threat	Rationale
T.RESIDUAL_DATA A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	O.RESIDUAL_INFORMATION counters this threat by ensuring that TSF data and user data is not persistent when resources are released by one user/process and allocated to another user/process. This means that network packets will not have residual data from another packet due to the padding of a packet.
T.RESOURCE_EXHAUSTION A malicious process or user may block others from system resources (e.g., connection state tables) via a resource exhaustion denial of service attack.	O.RESOURCE_SHARING The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; TCP connections used by proxies).	O.RESOURCE_SHARING mitigates this threat by requiring the TOE to provide controls over connection-oriented resources. These controls provide the administrator ability to specify which network identifiers have access to the TOE's connection-oriented resources over a time period that is specified by the administrator. This objective also addresses the denial-of-service attack of a user attempting to exhaust the connection-oriented resources by generating a large number of half-open connections (e.g., SYN attack).
T.SPOOFING An entity may mis-represent itself as the TOE to obtain authentication data.	O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	It is possible for an entity other than the TOE (a subject on the TOE, or another IT entity) to provide an environment that may lead a user to mistakenly believe they are interacting with the TOE thereby fooling the user into divulging identification and authentication information. O.TRUSTED_PATH mitigates this threat by ensuring users have the capability to ensure they are communicating with the TOE when providing identification and authentication data to the TOE.
T.UNATTENDED_SESSION A user may gain unauthorized access to an unattended session.	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	O.ROBUST_TOE_ACCESS helps to mitigate this threat by including mechanisms that place controls on user's sessions. Local administrator's sessions are locked and remote sessions are dropped after a Security Administrator defined time period of inactivity. Locking the local administrator's session reduces the opportunity of someone gaining unauthorized access the session when the console is unattended. Dropping the connection of a remote session (after the specified time period) reduces the risk of someone accessing the remote machine where the session was established, thus gaining unauthorized access to the session.
T.UNAUTHORIZED_ACCESS A user may gain access to services (either on the TOE or by sending data through the TOE) for which they are not authorized according to the TOE security policy.	O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	O.MEDIATE works to mitigate this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. One of the rules ensures that the network identifier in a packet is in the set of network identifiers associated with a TOE's network interface. Therefore, if a user supplied a network identifier in a packet that purported to originate from a network associated with a TOE network interface other than the one the user supplied the packet on, the packet would not be allowed to flow through the TOE, or access TOE services. The VPN policy ensures that user data being sent between PEER TOEs is encrypted if there is a rule (specified by the Security Administrator)

Threat/Policy	Objectives Addressing the Threat	Rationale
		that states data is to be encrypted between those two hosts. The VPN policy allows the administrator to specify for each originating host (identified by IP address), which destination addresses must be accessed through a VPN (using ESP tunnel mode) and which destination addresses may be access without VPN encryption. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrative consoles. The consoles include the local TOE console and any active remote administrative sessions. If an administrator is not currently accessing the TOE, the message is stored and immediately displayed the next time an administrator accesses the TOE. Another rule provides further granularity of access control by enabling the administrator to control the source and destination address, destination port, AND protocol. By implementing this level of access control an attacker would not be allowed access to other hosts residing on the protected network. Additionally, the TOE maintains "state" information of all approved connections. This function ensures that each packet arriving at a TOE interface purporting to be part of an approved connection through the TOE, is checked against a current and valid list of connection parameters (e.g. for a TCP/IP connection; source and destination address, ports, SYN and ACK numbers, flags, etc.) prior to allowing the packet through or to the TOE. The TOE requires successful authentication through a protected communication path (with account lock-out capability) to the TOE prior to gaining access to certain services on or mediated by the TOE. By implementing "protected" authentication to gain access to these services, an attacker's opportunity to successfully conduct a man-in-the-middle and/or password guessing attack is greatly reduced. Lastly, the TSF must ensure that all configured enforcement functions (authentication, access control rules, etc.) must be invoked prior to allowing a user to gain access to TOE or TOE mediated services. The TOE restricts the ability to modify the security attributes associated with access control rules, access to authenticated and unauthenticated services, etc to the Security Administrator. This feature ensures that no other user can modify the information flow policy to bypass the intended TOE security policy.
T.UNAUTHORIZED_PEER An unauthorized IT entity may attempt to establish a security association with the TOE.	O.PEER_AUTHENTICATION The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.	O.PEER_AUTHENTCATION mitigates this threat by requiring that the TOE implement the Internet Key Exchange protocol, as specified in RFC2409, to establish a secure, authenticated channel between the TOE and another remote VPN endpoint before establishing a security association with that remote endpoint.
T.UNIDENTIFIED_ACTIONS The administrator may fail to notice potential security violations, thus	O.AUDIT_REVIEW The TOE will provide the capability to selectively view audit information, and alert the administrator of	O.AUDIT_REVIEW helps to mitigate this threat by providing the Security Administrator with a required minimum set of configurable audit events that could indicate a potential security violation. By

Threat/Policy	Objectives Addressing the Threat	Rationale
limiting the administrator's ability to identify and take action against a possible security breach.	identified potential security violations.	configuring these auditable events, the TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number of information policy flow failures, self-test failures, etc.) and immediately notifies all TOE administrators once an event has occurred or a set threshold has been met. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrative consoles. The consoles include the local TOE console and any active remote administrative sessions. If an administrator is not currently logged into the TOE, the message is stored and immediately displayed the next time an administrator logs into the TOE. This message is displayed to all administrative roles and will remain on the screen for each administrative role until each administrative role acknowledges the message. In addition to displaying the potential security violation, the message must contain all audit records that generated the potential security violation. By enforcing the message content and display, this objective provides assurance that a TOE administrator will be notified of a potential security violation. The TOE can also be configured to generate an audible alarm, which may alert administrators who are not sitting at their administrative workstation or console. The TOE also requires an Audit Administrative role. This role is restricted to Audit record review and the deletion of the audit trail for maintenance purposes. A search and sort capability provides an efficient mechanism for the Audit Administrator to view pertinent audit information.
T.UNKNOWN_STATE When the TOE is initially started or restarted after a failure, design flaws, or improper configurations may cause the security state of the TOE to be unknown which may result in the TOE being affect by unauthorised users[69].	O.MAINT_MODE The TOE shall provide a mode from which recovery or initial startup procedures can be performed. O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment. O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented. O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	O.MAINT_MODE helps to mitigate this threat by ensuring that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. After a failure, the TOE enters a state that disallows traffic flow and requires an administrator to follow documented procedures to return the TOE to a secure state. O.CORRECT_TSF_OPERATION counters this threat by ensuring that the TSF runs a suite of tests to successfully demonstrate the correct operation of the TSF's underlying abstract machine (hardware and software), the TSF, and the TSF's cryptographic components at initial startup of the TOE. In addition to ensuring that the TOE's security state can be verified, the Security Administrator can verify the integrity of the TSF's data and stored code as well as the TSF's cryptographic data and stored code. O.SOUND_DESIGN works to mitigate this threat by requiring that the TOE developers provide accurate and complete design documentation of the security mechanisms in the TOE, including a security model. By providing this documentation, the possible

[69] The IDSS PP threat was modified in order to identify a threat agent and the asset being attacked.

Threat/Policy	Objectives Addressing the Threat	Rationale
		security states of the TOE at startup or restart after failure should be documented and understood, thereby reducing the possibility that the TOE's security state could be unknown to users of the TOE. O.ROBUST_ADMIN_GUIDANCE provides administrative guidance for the secure start-up of the TOE as well as guidance to configure and administer the TOE securely. This guidance provides administrators with the information necessary to ensure that the TOE is started and initialized in a secure manner. The guidance also provides information about the corrective measure necessary when a failure occurs (i.e., how to bring the TOE back into a secure state).
T.VIRUS A malicious agent may attempt to pass a virus through or to the TOE.	O.SECURE_UPDATE The TOE shall provide a secure mechanism for the receipt of virus and intrusion signature updates. O.VIRUS The TOE will detect and block viruses contained within an information flow which arrives at any of the TOE network interfaces.	The O.VIRUS objective ensures that the TOE detects and blocks viruses which are contained in any information flow which reaches one of the TOE's network interfaces. The O.SECURE_UPDATES objective ensures that the TOE becomes aware of newly discovered viruses. Together these objectives mitigate the threat of viruses.
P.ACCACT Users of the TOE shall be accountable for their actions within the IDS.	O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions. O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data.	The O.AUDITS objective implements this policy by requiring auditing of all data accesses and use of TOE functions. The O.IDAUTH objective supports this objective by ensuring each user is uniquely identified and authenticated.
P.ACCESS All data collected by the IDS shall only be used for authorized purposes.	O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.PROTCT objective provides for TOE self-protection.

Threat/Policy	Objectives Addressing the Threat	Rationale
P.ACCESS_BANNER The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	O.DISPLAY_BANNER satisfies this policy by ensuring that the TOE displays a Security Administrator configurable banner that provides all users with a warning about the unauthorized use of the TOE.
P.ACCOUNTABILITY The authorized users of the TOE shall be held accountable for their actions within the TOE.	O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users. O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	O.AUDIT_GENERATION addresses this policy by providing the Security Administrator with the capability of configuring the audit mechanism to record the actions of a specific user, or review the audit trail based on the identity of the user. Additionally, the administrator's ID is recorded when any security relevant change is made to the TOE (e.g. access rule modification, start-stop of the audit mechanism, establishment of a trusted channel, etc.). O.TIME_STAMPS plays a role in supporting this policy by requiring the TOE to provide a reliable time stamp (configured locally by the Security Administrator). The audit mechanism is required to include the current date and time in each audit record. All audit records that include the user ID, will also include the date and time that the event occurred. O.ROBUST_TOE_ACCESS supports this policy by requiring the TOE to identify and authenticate all authorized users prior to allowing any TOE access or any TOE mediated access on behalf of those users. While the user ID of authorized users can be assured, since they are authenticated, this ST allows unauthenticated users to access the TOE and the identity is then a presumed network identifier (e.g., IP address).
P.ADMIN_ACCESS Administrators shall be able to administer the TOE both locally and remotely through protected communications channels.	O.ADMIN_ROLE The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely. O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.ADMIN_ROLE supports this policy by requiring the TOE to provide mechanisms (e.g., local authentication, remote authentication, means to configure and manage the TOE both remotely and locally) that allow remote and local administration of the TOE. This is not to say that everything that can be done by a local administrator must also be provided to the remote administrator. In fact, it may be desirable to have some functionality restricted to the local administrator (e.g., setting the ruleset). O.TRUSTED_PATH satisfies this policy by requiring that each remote administrative session (all administrative roles) is authenticated and conducted via a secure channel. Additionally, all authorized IT entities (e.g. authentication/certificate servers) must adhere to the same requirements as the remote administrator.

Threat/Policy	Objectives Addressing the Threat	Rationale
P.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.	O.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.	O.CRYPTOGRAPHIC_FUNCTIONS implements this policy, requiring a combination of FIPS-validation and non-FIPS-validated cryptographic mechanisms that are used to provide encryption/decryption services, as well as digital signature functions. Functions include symmetric encryption and decryption, digital signatures, as well as key generation and establishment functions.
P.CRYPTOGRAPHY_VALIDATED Where the TOE requires FIPS-approved security functions, only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key distribution, and random number generation services).	O.CRYPTOGRAPHY_VALIDATED The TOE shall use NIST FIPS 140-2 validated cryptomodules for cryptographic services implementing FIPS-approved security functions and random number generation services used by cryptographic functions. O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated or upon completion of a function that residual biometric data could not be reused.	O.CRYPTOGRAPHY_VALIDATED satisfies this policy by requiring the TOE to implement NIST FIPS validated cryptographic services. These services will provide confidentiality and integrity protection of TSF data while in transit to remote parts of the TOE. O.RESIDUAL_INFORMATION satisfies this policy by ensuring that cryptographic data are cleared from resources that are shared between users. Keys must be zeroized according to FIPS 140-2.
P.DETECT All events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets must be collected.	O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions. O.IDACTS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.AUDITS and O.IDACTS objectives require collection of audit and Sensor data.
P.INTEGRITY The TOE shall support the IETF *Internet Protocol Security Encapsulating Security Payload* (IPSec ESP) as specified in RFC 2406. Sensitive information transmitted to a peer TOE shall apply integrity mechanisms as specified in *Use of HMAC-SHA-1-96 within ESP and AH* (RFC 2404).	O.INTEGRITY The TOE must be able to protect the integrity of data transmitted to a peer TOE via encryption and provide IPSec authentication for such data. Upon receipt of data from a peer TOE, the TOE must be able to decrypt the data and verify that the received data accurately represents the data that was originally transmitted.	O.INTEGRITY satisfies this policy by ensuring that all IPSec encrypted data received from a peer TOE is properly decrypted and authentication verified.
P.INTGTY Data collected by the TOE shall be protected from modification.	O.INTEGR The TOE must ensure the integrity of all audit and Sensor data.	The O.INTEGR objective ensures the protection of data from modification.

Threat/Policy	Objectives Addressing the Threat	Rationale
P.MANAGE The TOE shall only be managed by authorized users.	O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data. O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data. O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data. O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.EADMIN objective ensures there is a set of functions for administrators to use, and is supported by the OE.PERSON objective, which ensures competent administrators will manage the TOE. The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.PROTCT objective provides for TOE self-protection.
P.PROTCT The TOE shall be protected from unauthorized accesses and disruptions of collection activities.	O.OFLOWS The TOE must appropriately handle potential audit and Sensor data storage overflows.	The O.OFLOWS objective requires the TOE handle disruptions. It is supported by the OE.PHYCAL objective, which protects the TOE from unauthorized physical modifications.
P.VULNERABILITY_ ANALYSIS_TEST The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a low attack potential.	O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.	O.VULNERABILITY_ANALYSIS_TEST satisfies this policy by ensuring that an independent analysis is performed on the TOE and penetration testing based on that analysis is performed. Having an independent party perform the analysis helps ensure objectivity and eliminates preconceived notions of the TOE's design and implementation that may otherwise affect the thoroughness of the analysis. The level of analysis and testing requires that an attacker with a low attack potential cannot compromise the TOE's ability to enforce its security policies.

Table 13 - Security Objectives to Threats and Policies Mappings

8.1.3 Rationale for the Security Objectives and Security Functional Requirements for the Environment

Table 14 provides detailed descriptions and rationale for the mapping from Security Objectives to Threats and Policies. Where relevant, the objectives are also mapped to the Security Functional Requirements for the Environment.

Assumption	Objectives Addressing the Assumption	Rationale
A.ACCESS The TOE has access to all the IT System data it needs to perform its functions.	OE.INTROP The TOE is interoperable with the IT System it monitors and other IDS components within its IDS.	The OE.INTROP objective ensures the TOE has the needed access.
A.LOCATE The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	OE.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL provides for the physical protection of the TOE.
A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.	OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the Sensor.	The OE.PERSON objective ensures all authorized administrators are qualified and trained to manage the TOE.
A.NO_TOE_BYPASS Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.	OE.NO_TOE_BYPASS Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.	OE.NO_TOE_BYPASS is a restatement of the assumption, and therefore traces to the assumption trivially and is suitable for covering the assumptions.
A.NOEVIL The authorized administrators are not careless, wilfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.	OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security. OE.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack. OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.INSTAL objective ensures that the TOE is properly installed and operated and the OE.PHYCAL objective provides for physical protection of the TOE by authorized administrators. The OE.CREDEN objective supports this assumption by requiring protection of all authentication data.
A.NOTRST The TOE can only be accessed by authorized users.	OE.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack. OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.PHYCAL objective provides for physical protection of the TOE to protect against unauthorized access. The OE.CREDEN objective supports this assumption by requiring protection of all authentication data.

Assumption	Objectives Addressing the Assumption	Rationale
A.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.	OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.	OE.PHYSICAL is a restatement of the assumption, and therefore traces to the assumption trivially and is suitable for covering the assumptions.
A.PROTCT The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.	OE.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL provides for the physical protection of the TOE hardware and software.
T.CRYPTO_COMPROMISE A malicious user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromise the cryptographic mechanisms and the data protected by those mechanisms.	OE.CRYPTANALYTIC Cryptographic methods used in the IT environment shall be interoperable with the TOE, should be FIPS 140-2 validated and should be resistant to cryptanalytic attacks (i.e., will be of adequate strength to protect unclassified Mission Support, Administrative, or Mission Critical data).	The IT environment security objective OE.CRYPTANALYTIC is necessary to play a role in countering the threat T.CRYPTO_COMPROMISE. This IT environment security objective ensures that the cryptographic methods used in the IT environment are interoperable with the mechanisms provided by the TOE. The IT environment's cryptographic methods should be independently validated to be FIPS 140-2 compliant. OE.CRYPTANALYTIC maps to the IT environmental iterated requirements FTP_ITC.1 (ensuring that encryption is used on the communication channel between authorized IT entities and the TOE), and FTP_TRP (ensuring that an administrator can be assured that they are communicating with the TOE).
T.IMPCON The TOE may be susceptible to improper configuration by any user causing potential intrusions to go undetected.	OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.	The OE.INSTAL objective states the authorized administrators will configure the TOE properly. This supports the objectives O.EADMIN, O.IDAUTH, and O.ACCESS
P.MANAGE The TOE shall only be managed by authorized users.	OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the Sensor. OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security. OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.PERSON objective ensures competent administrators will manage the TOE and supports the O.EADMIN objective, which ensures there is a set of functions for administrators to use. The OE.INSTAL objective supports the O.PERSON objective by ensuring administrator follow all provided documentation and maintain the security policy. The OE.CREDEN objective requires administrators to protect all authentication data.
P.PROTCT	OE.PHYCAL	The OE.PHYCAL objective protects the TOE from unauthorized physical modifications, and thus

Assumption	Objectives Addressing the Assumption	Rationale
The TOE shall be protected from unauthorized accesses and disruptions of collection activities.	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	supports the O.OFLOWS objective.

Table 14 – Rationale for the Objectives and Security Functional Requirements for the Environment

8.2 RATIONALE FOR TOE SECURITY REQUIREMENTS

Table 16 provides a bi-directional mapping of Security Functional Requirements and Security Assurance Requirements to Security Objectives. It shows that each of the objectives for the TOE is addressed by at least one of the functional or assurance requirements, and that each of the functional and assurance requirements addresses at least one of the objectives for the TOE.

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SELF_PROTECTION	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST
FAU_ARP.1					X																																
FAU_ARP_ACK_EXP.1					X															X																	
FAU_GEN.1			X			X																															
FAU_GEN.2			X																																		
FAU_SAA.1					X																																
FAU_SAR.1					X							X								X																	
FAU_SAR.2	X			X											X					X																	
FAU_SAR.3					X							X								X																	
FAU_SEL.1			X			X						X								X																	
FAU_STG.2	X			X											X	X				X		X		X													
FAU_STG.3			X	X																X																	
FAU_STG.4			X	X		X														X		X															
FAV_ACT_EXP.1																																				X	
FCS_BCM._EXP.1										X																											
FCS_CKM.1(1)									X																												
FCS_CKM.1(2)									X																												
FCS_CKM.1(3)									X																												
FCS_CKM.1(4)									X																												
FCS_CKM.1(5)									X														X														
FCS_CKM.2(1)									X																												
FCS_CKM.2(2)									X																												
FCS_CKM.4									X																	X											
FCS_COP.1(1)									X																												

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SELF_PROTECTION	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST
FCS_COP.1(2)									X																												
FCS_COP.1(3)									X																												
FCS_COP.1(4)									X																												
FDP_IFC.1(1)																					X																
FDP_IFC.1(2)																					X																
FDP_IFC.1(3)																					X																
FDP_IFC.1(4)																	X																				
FDP_IFF.1(1)																					X																
FDP_IFF.1(2)																					X																
FDP_IFF.1(3)																					X																
FDP_IFF.1(4)																	X																				
FDP_RIP.2																										X											
FIA_AFL.1																													X								
FIA_ATD.1(1)															X														X								
FIA_ATD.1(2)															X														X								
FIA_ATD.1(3)															X														X								
FIA_UAU.1(1)	X														X														X								
FIA_UAU.1(2)	X														X														X								
FIA_UAU.2																													X								
FIA_UAU.5																													X								
FIA_UID.2	X														X														X								
FIA_USB.1			X																																		
FIP_ACT_EXP.1																		X																			
FMT_MOF.1(1)	X														X					X				X													
FMT_MOF.1(2)	X			X											X					X				X													
FMT_MOF.1(3)	X				X										X					X				X													
FMT_MOF.1(4)	X				X										X					X				X													
FMT_MOF.1(5)	X				X										X					X				X													
FMT_MOF.1(6)	X														X					X				X													
FMT_MOF.1(7)	X														X					X				X			X										
FMT_MOF.1(8)	X														X					X				X													
FMT_MOF.1(9)	X														X					X				X													
FMT_MOF.1(10)	X														X					X				X													
FMT_MOF.1(11)	X														X					X				X													
FMT_MOF.1(12)	X														X					X				X													
FMT_MOF.1(13)	X														X					X				X													
FMT_MSA.1																				X																	
FMT_MSA.2																				X																	
FMT_MSA.3(1)																				X																	
FMT_MSA.3(2)																				X																	
FMT_MTD.1(1)	X														X	X				X				X													
FMT_MTD.1(2)	X														X	X				X				X													
FMT_MTD.1(3)	X														X	X				X				X										X			
FMT_MTD.1(4)	X														X	X				X				X													
FMT_MTD.1(5)	X														X	X				X				X													
FMT_MTD.1(6)	X														X	X				X				X													
FMT_MTD.1(7)	X														X	X				X				X													
FMT_MTD.1(8)	X														X	X				X				X													

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SELF_PROTECTION	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST
FMT_MTD.1(9)	X														X	X				X				X													
FMT_MTD.2(1)																											X										
FMT_MTD.2(2)																											X										
FMT_REV.1																					X																
FMT_SMR.2		X													X																						
FPT_AMT.1								X																													
FPT_FLS.1																														X							
FPT_ITA.1													X																								
FPT_ITC.1													X			X																					
FPT_ITI.1													X			X																					
FPT_RCV.1																			X																		
FPT_RPL.1																									X												
FPT_RVM.1						X						X			X	X					X			X						X							
FPT_SEP.2						X						X			X	X								X						X							
FPT_STM.1						X																												X			
FPT_TST.1(1)								X																													
FPT_TST.1(2)								X																													
FRU_FLT.1																														X							
FRU_RSA.1(1)																											X										
FRU_RSA.1(2)																											X										
FTA_SSL.1																													X								
FTA_SSL.2																													X								
FTA_SSL.3																													X								
FTA_TAB.1											X																										
FTA_TSE.1																													X								
FTP_ITC.1(1)																														X					X		
FTP_ITC.1(2)																														X					X		
FTP_TRP.1(1)																														X					X		
FTP_TRP.1(2)																														X					X		
IDS_COL_EXP.1														X																							
IDS_RDR_EXP.1	X											X			X																						
IDS_STG_EXP.1	X														X	X						X		X													
IDS_STG_EXP.2																						X															
FTP_ITC.1(3) (ENV)																														X					X		
FTP_ITC.1(4) (ENV)																														X					X		
FTP_TRP.1(3) (ENV)																														X					X		
FTP_TRP.1(4) (ENV)																														X					X		
ACM_AUT.1							X																														
ACM_CAP.4							X																														
ACM_SCP.2							X																														
ADO_DEL.2																												X									
ADO_IGS.1																												X									
ADV_FSP.2																															X						
ADV_HLD.2																															X						
ADV_IMP.1																																X					
ADV_LLD.1																															X	X					
ADV_RCR.1																															X	X					
ADV_SPM.1																															X						

	O.ACCESS	O.ADMIN_ROLE	O.AUDIT_GENERATION	O.AUDIT_PROTECTION	O.AUDIT_REVIEW	O.AUDITS	O.CHANGE_MANAGEMENT	O.CORRECT_TSF_OPERATION	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHY_VALIDATED	O.DISPLAY_BANNER	O.EADMIN	O.EXPORT	O.IDACTS	O.IDAUTH	O.INTEGR	O.INTEGRITY	O.INTRUSION	O.MAINT_MODE	O.MANAGE	O.MEDIATE	O.OFLOWS	O.PEER_AUTHENTICATION	O.PROTCT	O.REPLAY_DETECTION	O.RESIDUAL_INFORMATION	O.RESOURCE_SHARING	O.ROBUST_ADMIN_GUIDANCE	O.ROBUST_TOE_ACCESS	O.SELF_PROTECTION	O.SOUND_DESIGN	O.SOUND_IMPLEMENTATION	O.THOROUGH_FUNCTIONAL_TESTING	O.TIME_STAMPS	O.TRUSTED_PATH	O.VIRUS	O.VULNERABILITY_ANALYSIS_TEST
AGD_ADM.1																												X									
AGD_USR.1																												X									
ALC_DVS.1							X																														
ALC_FLR.3							X																														
ALC_LCD.1							X																														
ALC_TAT.1																																X					
ATE_COV.2																																	X				
ATE_DPT.1																																	X				
ATE_FUN.1																																	X				
ATE_IND.2																																	X				
AVA_MSU.2																												X									
AVA_SOF.1																													X								
AVA_VLA.2																																					X

Table 15 - Security Requirements Rationale Summary

Table 16 provides detailed descriptions and rationale for the mapping from Security Objectives to TOE Security Functional Requirements.

Objective	Requirements Addressing the Objective	Rationale
O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	FAU_SAR.2 FAU_STG.2 FIA_UAU.1(1), FIA_UAU.1(2) FIA_UID.2 FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(7), FMT_MOF.1(8), FMT_MOF.1(9), FMT_MOF.1(10), FMT_MOF.1(11), FMT_MOF.1(12),	The TOE is required to restrict the review of audit data to those granted with explicit read-access [FAU_SAR.2]. The Sensor is required to restrict the review of collected Sensor data to those granted with explicit read access [IDS_RDR_EXP.1]. The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The Sensor is required to protect the Sensor data collected from an IT System from any modification and unauthorized deletion [IDS_STG_EXP.1]. Users authorized to access the TOE are defined using an identification and authentication process [FIA_UID.2, FIA_UAU.1]. The TOE is required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. Only authorized administrators of the Sensor may query and add Sensor and audit data, and authorized administrators of the TOE may query and modify all other TOE data [FMT_MTD.1].

Objective	Requirements Addressing the Objective	Rationale
	FMT_MOF.1(13) FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7), FMT_MTD.1(8), FMT_MTD.1(9) IDS_RDR_EXP.1 IDS_STG_EXP.1	
O.ADMIN_ROLE The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.	FMT_SMR.2	FMT_SMR.2 requires that three roles exist for administrative actions: the Security Administrator, who is responsible for configuring the TOE's security policies; the Cryptographic Administrator, who is restricted to managing the security data that is critical to the cryptographic operations; and the Audit Administrator, who is restricted to reading and deleting the audit trail. The TSF is able to associate a human user with one or more roles and these roles isolate administrative functions in that the functions of these roles do not overlap. The functionality of the roles, as defined by this ST, is predicated on the notion that once the TOE has been setup and is running in a stable configuration the Security Administrator would not be required to frequently administer the TOE. The Audit Administrator will probably be logging into the TOE most often to review the audit trail. Restricting the Audit Administrator's capabilities thus reduces the potential harm that could occur due to an error, or the execution of malicious code.
O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	FAU_GEN.1 FAU_GEN.2 FAU_SEL.1 FAU_STG.3 FAU_STG.4 FIA_USB.1	FAU_GEN.1 defines the set of events that the TOE must be capable of recording. This requirement ensures that the Security Administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. There is a minimum of information that must be present in every audit record and this requirement defines that, as well as the additional information that must be recorded for each auditable event. FAU_GEN.2 ensures that the audit records associate a user identity with the auditable event. In the case of authorized users, the association is accomplished with the userid. In all other cases, the association is based on the source network identifier, which is presumed to be the correct identity, but cannot be confirmed since these subjects are not authenticated. FAU_SEL.1 allows the Security Administrator to configure which auditable events will be recorded in the audit trail. This provides the administrator with the flexibility in recording only those events that are deemed necessary by site policy, thus reducing the amount of resources consumed by the audit mechanism. FAU_STG.3 requires that the administrators are alerted when the audit trail exceeds a capacity threshold established by the Security Administrator. This ensures that the Security Administrator has the opportunity to manage the audit trail before it becomes full and

Objective	Requirements Addressing the Objective	Rationale
		the avoiding the possible loss of audit data. FAU_STG.4 allows the Security Administrator to configure the TOE so that if the audit trail does become full, either the TOE will prevent any events from occurring (other than actions taken by the Security Administrator or Audit Administrator) that would generate an audit record (e.g., depending on the FAU_SEL.1 configuration, traffic may no longer flow through the TOE) or the audit mechanism will overwrite the oldest audit records with new records. FIA_USB.1 plays a role is satisfying this objective by requiring a binding of security attributes associated with users that are authenticated with the subjects that represent them in the TOE. This only applies to authorized users, since the identity of unauthenticated users cannot be confirmed. Therefore, the audit trail may not always have the proper identity of the subject that causes an audit record to be generated (e.g., presumed network address of an unauthenticated user may be a spoofed address).
O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information.	FAU_SAR.2 FAU_STG.2 FAU_STG.3 FAU_STG.4 FMT_MOF.1(2)	FAU_SAR.2 restricts the ability to read the audit trail to the administrators, thus preventing the disclosure of the audit data to any other user. However, the TOE is not expected to prevent the disclosure of audit data if it has been archived or saved in another form (e.g., moved or copied to an ordinary file). The FAU_STG family dictates how the audit trail is protected. FAU_STG.2 restricts the ability to delete audit records to the Audit Administrator or if the option of overwriting old audit records is chosen by the Security Administrator in FAU_STG.4, the audit data may be deleted/overwritten. This helps ensure that audit records are kept until the Audit Administrator deems they are no longer necessary. This requirement also ensures that no one has the ability to modify audit records (e.g., edit any of the information contained in an audit record). This ensures the integrity of the audit trail is maintained. FMT_MOF.1(2) restricts the capability to modify the behavior of the audit and alarm functions to the Security Administrator. While the Audit Administrator has the capability to choose how they will review the audit trail, they do not have the capability to select what events are audited. This requirement ensures that only the Security Administrator can turn audit on or off, thus ensuring user's actions are audited according to a site-defined policy.
O.AUDIT_REVIEW The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.	FAU_SAA.1 FAU_ARP.1 FAU_ARP_ACK_EXP.1 FAU_SAR.3 FAU_SAR.1 FMT_MOF.1(3) FMT_MOF.1(4)	FAU_SAA.1 defines the events that indicate a potential security violation and will generate an alarm. The triggers for these events are configurable, for the most part, by the Security Administrator. The exception is that any failure of the TSF self-tests will generate an alarm. FAU_ARP.1 requires that the alarm be displayed at the local administrative console and at the remote administrative console(s) when an administrative session exists. For the latter, the alarm is sent to each role either during an established session or upon session establishment. This is required to ensure that no matter which role an administrator logs into the alarm will be received as soon as possible. This requirement also dictates the information that must be displayed with the alarm. The potential security violation is identified in the alarm, as are the contents of the audit records of the events that accumulated and triggered the alarm.

Objective	Requirements Addressing the Objective	Rationale
	FMT_MOF.1(5)	The information in the audit records is necessary it allows the administrators to react to the potential security violation without having to search through the audit trail looking for the related events. The TOE can also be configured to generate an audible alarm, which notifies administrators that are not attending their workstations of the potential violation. FAU_ARP_ACK_EXP.1 requires that the alarm be displayed at the local administrative console until it is acknowledged by an administrator, and at the remote administrative console(s) until it has been acknowledged by an administrator acting in each of the administrative roles. This ensures that the alarm message will not be obstructed and the administrators will be alerted of a potential security violation. The audible alarm, if configured, sounds continuously until acknowledged by an administrator. FAU_SAR.1 provides the administrators with the capability to read all the audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the administrators to interpret the audit trail, which is subject to interpretation. It is expected that the audit information be presented in such a way that the administrators can examine an audit record and have the appropriate information (that required by FAU_GEN.2) presented together to facilitate the analysis of the audit review. FAU_SAR.3 complements FAU_SAR.1 by providing the administrators the flexibility to specify criteria that can be used to search or sort the audit records residing in the audit trail. FAU_SAR.3 requires the administrators be able to establish the audit review criteria based on a userid and source subject identity, so that the actions of a user can be readily identified and analyzed. The criteria also includes a destination subject identity so the administrators can determine what network traffic is destined for an individual machine. Allowing the administrators to perform searches or sort the audit records based on dates, times, subject identities, destination service identifier, or transport layer protocol provides the capability to extract the network activity to what is pertinent at that time in order facilitate the administrator's review. Being able to search on the destination service identifier affords the administrators the opportunity to see what traffic is destined for a service (e.g., TCP port) or set of services regardless of where the traffic originated. It is important to note that the intent of sorting in this requirement is to allow the administrators the capability to organize or group the records associated with a given criteria. For example, if the administrators wanted to see what network traffic was destined for the set of TCP ports 1-1024, they would be able to have the audit data presented in such a way that all the traffic for TCP port 1 was grouped together, all the traffic for port 2 was grouped together and so on. The criteria includes the rule identity that determines whether a packet was allowed or denied to flow. This provides the administrators to determine what network traffic a given rule is governing. FMT_MOF.1(3) restricts the ability to control the behavior of the audit and alarm mechanism to the administrators. The Security Administrator is the only user that controls the behavior of the events that generate alarms. FMT_MOF.1(4) provides the administrators "read only" access to the audit records and prohibits access to all other users. Additionally the administrators are provided the capability to

Objective	Requirements Addressing the Objective	Rationale
		"search and sort" audit on defined criteria. This capability expedites problem resolution analysis. FMT_MOF.1(5) ensures that only an administrators can "enable or disable" the security alarms. This requirement works with FMT_MOF.1(4) to provide detailed granularity to the administrator when determining which actions constitute a security violation
O.AUDITS The TOE must record audit records for data accesses and use of the Sensor functions.	FAU_GEN.1 FAU_SEL.1 FAU_STG.4 FPT_RVM.1 FPT_SEP.2 FPT_STM.1	Security-relevant events must be defined and auditable for the TOE [FAU_GEN.1]. The TOE must provide the capability to select which security-relevant events to audit [FAU.SEL.1]. The TOE must prevent the loss of collected data in the event its audit trail is full [FAU_STG.4]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.2]. Time stamps associated with an audit record must be reliable [FPT_STM.1].
O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.	ACM_CAP.4 ACM_SCP.2 ALC_DVS.1 ALC_FLR.3 ALC_LCD.1 ACM_AUT.1	ACM_CAP.4 contributes to this objective by requiring the developer have a configuration management plan that describes how changes to the TOE and its evaluation deliverables are managed. The developer is also required to employ a configuration management system that operates in accordance with the CM plan and provides the capability to control who on the development staff can make changes to the TOE and its developed evidence. This requirement also ensures that authorized changes to the TOE have been analyzed and the developer's acceptance plan describes how this analysis is performed and how decisions to incorporate the changes to the TOE are made. ACM_SCP.2 is necessary to define what items must be under the control of the CM system. This requirement ensures that the TOE implementation representation, design documentation, test documentation (including the executable test suite), user and administrator guidance, CM documentation and security flaws are tracked by the CM system. ALC_DVS.1 requires the developer describe the security measures they employ to ensure the integrity and confidentiality of the TOE are maintained. The physical, procedural, and personnel security measures the developer uses provides an added level of control over who and how changes are made to the TOE and its associated evidence. ALC_FLR.3 plays a role in satisfying the "analyzed" portion of this objective by requiring the developer to have procedures that address flaws that have been discovered in the product, either through developer actions (e.g., developer testing) or those discovered by others. The flaw remediation process used by the developer corrects any discovered flaws and performs an analysis to ensure new flaws are not created while fixing the discovered flaws. ALC_LCD.1 requires the developer to document the life-cycle model used in the development and maintenance of the TOE. This life-cycle model describes the procedural aspects regarding the development of the TOE, such as design methods, code or

Objective	Requirements Addressing the Objective	Rationale
		documentation reviews, how changes to the TOE are reviewed and accepted or rejected. ACM_AUT.1 complements ACM_CAP.4, by requiring that the CM system use an automated means to control changes made to the TOE. If automated tools are used by the developer to analyze, or track changes made to the TOE, those automated tools must be described. This aids in understanding how the CM system enforces the control over changes made to the TOE.
O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.	FPT_TST.1(1), FPT_TST.1(2) FPT_AMT.1	O_CORRECT_TSF_OPERATION requires two security functional requirements in the FPT class, FPT_TST and FPT_AMT. These functional requirements provide the end user with the capability to ensure the TOE's security mechanisms continue to operate correctly in the field. FPT_TST.1(1) ensures that end user tests exist to demonstrate the correct operation of the security mechanisms required by the TOE that are provided by the hardware. Hardware failures could render a TOE's software ineffective in enforcing its security policies and this requirement provides the end user the ability to discover any failures in the hardware security mechanisms. FPT_TST.1(2) is necessary to ensure the correctness of the TSF cryptographic functions and the TSF data which support those functions. If TSF software is corrupted it is possible that the TSF would no longer be able to enforce the security policies. This also holds true for TSF data, if TSF data is corrupt the TOE may not correctly enforce its security policies. FPT_AMT.1 is necessary to support FPT_TST.1(1) and FPT_TST.1(2) by ensuring the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

O.CRYPTOGRAPHIC_FUNCTIONS
The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.
	FCS_CKM.1(1)
	FCS_CKM.1(2)
	FCS_CKM.1(3)
	FCS_CKM.1(4)
	FCS_CKM.1(5)
	FCS_CKM.2(1)
	FCS_CKM.2(2)
	FCS_CKM.4
	FCS_COP.1(1)
	FCS_COP.1(2)
	FCS_COP.1(3)
	FCS_COP.1(4)
		The FCS requirements used in this ST satisfy this objective by levying requirements that ensure the cryptographic standards include the NIST FIPS publications (where possible) and NIST approved ANSI standards. The intent is to have the satisfaction of the cryptographic standards be validated through a NIST FIPS 140 validation.
		In contrast to O.CRYPTOGRAPHY_VALIDATED, this objective is to provide cryptographic functionality that is used by the TOE. The core functionality to be supported is encryption/decryption using a symmetric algorithm, and digital signature generation and verification using asymmetric algorithms. Since these operations involve cryptographic keys, how the keys are generated and/or otherwise obtained have to also be specified.
		FCS_CKM.1(1) is a requirement for the generation of symmetric and asymmetric keys. FCS_CKM.1(2) specifies requirements for AES keys. The requirement FCS_CKM.1(3) describes key entry methods. FCS_CKM.1(4) specifies the key validation techniques. Internet key exchange is addressed in FCS_CKM.1(15).
		FCS_CKM.2(1) specifies how keys are to be stored and handled including the destruction of non-persistent keys that have not been used for an administrator defined period of time. FCS_CKM.2(2) provides for key distribution.
		FCS_CKM.4 provides the functionality for ensuring key and key material is zeroized. This applies not only to key that resides in the TOE, but also to intermediate areas (physical memory, page files,

Objective	Requirements Addressing the Objective	Rationale
		memory dumps, etc.) where key may appear. The requirements FCS_COP.1(1) through FCS_COP.1(4) specify the cryptographic algorithms.
O.CRYPTOGRAPHY_VALIDATED The TOE shall use NIST FIPS 140-2 validated cryptomodules for cryptographic services implementing FIPS-approved security functions and random number generation services used by cryptographic functions.	FCS_BCM_EXP.1	This objective deals with the issue of using FIPS 140-2-approved cryptomodules in the TOE. A cryptomodule, as used in the components, is a module that is FIPS 140-2 validated (in accordance with FCS_BCM_EXP.1); the cryptographic functionality implemented in that module are FIPS-approved security functions that have been validated; and the cryptographic functionality is available in a FIPS-approved mode of the cryptomodule. This objective is distinguished from O.CRYPTOGRAPHIC_FUNCTIONS in that this deals only with a requirement to use FIPS 140-2-validated cryptomodules where the TOE requires such functionality; it does not dictate the specific functionality that is to be used. FCS_BCM_EXP.1 is an explicit requirement that specifies not only that cryptographic functions that are FIPS-approved must be validated by FIPS, but also what NIST FIPS rating level the cryptographic module must satisfy. The level specifies the degree of testing of the module. The higher the level, the more extensive the module is tested.
O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1	FTA_TAB.1 meets this objective by requiring the TOE display a Security Administrator defined banner before a user can establish an authenticated session. This banner is under complete control of the Security Administrator in which they specify any warnings regarding unauthorized use of the TOE and remove any product or version information if they desire.
O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data.	FAU_SAR.1 FAU_SAR.3 FAU_SEL.1 FPT_RVM.1 FPT_SEP.2 IDS_RDR_EXP.1	The TOE must provide the ability to review and manage the audit trail of a Sensor [FAU_SAR.1, FAU_SAR.3, FAU_SEL.1]. The Sensor must provide the ability for authorized administrators to view the Sensor data collected from an IT System [IDS_RDR_EXP.1]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.2].
O.EXPORT When the TOE makes its Sensor data available to other IDS components, the TOE will ensure the confidentiality of the Sensor data.	FPT_ITA.1 FPT_ITC.1 FPT_ITI.1	The TOE must make the collected data available to other IT products [FPT_ITA.1]. The TOE must protect the collected data from modification and ensure its integrity when the data is transmitted to another IT product [FPT_ITC.1, FPT_ITI.1].
O.IDACTS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	IDS_COL_EXP.1	The Sensor is required to collect events indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets of an IT System. These events must be defined in the ST [IDS_COL_EXP.1].

Objective	Requirements Addressing the Objective	Rationale
O.IDAUTH The TOE must be able to identify and authenticate authorized users prior to allowing access to TOE functions and data.	FAU_SAR.2 FAU_STG.2 FIA_UAU.1(1), FIA_UAU.1(2) FIA_ATD.1(1), FIA_ATD.1(2), FIA_ATD.1(3) FIA_UID.2 FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(7), FMT_MOF.1(8), FMT_MOF.1(9), FMT_MOF.1(10), FMT_MOF.1(11), FMT_MOF.1(12), FMT_MOF.1(13) FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7), FMT_MTD.1(8), FMT_MTD.1(9) FMT_SMR.2 FPT_RVM.1 FPT_SEP.2 IDS_RDR_EXP.1 IDS_STG_EXP.1	The TOE is required to restrict the review of audit data to those granted with explicit read-access [FAU_SAR.2]. The Sensor is required to restrict the review of collected Sensor data to those granted with explicit read access [IDS_RDR.1]. The TOE is required to protect the stored audit records from unauthorized deletion [FAU_STG.2]. The Sensor is required to protect the Sensor data collected from an IT System from any modification and unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG.1]. Security attributes of subjects use to enforce the authentication policy of the TOE must be defined [FIA_ATD.1]. Users authorized to access the TOE are defined using an identification and authentication process [FIA_UID.2, FIA_UAU.1]. The TOE is required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. Only authorized administrators of the Sensor may query and add Sensor and audit data, and authorized administrators of the TOE may query and modify all other TOE data [FMT_MTD.1]. The TOE must be able to recognize the different administrative and user roles that exist for the TOE [FMT_SMR.2]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.2].
O.INTEGR The TOE must ensure the integrity of all audit and Sensor data.	FAU_STG.2 FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7), FMT_MTD.1(8), FMT_MTD.1(9) FPT_ITC.1	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The Sensor is required to protect the Sensor data collected from an IT System from any modification and unauthorized deletion [IDS_STG_EXP.1]. Only authorized administrators of the Sensor may query or add audit and Sensor data [FMT_MTD.1]. The Sensor must protect the collected data from modification and ensure its integrity when the data is transmitted to another IT product [FPT_ITC.1, FPT_ITI.1]. The TOE must ensure that all functions to protect the data are not bypassed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.2].

Objective	Requirements Addressing the Objective	Rationale
	FPT_ITI.1 FPT_RVM.1 FPT_SEP.2 IDS_STG_EXP.1	

O.INTEGRITY
The TOE must be able to protect the integrity of data transmitted to a peer TOE via encryption and provide IPSec authentication for such data. Upon receipt of data from a peer TOE, the TOE must be able to decrypt the data and verify that the received data accurately represents the data that was originally transmitted.
	FDP_IFC.1(3)
	FDP_IFF.1(3)
		FDP_IFC.1(3) and FDP_IFF.1(3)) satisfies this objective by defining the VPN Information Flow Security Functional Policy that ensures that all IPSec encrypted data received from a peer TOE is properly decrypted and authentication verified.

O.INTRUSION The TOE will detect and prevent intrusion attacks which are contained within an information flow which arrives at any of the TOE network interfaces.	FIP_ACT_EXP.1	The FIP_ACT_EXP.1 requirement dictates that the TOE detect and prevent intrusion attacks which are contained within any information flow which arrives at any of the TOE network interface. This requirement satisfies the O.INTRUSION objective.
O.MAINT_MODE The TOE shall provide a mode from which recovery or initial startup procedures can be performed.	FPT_RCV.1	This objective is met by using the FPT_RCV.1 requirement, which ensures that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. Upon the failure of the TSF self-tests the TOE will enter a mode where it can no longer be assured of enforcing its security policies. Therefore, the TOE enters a state that disallows traffic flow and requires an administrator to follow documented procedures that instruct them on how to return the TOE to a secure state. These procedures may include running diagnostics of the hardware, or utilities that may correct any integrity problems found with the TSF data or code. Solely specifying that the administrator reload and install the TOE software from scratch, while might be required in some cases, does not meet the intent of this requirement.
O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	FMT_MSA.1 FMT_MSA.2 FMT_MSA.3(1) , FMT_MSA.3(2) FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(7), FMT_MOF.1(8), FMT_MOF.1(9), FMT_MOF.1(10), FMT_MOF.1(11), FMT_MOF.1(12), FMT_MOF.1(13)	The FMT requirements are used to satisfy this management objective, as well as other objectives that specify the control of functionality. The requirement's rationale for this objective focuses on the administrator's capability to perform management functions in order to control the behavior of security functions. FMT_MSA.1 provides the Security Administrator the capability to manipulate the security attributes to facilitate the construction of the ruleset. An example of this would be to group a set of service identifiers that are to have the same rule applied, rather than having to specify a separate rule for each service identifier. FMT_MSA.2 requires that only secure values for security attributes are accepted. FMT_MSA.3(1) requires that by default, the TOE does not allow an information flow, rather than allowing information flows until a rule in the ruleset disallows it. FMT_MOF.1(2) and FMT_MSA.3(2) are related to the services

Objective	Requirements Addressing the Objective	Rationale
	FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7), FMT_MTD.1(8), FMT_MTD.1(9) FAU_SAR.1 FAU_SAR.2 FAU_SAR.3 FAU_SEL.1 FAU_STG.2 FAU_STG.3 FAU_STG.4 FAU_ARP_ACK_EXP.1	provided by FAU_UAU.1(1) and provide the Security Administrator control as to the availability of these services. FMT_MOF.1(2) provides the ability to enable or disable the TOE services to the Security Administrator. FMT_MSA.3(2) requires that these services by default are disabled. Since the Security Administrator must explicitly enable these services it ensures the Security Administrator is aware that they are running. This requirement does afford the Security Administrator the capability to override this restrictive default and allow the services to be started whenever the TOE reboots or is restarted. FMT_MOF.1(1) is used to ensure the administrators have the ability to invoke the TOE self-tests at any time. The ability to invoke the self-tests is provided to all administrators. The Security Administrator is able to modify the behavior of the tests (e.g., select when they run, select a subset of the tests). FMT_MOF.1(3) specifies the ability of the administrators to control the security functions associated with audit and alarm generation. The ability to control these functions has been assigned to the appropriate administrative roles. FMT_MOF.1(7) This requirement limits the ability to manipulate the values that are used in the FRU_RSA.1(2) requirements to the Security Administrator. The Security Administrator is provided the capability to assign the network identifier(s) they wish to place resource restrictions on and allows them to also specify over what period of time those quota limitations are in place. FMT_MOF.1(4) provides the administrators "read only" access to the audit records and prohibits access to all other users. Additionally the administrators are provided the capability to "search and sort" audit on defined criteria. This capability expedites problem resolution analysis. FMT_MOF.1(5) ensures that only an administrators can "enable or disable" the security alarms. This requirement works with FMT_MOF.1(4) to provide detailed granularity to the administrator when determining which actions constitute a security violation FMT_MOF.1(6) limits the ability to enable or disable unauthenticated TOE services for both IP based networks and non-IP based networks to the Security Administrator. These TOE services would be available to appropriate network users at the discretion of the Security Administrator. FMT_MOF.1(7) provides the Security Administration configuration control of the allocation of connection-oriented TOE resources. This requirement provides the Security Administrator with a capability to thwart possible external "resource allocation" attacks on the TOE. FMT_MOF.1(13) This requirement limits the ability to manipulate the values that are used in the IDS_COL requirements to the Security Administrator. The Security Administrator is provided the capability to configure the IDS data collection. FMT_MTD.1(8) provides the Cryptographic Administrator, and only the Cryptographic Administrator, the ability to modify the cryptographic security data. This allows the Cryptographic

Objective	Requirements Addressing the Objective	Rationale
		Administrator to change the critical data that affects the TOE's ability to perform its cryptographic functions properly. FMT_MTD.1(3) provides the capability of setting the date and time that is used to generate time stamps to the Security Administrator or an authorized IT entity. It is important to allow this functionality, due to clock drift and other circumstances, but the capability must be restricted. FMT_MTD.1(4) provides the Security Administrator the capability to manage the TOE's ruleset. This capability is restricted to only the Security Administrator and allows them to create, view, modify and delete the rules that comprise the ruleset. FAU_SAR.1 ensures that the Audit Administrator has the capability to review the audit records and that they are presented in a manner that is suitable for review (e.g., the Audit Administrator can construct a sequence of events provided the necessary events were audited). FAU_SAR.2 restricts the ability to read the audit records to the administrators. This capability exists for the Security and Crypto administrators to help facilitate any trouble shooting that they may have to perform. FAU_SAR.3 provides the administrators with the ability to selectively review the contents of the audit trail based on established criteria. This capability allows the administrators to focus their audit review to what is pertinent at that time. FAU_STG.2 specifies that only the Audit Administrator can delete the audit trail. This prevents the accidental or intentional deletion of the audit trail by administrators acting in another role. FAU_STG.3 provides the Security Administrator the capability to establish a threshold of audit trail capacity, that when reached an alarm will be generated. If the audit trail becomes full FAU_STG.4 provides the Security Administrator the option of having the TOE prevent auditable events from occurring, or having the TOE overwrite the oldest audit records. While the option of overwriting old audit records does not technically prevent audit data loss, it is provided to the Security Administrator as an option to prevent a possible denial-of-service. FAU_ARP_ACK_EXP.1 contributes to this objective in that it requires the administrators to acknowledge an alarm before it is no longer displayed. Without this requirement an alarm display message may be overwritten or lost without an administrator being aware of the alarm condition. FAU_SEL.1 provides the Security Administrator the ability to define what events will be included or excluded from the list of audited events. This allows a site to audit only those events that are of interest to them and reduces the amount of unwanted audit data that is collected.
O.MEDIATE	FDP_IFF.1(1)	The FDP_IFF and FDP_IFC requirements were chosen to define the policies, the subjects, objects, and operations for how and when

Objective	Requirements Addressing the Objective	Rationale

The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	FDP_IFF.1(2)
	FDP_IFF.1(3)
	FDP_IFC.1(1)
	FDP_IFC.1(2)
	FDP_IFC.1(3)
	FMT_REV.1
	FPT_RVM.1
		mediation takes place.
		FDP_IFC.1(1), FDP_IFC.1(2), and FDP_IFC.1(3) define the subjects, information (e.g., objects) and the operations that are performed with respect to the three information flow policies.
		FDP_IFC.1(1), the subjects are defined to be a source subject, which is the TOE's network interface on which a packet is received, and a destination subject, which is the TOE's network interface on which the packet is destined. The information flow control requirements are not well suited for a firewall. This subject determination was made since the TOE network interfaces are something the TOE has control over (e.g., the administrator has the ability to assign network identifiers to these interfaces, which is a critical component in the mediation decision) and rules could be identified in FDP_IFF.1(1) that make sense with respect to mediation of information. The alternative was to classify the sender and receiver of the data packets as subjects, but the sender and receiver are not under the control of the TOE and would not make sense to perform mediation under those circumstances. The objects in this policy are defined to be the network packets, since that is the entity that the operations are performed on. Those operations are to pass the information if the mediation allows the flow, otherwise the packet is dropped. Due to the inclusion of unauthenticated proxies, another specified operation is to establish a connection for the unauthenticated proxy user. The TOE establishes a connection between itself and the proxy user, and between itself and the target machine. This ensures that all traffic between the proxy user and the target machine is mediated by the TOE (e.g., the TOE is not simply providing a "pipe" between the user and target machine). FDP_IFF.1(1) is used to specify the policy of unauthenticated traffic flowing through the TOE. This requirement ensures that the network traffic is mediated (i.e., the ruleset is used) even though the subjects have not been authenticated. The policy specified by this requirement also includes an unauthenticated proxy (SMTP) that affords the administrator the ability to specify commands and Multipurpose Internet Mail Extensions (MIME) types that are filtered by the proxy. There is an open assignment that can be filled in by the ST author to identify proxies they may want to include in an ST that do not require authentication. If the ST author includes additional proxies, they should include the attributes the Security Administrator could specify that the TOE would filter. This requirement also mandates the TOE perform stateful inspection of the packets to determine if they should be allowed to flow through the TOE. The stateful inspection attributes are not intended to specifiable by the Security Administrator, rather these attributes are to be "managed" and mediated internally by the TOE.
		FDP_IFC.1(2) defines subjects for the unauthenticated access to any services the TOE provides. This is different from the other policies in that the TOE mediates access to itself, rather than determining if information should be allowed to flow through the TOE. The destination subject is defined to be the TOE, and the source subject is the TOE interface on which a network packet is received. The information remains the same, a network packet, and the operations are limited to accept or reject the packet. FDP_IFF.1(2) provides the rules that apply to the unauthenticated use of any services provided by the TOE. ICMP is the only service that is required to be provided by the TOE, and the security attributes associated with this protocol allow the Security Administrator to specify what degree the ICMP traffic is mediated (i.e., the ICMP message type and code). The ST author could

Objective	Requirements Addressing the Objective	Rationale
		specify other services they wish their TOE implementation to provide, and if they do so, they should also specify the security attributes associated with the additional services. FDP_IFC.1(1), the subjects are the TOE's network interfaces. The objects are defined as the network IP packets on which the TOE performs VPN operations. As packets enter the TOE, the network interface where they are received is the source subject. As packets are sent out of the TOE the network interface that they are sent out of is the destination subject. Subjects must be defined as entities that the TOE has control over. The TOE has control over its own network interfaces such that it can make information flow decisions to allow/disallow network packets to flow from in incoming interface to an outgoing interface, and can apply VPN operations to packets that are allowed to flow. To define subjects as the senders and receivers of network packets would not allow specification of an information flow policy that the TOE could enforce, since the sender and receiver of network packets are not under the control of the TOE. The operations defined are those of the VPN policy. The VPN policy either passes information unmodified, sends encrypted and authenticated packets to a peer TOE, or decrypts and verifies authentication of packets received from a peer TOE. FDP_IFF.1(3) specifies the attributes on which VPN information flow decisions are made. Each TOE interface has a set of source subject identifiers that is the list of senders of information packets that are allowed to send packets to this TOE interface. Each TOE interface also has a list of destination subject identifiers that specifies the receivers that network packets can be sent to on that TOE interface. As packets are received on a particular network interface, the TOE determines if they are allowed to enter on that interface. Then based on rules defined by the Security Administrator, the TOE applies VPN operations to the packet. Before the packet is sent out of a particular network interface, the TOE determines if the destination (i.e., receiver) of the packet is in the list of destinations that may be reached over that interface. FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user's ability to send network traffic to or through the TOE. If the Security Administrator revokes a user's access (e.g., via a rule in the ruleset, revoking an administrative role from a user, revoking a user's ability to use a proxy) the TOE will immediately enforce the new Security Administrator defined "policy". FPT_RVM.1 ensures that packets that flow through the TOE, or those that are destined for the TOE are mediated with respect to the identified policies. Each TSF interface that operates on subjects or objects that are identified in the explicit policies, or operates on TSF data or security attributes, must ensure that the operation is checked against the explicit and implicit security policies defined in this ST. If any TSF interface allows unchecked access to any of these resources, then the TOE cannot be relied upon to enforce the security policies.
O.OFLOWS The TOE must appropriately handle potential audit and Sensor data storage overflows.	FAU_STG.2 FAU_STG.4 IDS_STG_EXP.1	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The TOE must prevent the loss of audit data in the event its audit trail is full [FAU_STG.4]. The Sensor is required to protect the Sensor data collected from an IT System from any modification and

Objective	Requirements Addressing the Objective	Rationale
	IDS_STG_EXP.2	unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG_EXP.1]. The Sensor must prevent the loss of audit data in the event its audit trail is full [IDS_STG_EXP.2].
O.PEER_AUTHENTICATION The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.	FCS_CKM.1(5)	The O.PEER_AUTHENTICATION objective is satisfied by the requirement FCS_CKM.1(5), which specifies that the TOE must implement the Internet Key Exchange protocol defined in RFC 2409. By implementing this protocol, the TOE will establish a secure, authenticated channel with each peer TOE for purposes of establishing a security association, which includes the establishment of a cryptographic key, algorithm and mode to be used for all communication. It is possible to establish multiple security associations between two peer TOEs, each with its own cryptographic key. Authentication may be via a digital signature or pre-shared key.
O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	FAU_STG.2 FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(7), FMT_MOF.1(8), FMT_MOF.1(9), FMT_MOF.1(10), FMT_MOF.1(11), FMT_MOF.1(12), FMT_MOF.1(13) FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7), FMT_MTD.1(8), FMT_MTD.1(9) FPT_RVM.1 FPT_SEP.2 IDS_STG_EXP.1	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The Sensor is required to protect the Sensor data collected from an IT System from any modification and unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG_EXP.1]. The TOE is required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. Only authorized administrators of the Sensor may query and add Sensor and audit data, and authorized administrators of the TOE may query and modify all other TOE data [FMT_MTD.1]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.2].
O.REPLAY_DETECTION The TOE will provide a means to detect and reject the replay of TSF data and security attributes.	FPT_RPL.1	The O.REPLAY_DETECTION objective is satisfied by the requirement FPT_RPL.1, which requires the TOE to not only detect, but to also reject the attempted replay of TSF data, and security attributes. This requirement also requires the TOE to audit the detection of replay, which affords the administrators the opportunity to be aware of users attempting to replay critical data and affect the TOE's ability to enforce security policies as desired by the administrators.
O.RESIDUAL_INFORMATION	FDP_RIP.2	FDP_RIP.2 is used to ensure the contents of resources are not available to subjects other than those explicitly granted access to

Objective	Requirements Addressing the Objective	Rationale
The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	FCS_CKM.4	the data. For this TOE it is critical that the memory used to build network packets is either cleared or that some buffer management scheme be employed to prevent the contents of a packet being disclosed in a subsequent packet (e.g., if padding is used in the construction of a packet, it must not contain another user's data or TSF data). FCS_CKM.4 applies to the destruction of cryptographic keys used by the TSF. This requirement specifies how and when cryptographic keys must be destroyed. The proper destruction of these keys is critical in ensuring the content of these keys cannot possibly be disclosed when a resource is reallocated to a user.

O.RESOURCE_SHARING
The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; TCP connections used by proxies).
	FRU_RSA.1(1)
	FRU_RSA.1(2)
	FMT_MTD.2(1)
	FMT_MTD.2(2)
	FMT_MOF.1(7)
		While an availability security policy does not explicitly exist, FRU_RSA.1 was used to mitigate potential resource exhaustion attempts. FRU_RSA.1(1) was used to reduce the impact of an attempt being made to exhaust the transport-layer representation (e.g., attempt to make the TSF unable to respond to connection-oriented requests, such as SYN attacks). This requirement allows the administrator to specify the time period in which when maximum quota (which is defined by the ST) is met or surpassed, an ST defined action is to take place, which is specified in FMT_MTD.2(1). These two requirements together help limit the resources that can be utilized by the general population of users as a whole. An issue with treating all the users the same is that legitimate users may not be able to establish connections due to the connection table entries being exhausted. Therefore FRU_RSA.1(2) is also included.
		FRU_RSA.1(2) is more specific in that attempts to exhaust the connection-oriented resources by a single network address, or a set of network addresses can be controlled. This affords the administrator a finer granularity of control than FRU_RSA.1(1). FRU_RSA.1(2) has the advantage of providing the Security Administrator with the ability to define the maximum number of resources a particular address or set of addresses can use over a specified time period. This requirement works in conjunction with FMT_MTD.2(2) which restricts the ability to set the quotas to the security administrator and allows for the ST author to assign what actions will take place once the quotas are met or surpassed. This iteration of FPT_RSA.1 makes it less likely that a legitimate user of the TOE will be denied access due to resource exhaustion attempts.
		FMT_MOF.1(7) restricts the ability to assign the single network address or set of network addresses used in FRU_RSA.1(2) to the Security Administrator. This is in keeping with the TOE's notion of the Security Administrator is responsible for configuring the TOE's policy enforcement mechanisms.

O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	ADO_DEL.2 AGD_ADM.1 AVA_MSU.2 ADO_IGS.1 AGD_USR.1	ADO_DEL.2 ensures that the administrator is provided documentation that instructs them how to ensure the delivery of the TOE, in whole or in parts, has not been tampered with or corrupted during delivery. This requirement ensures the administrator has the ability to begin their TOE installation with a clean (e.g., malicious code has not been inserted once it has left the developer's control) version of the TOE, which is necessary for secure management of the TOE. The ADO_IGS.1 requirement ensures the administrator has the information necessary to install the TOE in the evaluated

Objective	Requirements Addressing the Objective	Rationale
		configuration. Often times a vendor's product contains software that is not part of the TOE and has not been evaluated. The Installation, Generation and Startup (IGS) documentation ensures that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration. The AGD_ADM.1 requirement mandates the developer provide the administrator with guidance on how to operate the TOE in a secure manner. This includes describing the interfaces the administrator uses in managing the TOE, security parameters that are configurable by the administrator, how to configure the TOE's ruleset and the implications of any dependencies of individual rules. The documentation also provides a description of how to setup and review the auditing features of the TOE. The AGD_USR.1 is intended for non-administrative users, but could be used to provide guidance on security that is common to both administrators and non-administrators (e.g., password management guidelines). The use of the authentication mechanism would not have to be repeated in the administrator's guide. AVA_MSU.2 ensures that the guidance documentation is complete and can be followed unambiguously to ensure the TOE is not mis-configured in an unsecure state due to confusing guidance.
O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	FIA_UID.2 FIA_ATD.1(1), FIA_ATD.1(2), FIA_ATD.1(3) FIA_AFL.1 FIA_UAU.1(1), FIA_UAU.1(2) FIA_UAU.2 FIA_UAU.5 FIA_UID.2 FTA_SSL.1 FTA_SSL.2 FTA_SSL.3 FTA_TSE.1 AVA_SOF.1	FIA_UID.2 plays a small role in satisfying this objective by ensuring that every user is identified before the TOE performs any mediated functions. In some cases, the identification cannot be authenticated (e.g., a user attempting to send a data packet through the TOE that does not require authentication; in which case the identity is presumed to be authentic). In other cases (e.g., administrators, and authorized IT entities), the identity of the user is authenticated. It is impractical to require authentication of all users that attempt to send data through the TOE, therefore, the requirements specified in the TOE require authentication where it is deemed necessary. This does impose some risk that a data packet was sent from an identity other than specified in the data packet. FIA_ATD.1 defines the attributes of users, including a userid that is used to by the TOE to determine a user's identity and enforce what type of access the user has to the TOE (e.g., the TOE associates a userid with any role(s) they may assume). This requirement allows a human user to have more than one user identity assigned, so that a single human user could assume all the roles necessary to manage the TOE. In order to ensure a separation of roles, this ST requires a single role to be associated with a user id. This is inconvenient in that the administrator would be required to log in with a different user id each time they wish to assume a different role, but this helps mitigate the risk that could occur if an administrator were to execute malicious code. FIA_UAU.1 contributes to this objective by limiting the services that are provided by the TOE to unauthenticated users. Management requirements and the unauthenticated information flow policy requirement provide additional control on these services. FIA_UAU.2 requires that administrators and authorized IT entities authenticate themselves to the TOE before performing

Objective	Requirements Addressing the Objective	Rationale
		administrative duties (including those performed by authorized IT entities). In order to control logical access to the TOE an authentication mechanism is required. The explicit requirement FIA_UAU.5 mandates that the TOE provide a local authentication mechanism. This requirement also affords the ST author the opportunity to add additional authentication mechanisms (e.g., single-use, certificates) if they desire. Local authentication is required to ensure someone that has physical access to the TOE and has not been granted logical access (e.g., a janitor) cannot gain unauthorized logical access to the TOE. The AVA_SOF.1 requirement is applied to the local authentication mechanism. For this TOE, the strength of function specified is basic. This requirement ensures the developer has performed an analysis of the authentication mechanism to ensure the probability of guessing a user's authentication data would require a low-attack potential, as defined in Annex B of the Common Evaluation Methodology (CEM). FTA_TSE.1.1 contributes to this objective by limiting a user's ability to logically access the TOE. This requirement provides the Security Administrator the ability to control when (e.g., time and day(s) of the week) and where (e.g., from a specific network address) remote administrators as well as authorized IT entities can access the TOE. FIA_AFL.1 provides a detection mechanism for unsuccessful authentication attempts by remote administrators and authorized IT entities. The requirement enables a Security Administrator settable threshold that prevents unauthorized users from gaining access to authorized user's account by guessing authentication data by locking the targeted account until the Security Administrator takes some action (e.g., re-enables the account) or for some Security Administrator defined time period. Thus, limiting an unauthorized user's ability to gain unauthorized access to the TOE. The FTA_SSL family partially satisfies the O.ROBUST_TOE_ACCESS objective by ensuring that user's sessions are afforded some level of protection. FTA_SSL.1 provides the Security Administrator the capability to specify a time interval of inactivity in which an unattended local administrative session would be locked and will require the administrator responsible for that session to re-authenticate before the session can be used to access TOE resources. FTA_SSL.2 provides administrators the ability to lock their local administrative session. This component allows administrators to protect their session immediately, rather than waiting for the time-out period and minimizes their session's risk of exposure. FTA_SSL.3 takes into account remote sessions. After a Security Administrator defined time interval of inactivity remote sessions will be terminated, this refers to remote administrative sessions. This component is especially necessary, since remote sessions are not typically afforded the same physical protections that local sessions are provided.
O.SECURE_UPDATES The TOE shall provide a secure mechanism for the	FTP_ITC.1(1)	The FTP_ITC.1(1) and FTP_ITC.1(2) requirements dictate that the TOE is capable of establishing a trusted channel between itself and a FortiGuard Distribution Server for the secure transmission of IPS

Objective	Requirements Addressing the Objective	Rationale
receipt of virus and intrusion signature updates.	FTP_ITC.1(2)	(attack) signatures and virus definitions. This trusted channel capability meets the O.SECURE_UPDATES objective.
O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	FPT_FLS.1 FPT_SEP.2 FPT_RVM.1 FTP_ITC.1(1), FTP_ITC.1(2), FTP_TRP.1(1), FTP_TRP.1(2) FTP_ITC.1(3) (ENV), FTP_ITC.1(4) (ENV) FTP_TRP.1(3) (ENV), FTP_TRP.1(4) (ENV)	FPT_FLS.1 ensures that the TSF preserves a secure state when a unit in a FortiGate cluster fails. FPT_SEP was chosen to ensure the TSF provides a domain that protects itself from untrusted users. If the TSF cannot protect itself it cannot be relied upon to enforce its security policies. FPT_SEP.1 could have been used to address the previous notion, however, FPT_SEP.2 was used to require that the *cryptographic module* be provided its own address space. This is necessary to reduce the impact of programming errors in the remaining portions of the TSF on the cryptographic module. The inclusion of FPT_RVM.1 ensures that the TSF makes policy decisions on all interfaces that perform operations on subjects and objects that are scoped by the policies. Without this nonbypassability requirement, the TSF could not be relied upon to completely enforce the security policies, since an interface(s) may otherwise exist that would provide a user with access to TOE resources (including TSF data and executable code) regardless of the defined policies. This includes controlling the accessibility to interfaces, as well as what access control is provided within the interfaces. FTP_ITC.1(1), FTP_ITC.1(2) and FTP_TRP.1(1), FTP_TRP.1(2) are necessary for communication between the TOE and other trusted IT entities (e.g., authentication server, authorized IT entities) and the TOE and remote administrators. In order to protect TSF data and security attributes there is need for a trusted channel/trusted path. The trusted channel ensures that the authentication data that is supplied to the TOE is not compromised. It may be the case that the TOE relies upon an authorized IT entity to supply/manage TSF data (e.g., time stamp). If this is the case, the trusted channel ensures the TSF data is not compromised. The aspect of the trusted path that applies to this objective is FTP_TRP.1.3, which requires that the entire remote administrative session be protected. The protection of the communication path when TSF data is being transmitted is critical to the TSF maintaining a domain of execution that cannot be tampered or interfered with, thus resulting in a possible unauthorized disclosure or security policy failure.
O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.	ADV_FSP.2 ADV_HLD.2 ADV_LLD.1 ADV_RCR.1 ADV_SPM.1	There are two different perspectives for this objective. One is from the developer's point of view and the other is from the evaluator's. The ADV class of requirements is levied to aid in the understanding of the design for both parties, which ultimately helps to ensure the design is sound. ADV_SPM.1 requires the developer to provide an informal model of the security policies of the TOE. Modeling these policies helps understand and reduce the unintended side-effects that occur during the TOE's operation that might adversely affect the TOE's ability to enforce its security policies. ADV_FSP.2 requires that the interfaces to the TSF be completely specified. In this TOE, a complete specification of the network interface (including the network interface card) is critical in understanding what functionality is presented to untrusted users and how that functionality fits into the enforcement of security

Objective	Requirements Addressing the Objective	Rationale
		policies. Some network protocols have inherent flaws and users have the ability to provide the TOE with network packets crafted to take advantage of these flaws. The routines/functions that process the fields in the network protocols allowed (e.g., TCP, UDP, ICMP, any application level) must fully specified: the acceptable parameters, the errors that can be generated, and what, if any, exceptions exist in the processing. The functional specification of the hardware interface (e.g., network interface card) is also extremely critical. Any processing that is externally visible performed by Network Interface Card (NIC) must be specified in the functional specification. Having a complete understanding of what is available at the TSF interface allows one to analyze this functionality in the context of design flaws. ADV_HLD.2 requires that a high-level design of the TOE be provided. This level of design describes the architecture of the TOE in terms of subsystems. It identifies which subsystems are responsible for making and enforcing security relevant (e.g., anything relating to a Security Functional Requirement (SFR)) decisions and provides a description, at a high level, of how those decisions are made and enforced. Having this level of description helps provide a general understanding of how the TOE works, without getting buried in details, and may allow the reader to discover flaws in the design. The low-level design, as required by ADV_LLD.1, provides the reader with the details of the TOE's design and describes at a module level how the design of the TOE addresses the SFRs. This level of description provides the detail of how modules interact within the TOE and if a flaw exists in the TOE's design, it is more likely to be found here rather than the high-level design. This requirement also mandates that the interfaces presented by modules be specified. Having knowledge of the parameters a module accepts, the errors that can be returned and a description of how the module works to support the security policies allows the design to be understood at its lowest level. The ADV_RCR.1 is used to ensure that the levels of decomposition of the TOE's design are consistent with one another. This is important, since design decisions that are analyzed and made at one level (e.g., functional specification) that are not correctly designed at a lower level may lead to a design flaw. This requirement helps in the design analysis to ensure design decisions are realized at all levels of the design.
O.SOUND_IMPLEMENTATION The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.	ADV_IMP.1 ADV_LLD.1 ADV_RCR.1 ALC_TAT.1	While ADV_LLD.1 is used to aid in ensuring that the TOE's design is sound, it also contributes to ensuring the implementation is correctly realized from the design. It is expected that evaluators will use the low-level design as an aid in understanding the implementation representation. The low-level design requirements ensure the evaluators have enough information to intelligently analyze (e.g., the documented interface descriptions of the modules match the entry points in the module, error codes returned by the functions in the module are consistent with those identified in the documentation) the implementation and ensure it is consistent with the design. ALC_TAT.1 provides evaluators with information necessary to understand the implementation representation and what the resulting implementation will consist of. Critical areas (e.g., the use of libraries, what definitions are used, compiler options) are documented so the evaluator can determine how the

Objective	Requirements Addressing the Objective	Rationale
		implementation representation is to be analyzed. ADV_RCR.1 is used here to provide the correspondence of the lowest level of decomposition (e.g., source code) to the adjoining level, low-level design. The correspondence analysis is used by the evaluator as a tool when determining if the low-level design is correctly reflected in the implementation representation.
O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.	ATE_COV.2 ATE_FUN.1 ATE_DPT.1 ATE_IND.2	In order to satisfy O.THOROUGH_FUNCTIONAL_ TESTING, the ATE class of requirements is necessary. The component ATE_FUN.1 requires the developer to provide the necessary test documentation to allow for an independent analysis of the developer's security functional test coverage. In addition, the developer must provide the test suite executables and source code, which are used for independently verifying the test suite results and in support of the test coverage analysis activities. ATE_COV.2 requires the developer to provide a test coverage analysis that demonstrates the TSFI are completely addressed by the developer's test suite. While exhaustive testing of the TSFI is not required, this component ensures that the security functionality of each TSFI is addressed. This component also requires an independent confirmation of the completeness of the test suite, which aids in ensuring that correct security relevant functionality of a TSFI is demonstrated through the testing effort. ATE_DPT.1 requires the developer to provide a test coverage analysis that demonstrates depth of coverage of the test suite. This component complements ATE_COV.2 by ensuring that the developer takes into account the high-level design when developing their test suite. Since exhaustive testing of the TSFI is not required, ATE_DPT.1 ensures that subtleties in TSF behavior that are not readily apparent in the functional specification are addressed in the test suite. ATE_IND.2 requires an independent confirmation of the developer's test results, by mandating a subset of the test suite be run by an independent party. This component also requires an independent party to attempt to craft functional tests that address functional behavior that is not demonstrated in the developer's test suite. Upon successful adherence to these requirements, the TOE's conformance to the specified security functional requirements will have been demonstrated.
O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	FPT_STM.1 FMT_MTD.1(3)	FPT_STM.1 requires that the TOE be able to provide reliable time stamps for its own use and therefore, partially satisfies this objective. Time stamps include date and time and are reliable in that they are always available to the TOE, and the clock must be monotonically increasing. FMT_MTD.1(3) satisfies the rest of this objective by providing the capability to set the time used for generating time stamps to either the Security Administrator, authorized IT entity, or both, depending on the selection made by the ST author.

O.TRUSTED_PATH
The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.
	FTP_ITC.1(1), FTP_ITC.1(2)
	FTP_TRP.1(1), FTP_TRP.1(2)
	FTP_ITC.1(3) (ENV), FTP_ITC.1(4) (ENV)
	FTP_TRP.1(3) (ENV), FTP_TRP.1(4) (ENV)
		FTP_TRP.1.1 requires the TOE to provide a mechanism that creates a distinct communication path that protects the data that traverses this path from disclosure or modification. This requirement ensures that the TOE can identify the end points and ensures that a user cannot insert themselves between the user and the TOE, by requiring that the means used for invoking the communication path cannot be intercepted and allow a "man-in-the-middle-attack" (this does not prevent someone from capturing the traffic and replaying it at a later time – see FPT_RPL.1). Since the user invokes the trusted path (FTP_TRP.1.2) mechanism they can be assured they are communicating with the TOE.

Objective	Requirements Addressing the Objective	Rationale
		FTP_TRP.1.3 mandates that the trusted path be the only means available for providing identification and authentication information, therefore ensuring a user's authentication data will not be compromised when performing authentication functions. Furthermore, the remote administrator's communication path is encrypted during the entire session. FTP_ITC.1(1) and FTP_ITC.1(2) are similar to FTP_TRP.1(1) and FTP_TRP.1(2), in that they require a mechanism that creates a distinct communication path with the same characteristics, however FTP_ITC.1(1) and FTP_ITC.1(2) is used to protect communications between IT entities, rather than between a human user and an IT entity. FTP_ITC.1.3 requires the TOE to initiate the trusted channel, which ensures that the TOE has established a communication path with an authorized IT entity and not some other entity pretending to be an authorized IT entity.
O.VIRUS The TOE will detect and block viruses contained within an information flow which arrives at any of the TOE network interfaces.	FAV_ACT_EXP.1	The FAV_ACT_EXP.1 requirement dictates that the TOE detect and block viruses which are contained within any information flow which reaches one of the TOE network interfaces. This requirement satisfies the O.VIRUS objective.
O.VULNERABILITY_ANALYSIS_TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with low attack potential to violate the TOE's security policies.	AVA_VLA.2	To maintain consistency with the overall assurance goals of this TOE, O.VULNERABILITY_ANALYSIS_TEST requires the AVA_VLA.2 component to provide the necessary level of confidence that vulnerabilities do not exist in the TOE that could cause the security policies to be violated. AVA_VLA.2 requires the developer to perform a systematic search for potential vulnerabilities in all the TOE deliverables. For those vulnerabilities that are not eliminated, a rationale must be provided that describes why these vulnerabilities cannot be exploited by a threat agent with a low attack potential, which is in keeping with the desired assurance level of this TOE. As with the functional testing, a key element in this component is that an independent assessment of the completeness of the developer's analysis is made, and more importantly, an independent vulnerability analysis coupled with testing of the TOE is performed. This component provides the confidence that security flaws do not exist in the TOE that could be exploited by a threat agent of low attack potential to violate the TOE's security policies.

Table 16 - Rationale for TOE Security Requirements

8.3 RATIONALE FOR ASSURANCE REQUIREMENTS

The selection of the EAL4+ level of assurance was made by Fortinet, Incorporated, in response to the needs of prospective clients.

8.4 RATIONALE FOR DEPENDENCIES

8.4.1 Rationale for Satisfying Functional Requirement Dependencies

Table 17 identifies the Security Functional Requirements from Part 2 of the CC and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency. Notes are provided for those cases where the dependencies are satisfied by components which are hierarchical to the specified dependency.

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
FAU_ARP.1	FAU_SAA.1	Yes	FAU_SAA.1 is in the ST
FAU_GEN.1	FPT_STM.1	Yes	FPT_STM.1 is in the ST
FAU_GEN.2	FAU_GEN.1	Yes	FAU_GEN.1 is in the ST
	FIA_UID.1	Yes	FIA_UID.2 is hierarchical and is in the ST
FAU_SAA.1	FAU_GEN.1	Yes	FAU_GEN.1 is in the ST
FAU_SAR.1	FAU_GEN.1	Yes	FAU_GEN.1 is in the ST
FAU_SAR.2	FAU_SAR.1	Yes	FAU_SAR.1 is in the ST
FAU_SAR.3	FAU_SAR.1	Yes	FAU_SAR.1 is in the ST
FAU_SEL.1	FAU_GEN.1	Yes	FAU_GEN.1 is in the ST
	FMT_MTD.1		FMT_MTD.1 is in the ST
FAU_STG.2	FAU_GEN.1	Yes	FAU_GEN.1 is in the ST
FAU_STG.3	FAU_STG.1	Yes	FAU_STG.2 is hierarchical and is in the ST
FAU_STG.4	FAU_STG.1	Yes	FAU_STG.2 is hierarchical and is in the ST
FCS_COP.1	[FDP_ITC.1 or	[No	
	FDP_ITC.2 or	No	
	FCS_CKM.1]	Yes]	FCS_CKM.1 is in the ST
	FCS_CKM.4	Yes	FCS_CKM.4 is in the ST
	FMT_MSA.2	Yes	FMT_MSA.2 is in the ST

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2	[Yes Yes] Yes Yes	FCS_CKM.2(1) and FCS_CKM.2(2) are in the ST. FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), and FCS_COP.1(4) are in the ST. FCS_CKM.4 is in the ST FMT_MSA.2 is in the ST
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FMT_MSA.2	[No No Yes] Yes	FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.1(3), FCS_CKM.1(4), and FCS_CKM.1(5) are in the ST. FMT_MSA.2 is in the ST
FDP_IFC.1	FDP_IFF.1	Yes	FDP_IFF.1(1), FDP_IFF.1(2), FDP_IFF.1(3), FDP_IFF.1(4) are in the ST
FDP_IFF.1	FDP_IFC.1 FMT_MSA.3	Yes Yes	FDP_IFC.1(1), FDP_IFC.1(2), FDP_IFC.1(3), FDP_IFC.1(4) are in the ST FMT_MSA.3 is in the ST
FDP_RIP.2	None	N/A	
FIA_AFL.1	FIA_UAU.1	Yes	FIA_UAU.1(1) and FIA_UAU.1(2) are in the ST
FIA_ATD.1	None	N/A	
FIA_UAU.1	FIA_UID.1	Yes	FIA_UID.2 is hierarchical and is in the ST
FIA_UAU.2	FIA_UID.1	Yes	FIA_UID.2 is hierarchical and is in the ST
FIA_UAU.5	None	N/A	
FIA_UID.2	None	N/A	
FIA_USB.1	FIA_ATD.1	Yes	FIA_ATD.1(1), FIA_ATD.1(2), and FIA_ATD.1(3) are in the ST

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
FMT_MOF.1	FMT_SMF.1 FMT_SMR.1	No Yes	See note below FMT_SMR.2 is hierarchical and is in the ST
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1] FMT_SMF.1 FMT_SMR.1	[No Yes] No Yes	FDP_IFC.1 is in the ST See note below FMT_SMR.2 is hierarchical and is in the ST
FMT_MSA.2	ADV_SPM.1 [FDP_ACC.1 or FDP_IFC.1] FMT_MSA.1 FMT_SMR.1	Yes [No Yes] Yes Yes	ADV_SPM.1 is in the ST FDP_IFC.1(1), FDP_IFC.1(2), FDP_IFC.1(3), FDP_IFC.1(4) are in the ST FMT_MSA.1 is in the ST FMT_SMR.2 is hierarchical and is in the ST
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1	Yes Yes	FMT_MSA.1 is in the ST FMT_SMR.2 is hierarchical and is in the ST
FMT_MTD.1	FMT_SMF.1 FMT_SMR.1	No Yes	See note below FMT_SMR.2 is hierarchical and is in the ST
FMT_MTD.2	FMT_MTD.1 FMT_SMR.1	Yes Yes	FMT_MTD.1 is in the ST FMT_SMR.2 is hierarchical and is in the ST
FMT_REV.1	FMT_SMR.1	Yes	FMT_SMR.2 is hierarchical and is in the ST
FMT_SMR.2	FIA_UID.1	Yes	FIA_UID.2 is hierarchical and is in the ST
FPT_AMT.1	None		
FPT_FLS.1	None		
FPT_ITA.1	None		
FPT_ITC.1	None		

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
FPT_ITI.1	None		
FPT_RCV.1	AGD_ADM.1 ADV_SPM.1	Yes Yes	AGD_ADM.1 is in the ST ADV_SPM.1 is in the ST
FPT_RPL.1	None	N/A	
FPT_RVM.1	None	N/A	
FPT_SEP.2	None	N/A	
FPT_STM.1	None	N/A	
FPT_TST.1	FPT_AMT.1	Yes	FPT_AMT.1 is in the ST
FRU_FLT.1	FPT_FLS.1	Yes	FPT_FLS.1 is in the ST
FRU_RSA.1	None	N/A	
FTA_SSL.1	FIA_UAU.1	Yes	FIA_UAU.1(1) and FIA_UAU.1(2) are in the ST
FTA_SSL.2	FIA_UAU.1	Yes	FIA_UAU.1(1) and FIA_UAU.1(2) are in the ST
FTA_SSL.3	None	N/A	
FTA_TAB.1	None	N/A	
FTA_TSE.1	None	N/A	
FTP_ITC.1	None	N/A	
FTP_TRP.1	None	N/A	

Table 17 - Security Functional Requirement Dependencies

Note: Although the FMT_SMF.1 requirement is a dependency of FMT_MOF.1, FMT_MSA.1 and FMT_MTD.1 it has not been included in this ST. The ST author concurs with the following rationale provided by the authors of the MR PPs:

> *'The requirements FMT_MOF.1, FMT_MSA.1 and FMT_MTD.1 express the functionality required by the TSF to provide the specified functions to manage TSF data, security attributes and management functions. These requirements make it*

clear that the TSF has to provide the functions to manage the identified data, attributes and functions. Therefore FMT_SMF.1 is not necessary.'

8.4.2 Rationale for Satisfying Assurance Requirement Dependencies

Table 18 identifies the Security Assurance Requirements from CC Part 3 and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency. Notes are provided for those cases where the dependencies are satisfied by components which are hierarchical to the specified dependency.

Security Assurance Requirement	Dependencies	Dependency Satisfied	Notes
ACM_AUT.1	ACM_CAP.3	Yes	ACM_CAP.4 is hierarchical and is in the ST
ACM_CAP.4	ALC_DVS.1	Yes	ALC_DVS.1 is in the ST
ACM_SCP.2	ACM_CAP.3	Yes	ACM_CAP.4 is hierarchical and is in the ST
ADO_DEL.2	ACM_CAP.3	Yes	ACM_CAP.4 is hierarchical and is in the ST
ADO_IGS.1	AGD_ADM.1	Yes	AGD_ADM.1 is in the ST
ADV_FSP.2	ADV_RCR.1	Yes	ADV_RCR.1 is in the ST
ADV_HLD.2	ADV_FSP.1 ADV_RCR.1	Yes Yes	ADV_FSP.2 is hierarchical and is in the ST ADV_RCR.1 is in the ST
ADV_IMP.1	ADV_LLD.1 ADV_RCR.1 ALC_TAT.1	Yes Yes Yes	ADV_LLD.1 is in the ST ADV_RCR.1 is in the ST ALC_TAT.1 is in the ST
ADV_LLD.1	ADV_HLD.2 ADV_RCR.1	Yes Yes	ADV_HLD.2 is in the ST ADV_RCR.1 is in the St
ADV_RCR.1	None	N/A	
ADV_SPM.1	ADV_FSP.1	Yes	ADV_FSP.2 is hierarchical and is in the ST
AGD_ADM.1	ADV_FSP.1	Yes	ADV_FSP.2 is hierarchical and is in the ST
AGD_USR.1	ADV_FSP.1	Yes	ADV_FSP.2 is hierarchical and is in the ST
ALC_DVS.1	None	N/A	
ALC_FLR.3	None	N/A	
ALC_LCD.1	None	N/A	
ALC_TAT.1	ADV_IMP.1	Yes	ADV_IMP.1 is in the ST
ATE_COV.2	ADV_FSP.1 ATE_FUN.1	Yes Yes	ADV_FSP.2 is hierarchical and is in the ST ATE_FUN.1 is in the ST
ATE_DPT.1	ADV_HLD.1 ATE_FUN.1	Yes Yes	ADV_HLD.2 is hierarchical and is in the ST ATE_FUN.1 is in the ST

Security Assurance Requirement	Dependencies	Dependency Satisfied	Notes
ATE_FUN.1	None	N/A	
ATE_IND.2	ADV_FSP.1 AGD_ADM.1 AGD_USR.1 ATE_FUN.1	Yes Yes Yes Yes	ADV_FSP.2 is hierarchical and is in the ST AGD_ADM.1 is in the ST AGD_USR.1 is in the ST ATE_FUN.1 is in the ST
AVA_MSU.2	ADO_IGS.1 ADV_FSP.1 AGD_ADM.1 AGD_USR.1	Yes Yes Yes Yes	ADO_IGS.1 is in the ST ADV_FSP.2 is hierarchical and is in the ST AGD_ADM.1 is in the ST AGD_USR.1 is in the ST
AVA_SOF.1	ADV_FSP.1 ADV_HLD.1	Yes Yes	ADV_FSP.2 is hierarchical and is in the ST ADV_HLD.2 is hierarchical and is in the ST
AVA_VLA.2	ADV_FSP.1 ADV_HLD.2 ADV_IMP.1 ADV_LLD.1 AGD_ADM.1 AGD_USR.1	Yes Yes Yes Yes Yes Yes	ADV_FSP.2 is hierarchical and is in the ST ADV_HLD.2 is in the ST ADV_IMP.1 is in the ST ADV_LLD.1 is in the ST AGD_ADM.1 is in the ST AGD_USR.1 is in the ST

Table 18 - Security Assurance Requirement Dependencies

8.5 RATIONALE FOR STRENGTH OF FUNCTION CLAIM

Part 1 of the CC defines “strength of function” in terms of the minimum efforts assumed necessary to defeat the expected security behavior of a TOE security function. There are three strength of function levels defined in Part 1: SOF-Basic, SOF-Medium and SOF-High. SOF-Basic is the strength of function level chosen for this ST. SOF-Basic states, “a level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by attackers possessing a low attack potential.” The rationale for choosing SOF-Basic was to be consistent with the TOE objective O.VULNERABILITY_ANALYSIS_TEST and assurance requirements included in this ST. Specifically, AVA_VLA.2 requires that the TOE be resistant to an attacker with a low attack potential, this is consistent with SOF-Basic.

FortiGate Unified Threat Management Solutions provide a level of protection that is appropriate against threat agents whose attack potential is low, in IT environments that require that information flows be controlled and restricted among network nodes where the FortiGate unit can be appropriately protected from physical attacks. The FortiGate unit’s management console must be controlled to restrict access to only authorized administrators. It is expected that the FortiGate units will be protected to the extent necessary to ensure that they remain connected to the networks they protect. The minimum strength of function, SOF-Basic is consistent with those requirements.

8.6 RATIONALE FOR EXPLICIT REQUIREMENTS

Table 19 presents the rationale for the inclusion of the explicit functional and assurance requirements found in the ST.

Explicit Requirement	Identifier	Rationale
FAU_ARP_ACK_EXP.1	Security alarm acknowledgement	This explicit requirement is necessary since a CC requirement does not exist to ensure an administrator will be aware of the alarm. The intent is to ensure that if an administrator is logged in and not physically at the console or remote workstation the message will remain displayed until the administrators have acknowledged it. The message will not be scrolled off the screen due to other activity-taking place (e.g., the auditor is running an audit report).
FAV_ACT_EXP.1	Anti Virus Actions	This explicit requirement is necessary since the CC does not provide a means to specify antivirus detection and blocking capabilities.
FCS_BCM_EXP.1	Baseline cryptographic module	This explicit requirement is necessary since the CC does not provide a means to specify a cryptographic baseline of implementation. This requirement specifies that all implemented cryptographic functions must be FIPS 140-2 validated to a stated FIPS 140-2 security level.
FIP_ACT_EXP.1	Intrusion Prevention Actions	This explicit requirement is necessary as the CC does not provide any requirements which specify the ability to detect and prevent intrusion attacks.
IDS_COL_EXP.1 IDS_RDR_EXP.1 IDS_STG_EXP.1	Sensor Data Collection Restricted Data Review Guarantee of Sensor Availability	A family of IDS requirements was created to specifically address the data collected and analysed by an IDS. The audit family of the CC (FAU) was used as a model for creating these requirements. The purpose of this family of requirements is to address the

Explicit Requirement	Identifier	Rationale
IDS_STG_EXP.2	Prevention of Sensor Data Loss	unique nature of IDS data and provide for requirements about collecting, reviewing and managing the data. These requirements have no dependencies since the stated requirements embody all the necessary security functions.

Table 19 - Rationale for Explicit Requirements

8.7 TOE SUMMARY SPECIFICATION RATIONALE

8.7.1 TOE Security Functions Rationale

Table 16 provides a bi-directional mapping of Security Functions to Security Functional Requirements from the CC Part 2. Table 17 provides a bi-directional mapping of Security Functions to the Explicit Security Functional Requirements. The tables, taken together, show that each of the SFRs is addressed by at least one of the Security Functions and that each of the Security Functions addresses at least one of the SFRs. The tables are followed by a discussion of how each Security Functional Requirement is addressed by the corresponding Security Function.

	F.ADMIN	F.AUDIT	F.CRYPTO	F.I&A	F.IFC	F.IPS	F.PROTECT	F.TRSTCOMM
FAU_ARP.1		X						
FAU_GEN.1		X						
FAU_GEN.2		X						
FAU_SAA.1	X	X						
FAU_SAR.1	X	X						
FAU_SAR.2	X	X						
FAU_SAR.3		X						
FAU_SEL.1	X	X						
FAU_STG.2	X	X					X	
FAU_STG.3		X						
FAU_STG.4	X	X					X	
FCS_CKM.1(1)			X					
FCS_CKM.1(2)			X					

	F.ADMIN	F.AUDIT	F.CRYPTO	F.I&A	F.IFC	F.IPS	F.PROTECT	F.TRSTCOMM
FCS_CKM.1(3)			X					
FCS_CKM.1(4)			X					
FCS_CKM.1(5)			X					
FCS_CKM.2(1)			X					
FCS_CKM.2(2)			X					
FCS_CKM.4			X					
FCS_COP.1(1)			X					
FCS_COP.1(2)			X					
FCS_COP.1(3)			X					
FCS_COP.1(4)			X					
FDP_IFC.1(1)					X			
FDP_IFC.1(2)					X			
FDP_IFC.1(3)					X			
FDP_IFC.1(4)					X			
FDP_IFF.1(1)	X				X			
FDP_IFF.1(2)	X				X			
FDP_IFF.1(3)	X				X			
FDP_IFF.1(4)	X				X			
FDP_RIP.2							X	
FIA_AFL.1	X			X				
FIA_ATD.1(1)	X			X				
FIA_ATD.1(2)	X			X				
FIA_ATD.1(3)	X			X				
FIA_UAU.1(1)				X				
FIA_UAU.1(2)				X				
FIA_UAU.2				X				
FIA_UAU.5				X				
FIA_UID.2				X				
FIA_USB.1				X				
FMT_MOF.1(1)	X							
FMT_MOF.1(2)	X							
FMT_MOF.1(3)	X							
FMT_MOF.1(4)	X							
FMT_MOF.1(5)	X							
FMT_MOF.1(6)	X							
FMT_MOF.1(7)	X							
FMT_MOF.1(8)	X							

	F.ADMIN	F.AUDIT	F.CRYPTO	F.I&A	F.IFC	F.IPS	F.PROTECT	F.TRSTCOMM
FMT_MOF.1(9)	X							
FMT_MOF.1(10)	X							
FMT_MOF.1(11)	X							
FMT_MOF.1(12)	X							
FMT_MOF.1(13)	X							
FMT_MSA.1	X							
FMT_MSA.2	X							
FMT_MSA.3(1)	X				X			
FMT_MSA.3(2)	X				X			
FMT_MTD.1(1)	X							
FMT_MTD.1(2)	X							
FMT_MTD.1(3)	X							
FMT_MTD.1(4)	X							
FMT_MTD.1(5)	X							
FMT_MTD.1(6)	X							
FMT_MTD.1(7)	X							
FMT_MTD.1(8)	X							
FMT_MTD.1(9)	X							
FMT_MTD.2(1)	X							
FMT_MTD.2(2)	X							
FMT_REV.1	X				X			
FMT_SMR.2	X							
FPT_AMT.1							X	
FPT_FLS.1							X	
FPT_ITA.1					X		X	
FPT_ITC.1							X	
FPT_ITI.1							X	
FPT_RCV.1							X	
FPT_RPL.1							X	
FPT_RVM.1					X		X	
FPT_SEP.2							X	
FPT_STM.1							X	
FPT_TST.1(1)	X		X				X	
FPT_TST.1(2)	X		X				X	
FRU_FLT.1							X	
FRU_RSA.1(1)							X	
FRU_RSA.1(2)							X	

	F.ADMIN	F.AUDIT	F.CRYPTO	F.I&A	F.IFC	F.IPS	F.PROTECT	F.TRSTCOMM
FTA_SSL.1	X			X			X	
FTA_SSL.2	X			X			X	
FTA_SSL.3	X			X			X	
FTA_TAB.1	X						X	
FTA_TSE.1							X	
FTP_ITC.1(1)			X					X
FTP_ITC.1(2)			X					X
FTP_TRP.1(1)			X					X
FTP_TRP.1(2)			X					X

Table 20 - Mapping of Security Functions to Security Functional Requirements from CC Part 2

	F.ADMIN	F.AUDIT	F.CRYPTO	F.I&A	F.IFC	F.IPS	F.PROTECT	F.TRSTCOMM
FAU_ARP_ACK_EXP.1		X						
FAV_ACT_EXP.1	X				X			X
FCS_BCM_EXP.1			X					
FIP_ACT_EXP.1	X				X			X
IDS_COL_EXP.1						X	X	
IDS_RDR_EXP.1						X	X	
IDS_STG_EXP.1						X	X	
IDS_STG_EXP.2						X	X	

Table 21 - Mapping of Security Functions to Explicit Security Functional Requirements

Security Functional Requirement	TOE Security Function	Rationale

Security Functional Requirement	TOE Security Function	Rationale
FAU_ARP.1 - Security alarms	F.AUDIT	Upon detecting a potential violation, the TOE immediately displays an alarm message identifying the potential security violation, generates an audible alarm (at the option of the Security Administrator), and makes accessible the audit record contents associated with the auditable event(s) that generated the alarm. The TOE displays alarm messages and sounds the audible alarm until the alarm has been acknowledged. The alarm message will be displayed and the audible alarm will sound at the Local Console regardless of whether or not an administrator is currently logged into the Local Console. The alarm message will display and the audible alarm will sound at any Network Management Stations which have administrative sessions at the time the potential security violation was detected. The alarm message will display and the audible alarm will sound at any Network Managements Stations which establish administrative sessions with the TOE before the alarm is acknowledged.
FAU_ARP_ACK_EXP.1 - Security alarm acknowledgement	F.AUDIT	The TOE will display alarm messages and sound the audible alarm until the alarm is acknowledged. At the Local Console, the TOE will repeat the display of the alarm message to ensure that the message is not scrolled off the display by other activity at the Local Console. Alarms can only be acknowledged by an Administrator who has successfully authenticated to the TOE through the Local Console, Network CLI or Network Web-based GUI. The TOE creates an audit record which includes the identity of the Administrator that acknowledged the alarm and the time the alarm was acknowledged. When an alarm is acknowledged, the TOE displays a message at the Local Console and at any Network Management Stations with administrative sessions identifying: • Administrator authentication failures; • the potential security violation which caused the alarm; • the identity of the administrator who acknowledged the alarm; and

Security Functional Requirement	TOE Security Function	Rationale
		• the time the alarm was acknowledged. This acknowledgement message will be displayed at the Local Console regardless of whether or not an administrator is currently logged into the Local Console. When using the Network Web-based GUI, acknowledgement is done by selecting the "OK" button on the alarm notification. When using the Local Console or Network CLI, the administrator must execute an "ack-alarm" command in order to acknowledge the alarm.
FAU_GEN.1 - Audit data generation	F.AUDIT	The TOE generates audit records for the startup and shutdown of the audit function and all of the events defined in Table 7. The TOE generates timestamps for all audit events and records the timestamp with each audit record. Also recorded are the type of event, identity of the user or subject which caused the event (if applicable), outcome of the event and any additional information listed in the third column of Table 7. Standard audit records are 512 bytes in length. If an audit record exceeds 512 bytes, it is wrapped into a second 512 byte audit record to ensure no audit detail is lost. The TOE has 7 different classes of audit records: • Event log – includes all system level events such as identification, authentication, configuration changes, audit record deletion, etc; • Traffic log – includes all data flow decisions, source/destination information, etc; • Antivirus log – includes any AV related events such as the detection of an infected file and the action taken; • Attack log – includes any IPS or local protection events such as DoS events, etc; • Web filter log (not part of evaluated configuration); • Antispam log (not part of evaluated configuration); and

Security Functional Requirement	TOE Security Function	Rationale
		• IM/P2P log (not part of evaluated configuration). Application Note: It is not possible to startup and shutdown the auditing function independently of the TOE. However since the TOE writes audit records for the startup and termination of each TOE subprocess (daemon), the audit trail will contain records which show the startup and shutdown of the auditing function coincident with the startup and shutdown of the TOE itself.
FAU_GEN.2 - User identity association	F.AUDIT	The identity of the administrator is recorded for all audited administrator initiated events such as configuration changes, audit log deletion and key loading. The identity of the authenticated proxy user is recorded for all audited information flow requests under the Authenticated Information Flow SFP.
FAU_SAA.1 - Potential violation analysis	F.AUDIT F.ADMIN	The Security Administrator can specify thresholds for the following events: • Administrator authentication failures; • Proxy user authentication failures; • Cryptographic encrypt/decrypt failures; • Replay attempts of TSF data or security attributes; • Self-test failures; • Firewall rule violations, based on source/destination address and port and time period; and • Protection Profile (Anti-Virus and/or IPS) violations. An alarm is triggered if the number of events exceeds the defined threshold. Application Note: Administrative guidance is provided which instructs the Security Administrator to set the thresholds for replay attempts and self-test failures to a value of 1. This ensures that an alarm is triggered for all detected replay attempts and all failures of the cryptographic and non-cryptographic self -tests

Security Functional Requirement	TOE Security Function	Rationale
FAU_SAR.1 - Audit review	F.AUDIT F.ADMIN	All administrative roles have read access to the audits records (Security Administrator, Audit Administrator and Crypto Administrator). The audit records can be accessed through the Local Console, Network CLI, and the Network Web-based GUI. When using the Network Web-Based GUI, administrators can view all audit records either as raw data (all columns) or as a filtered subset of columns. Filtered audit records (columns and rows) can be viewed through the Local Console or Network CLI. Administrators can modify the filters to change the view of the audit records. The number of records to display at one time can also be specified.
FAU_SAR.2 - Restricted audit review	F.AUDIT H.ADMIN	The TOE restricts access to all TOE administrative functions to authenticated administrators by assigned role. All administrative roles have read access to the audit records (Security Administrator, Audit Administrator and Crypto Administrator). Non-administrative users have no access to the audit log files and the data that they contain.
FAU_SAR.3 - Selectable audit review	F.AUDIT	The TOE supports selectable review (display) of audit data through the Local Console or the Network CLI. Log data can be filtered for display. Specific audit information can be specified as part of the filter. For example, the administrator could execute a CLI command to filter (list) all audit records with a specific source IP or all records between 2 dates. Filter criteria include (but are not limited to): • user identification (including a range of users); • source subject identity; • destination subject identity; • dates and times (from/to, included/excluded); • a range of one or more subject service identifiers; • a range of one of more transport layer protocols; • firewall rule identity; • TOE network interface; • log severity level (information, alert, emergency, critical, error, warning, notification, debug); and

Security Functional Requirement	TOE Security Function	Rationale
		- action (accept, deny).
FAU_SEL.1 - Selective audit	F.AUDIT F.ADMIN	The TOE allows the Security Administrator to modify the set of auditable events using the Local Console or the Network CLI. Events can be excluded from the audit record as they are written to the log storage device based on a filter. Filter parameters include (but are not limited to); administrator identity, proxy user identity, event type, network identifier, source/destination IP address, subject service identifier, success or failure of the auditable event and firewall rule identity.
FAU_STG.2 - Protected audit trail storage	F.AUDIT F.ADMIN F.PROTECT	Deletion of individual audit records, sets of audit records, and the audit logs themselves is restricted to administrators with the Audit Administrator role. No user (administrative or otherwise) has the ability to modify the records in the audit logs.
FAU_STG.3 - Action in case of possible audit loss	F.AUDIT F.ADMIN	By default, the TOE generates an audit record and sends an alarm when the log storage device reaches 75%, 90% and 95% of capacity. The alarm is sent to any remote administrative sessions that exist as well as to the Local Console (whether or not an administrative session exists). The alarm consists of a displayed message and optionally, an audible alarm. The Security Administrator can modify: • the thresholds at which an alarm is generated; • whether or not an audit record is created when the alarm is generated; and • whether or not an audible alarm is sounded when the alarm is generated.
FAU_STG.4 - Site-Configurable Prevention of Audit Loss	F.ADMIN F.AUDIT F.PROTECT	The TOE supports three different Security Administrator settable actions to prevent loss of audit data: • Shut down network interfaces (default action); • Overwrite audit records (FIFO); and • Stop logging The enabled action is taken once the log storage device reaches 95% capacity. If the “shut down network

Security Functional Requirement	TOE Security Function	Rationale
		interfaces" option is enabled, the TOE enters an error mode in addition to shutting down the network interfaces. The Security Administrator must clear the error mode by freeing space on the log storage device using the Local Console connection. By taking action when the log size reaches 95% of log storage capacity, the TOE ensures that the Security Administrator actions taken in order to resolve the log storage problem are themselves logged and that no audit records are lost. Application Note: Administrative guidance is provided which informs the Security Administrator that only the first two options are permitted in the evaluated configuration of the TOE.
FAV_ACT_EXP.1 – Anti Virus Actions	F.IFC F.ADMIN F.TRSTCOMM	The TOE detects and prevents virus attacks contained within information flows which arrive at any of its network interfaces. The Security Administrator can configure the TOE to block and or quarantine a virus which is detected in an information flow. The TOE provides a secure mechanism via a trusted channel for the update of virus signatures used by the TSF.
FCS_BCM_EXP.1 - Baseline cryptographic module	F.CRYPTO	All cryptographic functions implemented by the TOE that are FIPS-approved cryptographic functions are implemented in crypto modules which are FIPS 140-2 validated to an overall Security Level of 2 and which meet Level 3 for the following security requirements: • cryptographic module ports and interfaces; • roles, services and authentication; • cryptographic key management; and • design assurance. All cryptographic functions implemented by the TOE that are FIPS-approved cryptographic functions are implemented using a FIPS-approved mode of operation. • The FIPS 140-2 Security Policies for the TOE are available on the NIST website (http://csrc.nist.gov/cryptval/). There are several Security Policies that apply to the TOE based on the form factor of the FortiGate model.
FCS_CKM.1(1) - Cryptographic Key Management (key	F.CRYPTO	The FIPS-validated cryptomodule is used by the TOE to generate symmetric and asymmetric cryptographic keys.

Security Functional Requirement	TOE Security Function	Rationale
generation)		
FCS_CKM.1(2) - Cryptographic Key Management (Key Establishment for symmetric keys)	F.CRYPTO	The TOE performs Cryptographic Key Establishment using Discrete Logarithm Key Agreement for AES symmetric keys.
FCS_CKM.1(3) - Cryptographic Key Management (Key Entry for Digital Signature/Verification Private Keys)	F.CRYPTO	The TOE supports uploading public and private keys to use as custom RSA keys for administrator authentication. This is accomplished by uploading the signed public certificate from a FortiUSB token to the FortiGate unit. If the private key was not generated on the FortiGate unit, it also must be uploaded from the FortiUSB token. Certificates must have a modulus of at least 2048 bits. The FortiGate detects errors in the uploaded public certificate by verifying the certificate structure.
FCS_CKM.1(4) - Cryptographic Key Management (Key Entry for Digital Signature/Verification Private Keys)	F.CRYPTO	The TSF applies validation techniques to generated symmetric and asymmetric keys in accordance.
FCS_CKM.1(5) - Cryptographic Key Management (Internet Key Exchange)	F.CRYPTO	The TSF provides cryptographic key establishment techniques for internet key exchange.
FCS_CKM.2(1) - Cryptographic Key Management (Key Handling and Storage)	F.CRYPTO	The TSF provides for key handling and storage.
FCS_CKM.2(2) - Cryptographic Key Management (Key Distribution)	F.CRYPTO	The TSF performs manual and automated key distribution.
FCS_CKM.4 - Cryptographic key destruction	F.CRYPTO	The TOE destroys cryptographic keys in accordance with a cryptographic key zeroization method which meets the Key Zeroization Requirements of FIPS PUB 140-2 Key Management Security Level 3. The zeroization of all private cryptographic keys, plaintext cryptographic keys and all other critical cryptographic security parameters is immediate and complete. Zeroization of intermediate storage areas for private cryptographic keys, plaintext cryptographic keys and all other critical cryptographic security parameters is accomplished by overwriting the storage area three times with an alternating pattern.

Security Functional Requirement	TOE Security Function	Rationale
		The storage area for private cryptographic keys, plaintext cryptographic keys and all other critical cryptographic security parameters is a flash RAM device. Zeroization of these storage areas occurs when the Security Administrator executes a factory reset. All non-preconfigured keys and critical security parameters are zeroized by overwriting the storage area with zeroes.
FCS_COP.1(1) - Cryptographic operation (Encryption/Decryption AES)	F.CRYPTO	The cryptomodule used by the TOE to perform encryption and decryption uses the AES algorithm in CBC mode with key sizes of 128-bits, 192-bits and 256-bits. .
FCS_COP.1(2) - Cryptographic operation (Digital Signature Generation/Verification	F.CRYPTO	The FIPS-validated cryptomodule used by the TOE for digital signature generation and verification implements the rDSA algorithm with the following specification: • the cryptomodule implements the rDSA algorithm with a modulus size of 2048 bits in a manner which that conforms to ANSI X9.31-1998, Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA). • The choices and options used in conforming to the X9.31-1998 are as follows: o public verification exponent, e: generated; o supported hash algorithm: SHA-1; o private signature key options: d and n derived, p and q derived, SEED value(s) for generation of p and q generated on first boot from unit specific information; o calculation speed up values: none; o random number generation method used: ANSI X9.31 Appendix A; and o G Function: SHA-1. The TOE's implementation of rDSA is compliant with both X9.31-1998 and PKCS#1. X9.31 compliance is the default mode of operation for the TOE. The Security Administrator can disable X9.31 compliance (which enables PKCS#1 compliance) for compatibility with

Security Functional Requirement	TOE Security Function	Rationale
		commercially available products.
FCS_COP.1(3) - Cryptographic operation (Cryptographic Hash function)	F.CRYPTO	The TOE performs all cryptographic hashing functions using a FIPS-approved cryptographic hashing function implemented in a FIPS approved cryptomodule running in a FIPS-approved mode. The SHA-1 hashing algorithm is used for all cryptographic hashing functions.
FCS_COP.1(4) - Cryptographic operation (Random number generator)	F.CRYPTO	The TOE performs all random number generation using a FIPS-approved random number generator implemented in a FIPS-approved cryptomodule operating in a FIPS-approved mode. The random number generator specified by Appendix A of ANSI X9.31 is used by the TOE's FIPS-approved cryptomodule.
FDP_IFC.1(1) - Subset information flow control (unauthenticated policy)	F.IFC	The TOE permits the Security Administrator to define firewall rules which determine whether or not the TOE permits information (packets) to flow through the TOE without authentication of the user sending the information. The TOE may permit two general types of unauthenticated information flow: • Information flow through the TOE from a source to a destination; and • SMTP information flow via an application proxy. For information which flows via an application proxy, the TOE ensures that the connection from the source terminates at the TOE and that the connection between the TOE and the destination does not include any of the stateful protocol attributes associated with the subject. The Security Administrators may define firewall rules which permit (or deny) the flow of information based upon (but not limited to) the following criteria: • The TOE interface, VLAN or VDOM which originates the information flow (the source subject); • The TOE interface, VLAN or VDOM which is the destination of the information flow (the destination subject); and • The information contained within the information flow (packet contents).

Security Functional Requirement	TOE Security Function	Rationale
FDP_IFC.1(2) - Subset information flow control (authenticated policy)	F.IFC	The TOE permits the Security Administrator to define firewall rules which determine whether or not the TOE permits information (packets) to flow through the TOE without authentication of the user sending the information. The TOE may permit two general types of unauthenticated information flow: • Information flow through the TOE from a source to a destination; and • SMTP information flow via an application proxy. For information which flows via an application proxy, the TOE ensures that the connection from the source terminates at the TOE and that the connection between the TOE and the destination does not include any of the stateful protocol attributes associated with the subject. The Security Administrators may define firewall rules which permit (or deny) the flow of information based upon (but not limited to) the following criteria: • The TOE interface, VLAN or VDOM which originates the information flow (the source subject); • The TOE interface, VLAN or VDOM which is the destination of the information flow (the destination subject); and • The information contained within the information flow (packet contents).
FDP_IFC.1(3) - Subset information flow control (unauthenticated TOE services policy)	F.IFC	The TOE permits the Security Administrator to define firewall rules which determine whether or not the TOE responds to information flows which request access to TOE services without requiring authentication of the user sending the information. The Security Administrators may define firewall rules which accept (or reject) packets which include unauthenticated requests for access to TOE services based upon (but not limited to) the following criteria: • The TOE interface, VLAN or VDOM which originates the information flow (the source subject); and

Security Functional Requirement	TOE Security Function	Rationale
		The TOE service which is the subject of the information flow request.
FDP_IFC.1(4) - Subset information flow control (VPN policy)	F.IFC	The TOE permits the Security Administrator to define firewall rules which determine whether or not the TOE responds to information flows for VPN users. The Security Administrators may define firewall rules which accept (or reject) packets based upon (but not limited to) the TOE interface, VLAN or VDOM which originates the information flow (the source subject).
FDP_IFF.1(1) - Simple security attributes (unauthenticated policy)	F.ADMIN F.IFC	The TOE provides the Security Administrator with the ability to define a set of firewall rules which determine whether or not the TOE permits an information flow. The Security Administrator has the ability to specify the order in which the firewall rules are applied to requested information flows. The first rule which explicitly applies to the requested information flow is used to determine whether or not the information flow is accepted or rejected. If there are no rules which explicitly apply to the requested information flow, the information flow is rejected. The TOE also provides tools which allow the Security Administrator to view information flows allowed by the set of defined firewall rules before applying the ruleset. The criteria that the Security Administrator may use in order to define a firewall rule are listed in Section 5 of this document under FDP_IFF.1.1(1). The TOE completely reassembles fragmented packets before applying the firewall policy rules to the packets. The TOE implements stateful packet inspection rules in that each, non-fragmented, packet that is received by the TOE is either associated with an existing allowed connection, or is considered as an attempt to establish a new connection and therefore subject to the firewall rules. Regardless of other firewall rules, the TOE will deny an information flow if: • The source of the information flow is a broadcast identity; • The source if the information flow is a loopback identifier; • The information flow specifies the route of information flow from the source subject to the

Security Functional Requirement	TOE Security Function	Rationale
		destination subject; and The information flow is SMTP traffic that includes source routing symbols.
FDP_IFF.1(2) - Simple security attributes (authenticated policy)	F.ADMIN F.IFC	The TOE provides the Security Administrator with the ability to define a set of firewall rules which determine whether or not the TOE requires authentication in order to access an application proxy for a specific transport-layer protocol. The Security Administrator has the ability to specify the order in which the firewall rules are applied to requested information flows. The first rule which explicitly applies to the application proxy request is used to determine whether or not the request is accepted or rejected. If there are no rules which explicitly apply to the requested application proxy, the request is rejected. The TOE also provides tools which allow the Security Administrator to view information flows allowed by the set of defined firewall rules before applying the ruleset. The criteria that the Security Administrator may use in order to define a firewall rule for access to an application proxy which requires authentication are listed in Section 5 of this document under FDP_IFF.1.1(2). The TOE completely reassembles fragmented packets before applying the firewall policy rules to the packets. The TOE implements stateful packet inspection rules in that each, non-fragmented, packet that is received by the TOE is either associated with an existing allowed connection, or is considered as an attempt to establish a new connection and therefore subject to the firewall rules.
FDP_IFF.1(3) - Simple security attributes (unauthenticated TOE services policy)	F.ADMIN F.IFC	The TOE provides the Security Administrator with the ability to define a set of firewall rules which determine whether or not the TOE permits access to a specified TOE service without authentication. The Security Administrator has the ability to specify the order in which the firewall rules are applied to unauthenticated TOE service requests. The first rule which explicitly applies to the request is used to determine whether or not the unauthenticated TOE service request is accepted or rejected. If there are no rules which explicitly apply to the unauthenticated TOE service request, the request is rejected. The TOE also provides tools which allow the Security Administrator to view information flows allowed by the set of defined firewall rules before applying the ruleset. The criteria that the Security Administrator may use in order to define a firewall rule are listed in Section 5 of this

Security Functional Requirement	TOE Security Function	Rationale
		document under FDP_IFF.1.1(3). Regardless of other firewall rules, the TOE will deny an unauthenticated TOE service request if: • The source of the request is a broadcast identity; • The source of the request is a loopback identifier; and The request specifies the route of information flow from the source subject to the TOE.
FDP_IFF.1(4) - Simple security attributes (VPN Policy)	F.ADMIN F.IFC	The TOE permits the Security Administrator to define firewall rules which determine whether or not the TOE responds to information flows for VPN users.
FDP_RIP.2 - Full residual information protection	F.PROTECT	Users of the TOE do not have access to any of the TOE's resources. Users do not have access to the file system maintained by the TOE and there are no operating system commands which provide access to either memory or the file system. The only resource provided by the TOE to users, is the information content of packets transmitted by the TOE. Packets transmitted by the TOE are assembled in memory which has been overwritten by the TOE before allocation to the packet. This ensures that any previous information content of the memory is not revealed.
FIA_AFL.1- Authentication failure handling	F.ADMIN F.I&A	The TOE generates an alarm indicating a possible security violation when the number of consecutive unsuccessful attempts to establish a remote session, by a given user account, exceeds a maximum limit. The maximum limit is set by the Security Administrator. In addition to the generation of an alarm, the Security Administrator can specify whether or not exceeding the maximum number of login attempts results in the account becoming locked. If the Security Administrator specifies that the account does become locked, the Security Administrator also specifies the period of time for which the account is locked. Once a user account has been locked, that user may not establish a remote session with the TOE until the lockout time period has expired or the Security Administrator has taken action to unlock the account.

Security Functional Requirement	TOE Security Function	Rationale
		Application Note: The authentication failure limits apply to remote administrator authentication attempts and proxy user authentication attempts. The TOE does not enforce an authentication limit for the Local Console.
FIA_ATD.1(1) - User attribute definition (administrator)	F.ADMIN F.I&A	For each Administrator account maintained by the TOE, the following information is recorded: • The user identifier (user name); • The administrative role associated with the user identifier (Security Administrator, Audit Administrator, Cryptographic Administrator); • Password; • Optionally, up to three trusted host IP Address/Netmask pairs from which the administrator can establish a remote administrative session; and • Optionally, a virtual domain identifier associated with the administrative account.
FIA_ATD.1(2) - User attribute definition (authorized proxy user)	F.ADMIN F.I&A	For each proxy user account maintained by the TOE, the following information is recorded: • The user identifier (user name); • The role associated with the user identifier (proxy user); • Password; • Any user groups of which the user is a member; and • Any firewall policy rules applicable to that user.
FIA_ATD.1(3) - User attribute definition (VPN Remote Devices)	F.ADMIN F.I&A	For each VPN Remote Device account maintained by the TOE maintains IPSec Phase 1 and IPSec Phase 2 information.
FIA_UAU.1(1) Timing of authentication (for TOE services)	F.I&A	The Security Administrator may configure the TOE to provide ICMP Services to an unauthenticated user. The TOE requires authentication for access to all other TOE services.

Security Functional Requirement	TOE Security Function	Rationale
FIA_UAU.1(2) Timing of authentication (for information flow through the TOE)	F.I&A	The TOE will allow SMTP traffic to flow through the TOE (subject to the firewall rules) without requiring user authentication. The TOE requires authentication for all other information flows.
FIA_UAU.2 - Specified user authentication before any action	F.I&A	The following types of users (and remote IT entities) must be authenticated before the TOE will allow any action (except authentication) on behalf of that user: • Administrative users • Proxy users attempting to use the FTP, Telnet or HTTP transport-layer protocols; • The FortiGuard Distribution Server; and • VPN users.
FIA_UAU.5 - Authentication mechanism	F.I&A	The TOE provides a local password mechanism with a strength of function of SOF-Basic.
FIA_UID.2 - User identification before any action	F.I&A	The TOE requires user identification before taking any action on behalf of a user (information flow or TOE services). For authenticated users (administrators and proxy users) a user name is provided during the authentication process. For unauthenticated users, the Network Interface on which information is received by the TOE is considered to be the user identification.
FIA_USB.1 - User-subject binding	F.I&A	Administrators, proxy users, and VPN users are identified by the user name provided during the authentication process. All security attributes applicable to that user (as defined in Section 5 for FIA_ATD.1(1), FIA_ATD.1(2), FIA_ATD.1(3)) are then associated with the user's session. Unauthenticated users are identified by the source IP address from where the session is initiated. Only a single IP address can be used to identify a given session. However, a single IP address can be used to identify multiple sessions. There are no security attributes associated with unauthenticated users.
FIP_ACT_EXP.1 – Intrusion Prevention Actions	F.IFC F.ADMIN	The TOE prevents intrusion attacks directed at the TOE. The TOE also provides the Security Administrator will the ability to configure the TOE to detect and prevents intrusion attacks contained within information flows which arrive at any of its network interfaces. The TOE provides a

Security Functional Requirement	TOE Security Function	Rationale
	F.TRSTCOMM	secure mechanism (via a trusted channel) for the update of the intrusion prevention signatures used by the TSF.
FMT_MOF.1(1) - Management of security functions behavior (TSF non-cryptographic self-test)	F.ADMIN	While any administrator can execute the integrity verification self-tests, only the Security Administrator is able to specify the frequency for the automatic execution of these self-tests. The integrity verification self-tests cannot be disabled.
FMT_MOF.1(2) - Management of security functions behavior (cryptographic self-test)	F.ADMIN	While any administrator can execute the cryptographic self-tests on demand, only the Cryptographic Administrator is able to control whether or not the cryptographic self-tests are executed automatically every time a key is generated.
FMT_MOF.1(3) - Management of security functions behavior (audit and alarms)	F.ADMIN	Only Administrator accounts (Security Administrator, Audit Administrator and Cryptographic Administrator) are able to read the audit data. All administrator accounts are able to perform searches and sorts of the audit data based upon the criteria defined in Section 5 for the FAU_SAR.3 security functional requirement.
FMT_MOF.1(4) - Management of security functions behavior (audit and alarms)	F.ADMIN	Only the Security Administrator is able to define or modify the rules which are enforced by the TOE in order to determine whether a potential security violation has taken place. An alarm is generated when a potential security violation is detected. The Security Administrator may define/modify these rules as described in the bullet points below: • Specify the authentication failure limits for remote sessions; • Specify whether an account is locked after reaching the authentication failure limits for remote sessions and if an account is locked, specify the length of time for which it is locked; • Specify the information flow policy violation limits for a specified period of time; • Specify the encryption/decryption failure limits; • Specify the Protection Profile violations limits; • Specify the usage percentage limits for available audit storage; Only the Security Administrator is able to specify whether or not auditable events are included or excluded from the

Security Functional Requirement	TOE Security Function	Rationale
		audit trail based upon the criteria specified in Section 5 for the FAU_SEL.1 security functional requirement.
FMT_MOF.1(5) - Management of security functions behavior (audit and alarms)	F.ADMIN	The TOE will generate an alarm whenever a potential security violation is detected. Alarms consist of an audit record, an alarm message displayed on the Local Console and an alarm message displayed at remote administration sessions which either exist when the alarm is generated or which are initiated after the alarm is generated, but before the alarm is acknowledged. Optionally, an alarm may also include an audible signal at the Local Console and remote administration sessions which either exist when the alarm is generated or which are initiated after the alarm is generated, but before it is acknowledged. It is the responsibility of the Security Administrator to determine whether or not an alarm includes an audible signal.
FMT_MOF.1(6) - Management of security functions behavior (available TOE services for unauthenticated users)	F.ADMIN	Only the Security Administrator can specify whether or not ICMP, DHCP, DNS, or SMTP are available as a TOE service to unauthenticated users of the TOE.
FMT_MOF.1(7) - Management of security functions behavior (quota mechanism)	F.ADMIN	Only the Security Administrator can specify the maximum quota limits for connection-oriented resources (TCP sessions). The maximum quota limits may be specified on the basis of individual network identifiers, groups of network identifiers and/or schedules which include specific days/dates and times.
FMT_MOF.1(8) - Management of security functions behavior (cryptographic self-test frequency)	F.ADMIN	While any administrator can execute the cryptographic self-tests on demand, only the Security Administrator may specify the frequency with which the TOE automatically executes the cryptographic self-tests. This frequency may be set within the range of 1 to 480 minutes.
FMT_MOF.1(9) - Management of security functions behavior (audit storage exhaustion)	F.ADMIN	Only the Security Administrator may specify the action to be taken by the TOE in the event of audit storage exhaustion. The actions which may be taken by the TOE are listed under the FAU_STG.4 requirement.
FMT_MOF.1(10) - Management of security functions behavior (session termination)	F.ADMIN	Only the Security Administrator can specify the period of inactivity which causes an administrative, proxy user, or VPN user session to be terminated by the TOE.
FMT_MOF.1(11) - Management of security functions behavior (alarm acknowledgement)	F.ADMIN	All Administrators (Security Administrator, Audit Administrator and Cryptographic Administrator) can acknowledge alarms which indicated potential security violations. An audit record is created whenever an alarm is acknowledged.

Security Functional Requirement	TOE Security Function	Rationale
FMT_MOF.1(12) - Management of security functions behavior (self-tests)	F.ADMIN	All Administrators (Security Administrator, Audit Administrator and Cryptographic Administrator) can execute the cryptographic self-tests and the non-cryptographic self-tests manually, on demand.
FMT_MOF.1(13) – Management of security functions behavior (IDS sensor)	F.ADMIN	All Administrators (Security Administrator, Audit Administrator and Cryptographic Administrator) can manage the sensor data collection and review functions are defined by the IDS_COL_EXP.1 requirement.
FMT_MSA.1 - Management of security attributes	F.ADMIN	Only the Security Administrator can specify the attributes which are used to define the firewall rules which implement the security functional policies described in this document. For details of the attributes which may be specified by the Security Administrator for each of the security functional policies refer to the iterations of the FDP_IFF.1 requirement in Section 5.
FMT_MSA.2 – Secure security attributes	F.ADMIN	The TOE will only accept secure values for its security attributes.
FMT_MSA.3(1) - Static attribute initialization (ruleset)	F.IFC F.ADMIN	The TOE implements two security functional policies for information flow control; the UNAUTHENTICATED INFORMATION FLOW security functional policy and the AUTHENTICATED INFORMATION FLOW security functional policy. These policies are implemented in the TOE via a set of firewall rules which determine which information flows are permitted by the TOE. By default, in the evaluated configuration, no firewall rules are defined and therefore no traffic can flow through the TOE. The absence of any firewall rules in the default configuration is considered to be ‘restrictive default values’. The Security Administrator can modify the default configuration of the TOE by creating firewall rules which determine what traffic is allowed to flow through the TOE. The specification of firewall rules by the Security Administrator is considered to be the specification of ‘alternative initial values to override the default values’.
FMT_MSA.3(2) -Static attribute initialization (services)	F.IFC F.ADMIN	The TOE implements the UNAUTHENTICATE TOE SERVICES security functional policy in order to determine which TOE services are available to unauthenticated users. By default, in the evaluated configuration, no TOE services are available to unauthenticated users. This is considered to be ‘restrictive default values’. The Security Administrator can modify the configuration of the TOE to provide ICMP services to unauthenticated users. The specification of ICMP as a TOE service available to

Security Functional Requirement	TOE Security Function	Rationale
		unauthenticated users is considered to be the specification of ‘alternative initial values to override the default values’.
FMT_MTD.1(1) - Management of TSF data (audit data)	F.ADMIN	Only the Audit Administrator may delete audit data. The TOE prevents all modifications (except deletion) to the audit data.
FMT_MTD.1(2) - Management of TSF data (cryptographic TSF data)	F.ADMIN	Only the Cryptographic Administrator has the ability to load cryptographic keys into the TOE using a FortiUSB token.
FMT_MTD.1(3) - Management of TSF data (time TSF data)	F.ADMIN	Only the Security Administrator has the ability to modify the time and date setting of the TOE’s hardware clock.
FMT_MTD.1(4) - Management of TSF data (information flow policy ruleset)	F.ADMIN	Only the Security Administrator has the ability to query, modify, delete, and create the firewall rules.
FMT_MTD.1(5) - Management of TSF data (user accounts)	F.ADMIN	Only the Security Administrator has the ability to create and subsequently modify user accounts. User accounts include all administrative accounts (Security Administrator, Audit Administrator and Cryptographic Administrator) as well as proxy user accounts.
FMT_MTD.1(6) - Management of TSF data (TOE banner)	F.ADMIN	On the Security Administrator has the ability to modify the TOE banner which is displayed to authenticated users of the TOE.
FMT_MTD.1(7) - Management of TSF data (AV and IPS signatures)	F.ADMIN	The TOE uses AV and IPS signatures in Protection Profiles which may be specified as attributes in firewall rules. The AV and IPS signatures may be updated automatically by Fortinet’s FortiGuard Distribution Server (push) or by the Security Administrator, either manually or via a download from the FortiGuard Distribution Server (pull).
FMT_MTD.1(8) – Management of TSF Data (VPN Policy Ruleset)	F.ADMIN	The TOE implements its VPN security functional policy via firewall rules. Only the Security Administrator has the ability to manipulate these firewall rules.
FMT_MTD.1(9) – Management of TSF Data (IDS Sensor Data)	F.ADMIN	Only the Security Administrator has the ability to query the IDS Sensor data.
FMT_MTD.2(1) - Management of limits on TSF data (transport-layer quotas)	F.ADMIN	Only the Security Administrator may create or modify the maximum transport-layer quotas. The Security Administrator is also responsible for specifying the action to be taken by the TOE in the event that the maximum quota is exceeded. The Security

Security Functional Requirement	TOE Security Function	Rationale
		Administrator may specify one of the following actions: • clear session; • drop; • drop session; • pass; • pass session; • reset; • reset client; or • reset server.
FMT_MTD.2(2) - Management of limits on TSF data (controlled connection-oriented quotas)	F.ADMIN	Only the Security Administrator may create or modify the maximum quota for connections. The Security Administrator is also responsible for specifying the action to be taken by the TOE in the event that the maximum quota for connections is exceeded. The Security Administrator may specify one of the following actions: • clear session; • drop; • drop session; • pass; • pass session; • reset; • reset client; or • reset server.
FMT_REV.1 - Revocation	F.IFC F.ADMIN	Only the Security Administrator may modify (including disable) or delete the account of another administrator. Only the Security Administrator may modify (including disable) or delete proxy user accounts.

Security Functional Requirement	TOE Security Function	Rationale
		Only the Security Administrator may modify the firewall rules. Only the Security Administrator may modify the list of TOE services which require authentication. When the account of an administrator, proxy user, or VPN user is disabled or deleted, any sessions belonging to that account are immediately terminated by the TOE. Security Administrator modification to the list of TOE services which require authentication are applied immediately after the Security Administrator completes the modification. System Administrator modifications to the firewall rules are applied immediately after the Security Administrator completed the modification.
FMT_SMR.2 - Restrictions on security roles	F.ADMIN	The TOE maintains the following four roles: • Security Administrator; • Cryptographic Administrator; • Audit Administrator; • Authenticated Proxy User; and • VPN User. All user identities who authenticate to the TOE will be associated with one or more of these roles. All user identities who are associated with one of the administrative roles are able to establish an administrative session via the Local Console, the Network Web-Based GUI and the Network CLI. All administrative roles are distinct in that there is no overlap of operations performed by each role, except: • all administrators are able to review the audit trail; and • all administrators are able to invoke the self-tests (cryptographic and non-cryptographic).

Security Functional Requirement	TOE Security Function	Rationale
FPT_AMT.1 – Abstract machine testing	F.PROTECT	The TSF includes a suite of self-tests which may be executed to demonstrate the correct operation of the abstract machine which underlies the security functional policies of the TSF.
FPT_FLS.1 - Failure with preservation of secure state	F.PROTECT	The TOE preserves its secure state following the failure of a unit in a FortiGate cluster. If the failed unit is a slave in the cluster, no additional data is transferred to that unit. If the failed unit is the master unit in the cluster, one of the slaves units is promoted to become the new master unit for the cluster.
FPT_ITA.1 – Inter-TSF availability within a defined availability metric	F.PROTECT	The TOE is capable of transferring IDS data (audit and sensor data) to a remote trusted IT product (FortiAnalyzer). This transfer will take place within one minte of the TOE receiving a request for data transfer from an authenticated FortiAnalyzer.
FPT_ITC.1 – Inter-TSF confidentiality during transmission	F.PROTECT	The TSF protects the confidentiality of IDS data which is transferred to a trusted remote IT product (FortiAnalyzer) via encryption.
FPT_ITI.1 – Inter-TSF detection of modification	F.PROTECT	The TSF provides a cryptographic hash of transmitted information to a remote trusted IT product so that the remote trusted IT product may verify the integrity of the transmitted data.
FPT_RCV.1 - Manual recovery	F.PROTECT	The TOE enters its FIPS-CC Error Mode when any of the following are detected: • Failure of an integrity verification self-test; • Failure of a cryptographic self-test; and • Audit log size reaches 95% of the allocated audit log storage capacity and the 'shutdown network interfaces' option is in effect. This mode provides the ability to return the TOE to a secure state.
FPT_RPL.1 - Replay detection	F.AUDIT F.PROTECT	The TOE detects attempted replay of TSF data and security attributes. When a replay attack is detected the TOE drops the packets containing the replayed data, generates an alarm and creates an audit record to record the details of the attack.
FPT_RVM.1 - Non-	F.IFC	The TSF ensures that TSP enforcement functions are

Security Functional Requirement	TOE Security Function	Rationale
bypassability of the TSP		invoked and succeed before each function within the TSC is allowed to proceed.
FPT_SEP.2 - SFP domain separation	F.PROTECT	The unisolated portion of the TSF maintains a protected security domain for its own execution that protects it from interference and tampering by untrusted subjects. The TSF enforces separation between the security domains of subjects in the TSC. The TSF maintains an address space for the execution of cryptographic functions that is protects from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to the cryptographic functions.
FPT_STM.1 - Reliable time stamps	F.AUDIT F.PROTECT	The TOE includes a hardware clock which is used to generate reliable time stamps which in turn are used for audit records and to provide scheduling features for flow control policies. The hardware clock does not rely upon any external factors in order to function correctly. The time setting of the hardware clock may only be modified by the System Administrator and all such modifications are recorded in the audit log. The integrity of the hardware clock is verified during the TOE self-tests.
FPT_TST.1(1) - TSF testing (with cryptographic integrity verification)	F.CRYPTO F.ADMIN F.PROTECT	The TOE maintains, in its flash memory, a HMAC SHA-1 digest value for its firmware and TSF data (configuration data). The stored values are updated whenever the TOE firmware is updated and whenever a change is made to the configuration data. The TOE performs a series of integrity verification self-tests at startup to ensure the integrity of the TOE firmware and TSF data (excluding audit data). The tests calculate separate HMAC SHA-1 digest values for the TOE firmware and the TSF data. The calculated values are compared with values which were calculated previously and which are stored on the flash memory file system. If the values do not match, the TOE enters its FIPS-CC Error Mode. The success or failure of the integrity verification self-tests is displayed on local console as each test is completed. The integrity verification self-tests may be run manually by any of the administrators. The tests also run periodically at

Security Functional Requirement	TOE Security Function	Rationale
		a frequency specified by the Security Administrator.
FPT_TST.1(2) – TSF Testing (Cryptographic self-test)	F.CRYPTO F.ADMIN F.PROTECT	The TOE performs a series of cryptographic self-tests at startup to ensure the integrity of the cryptographic functions. The self-tests include: AES, 3DES, SHA-1, HMAC-SHA1, RNG and HW-Acceleration. The success or failure of each cryptographic self-test is displayed on local console as the execution of the test is completed. If one of the cryptographic self-tests fails, the TOE enters its FIPS-CC Error Mode. The cryptographic self-tests will also be executed periodically by the TOE at a Security Administrator-specified interval which may not be less than once per day. The Security Administrator may also configure the TOE such that the cryptographic self-tests are executed immediately after the generation of a key. The cryptographic self-tests can also be run manually by an administrator via a Local Console session or a Network CLI session.
FRU_FLT.1 - Degraded fault tolerance	F.PROTECT	The status of each node in a clustered TOE is identified by a heartbeat. When the heartbeat response is not received from a slave node, the master node no longer routes packets to the failed node. In the event that the master fails, an existing node in the cluster will be promoted to become the master node.
FRU_RSA.1(1) - Maximum quotas (transport-layer quotas)	F.PROTECT	The Security Administrator can set a maximum quota for the amount of data received by a subject (source or destination) in a specified period of time. If a maximum quota has been set by the Security Administrator, this quota will be enforced by the TOE.
FRU_RSA.1(2) - Maximum quotas (controlled connection-oriented quotas)	F.PROTECT	When the number of concurrent connection attempts from a given host exceeds the value defined by the Security Administrator, the TOE considers the traffic to be an attempted Denial of Service attack and all further requests matching this pattern are dropped.
FTA_SSL.1 - TSF-initiated session locking	F.ADMIN F.I&A F.PROTECT	Only administrative uses may establish Local Console interactive sessions. After a Security Administrator specified period of inactivity (which may be set from 1 to 480 minutes), the TOE terminates the inactive Local Console interactive session. When terminating the Local Console session, the TOE issues sufficient carriage return

Security Functional Requirement	TOE Security Function	Rationale
		characters to ensure that the current contents of the display device are unreadable. No other activity may then be performed at the Local Console until another administrative session is established via the identification and authentication process.
FTA_SSL.2 - User-initiated locking	F.ADMIN F.I&A F.PROTECT	Only administrative users may establish Local Console interactive sessions. Guidance is provided to the administrators instructing them to terminate their Local Console interactive sessions if it is necessary to leave the Local Console unattended. When a Local Console session is terminated, the TOE issues sufficient carriage return characters to ensure that the current contents of the display device are unreadable. No other activity may then be performed at the Local Console until another administrative session is established via the identification and authentication process.
FTA_SSL.3 - TSF-initiated termination	F.ADMIN F.I&A F.PROTECT	All proxy user sessions, VPN user sessions, and administrator sessions are subject to a time out value. When a session is inactive for a period of time which exceeds this value, the session is terminated by the TOE. The Security Administrator may set the timeout value in the range from 1 to 480 minutes. The timeout values for proxy user sessions and administrator sessions are independent and may be set to different values by the Security Administrator.
FTA_TAB.1 - Default TOE access banners	F.ADMIN F.PROTECT	The TOE provides a system banner. The system banner is presented when a proxy user or an administrator attempts to establish a connection with the TOE. The user must indicate acceptance of the system banner before a connection to the TOE is created. The Security Administrator is able to modify the contents of the system banner.
FTA_TSE.1 - TOE session establishment	F.PROTECT	Authorized Proxy User sessions, Network GUI sessions and Network CLI sessions, and VPN users sessions can only be established when a firewall rule exists which explicitly permits the connection. Firewall rules may be defined which are based on specific network interfaces, source IP addresses or address ranges and recurring schedule profiles. A schedule profile defines a time and date range over which connections are allowed (or denied).

Security Functional Requirement	TOE Security Function	Rationale
FTP_ITC.1(1) - Inter-TSF trusted channel (Prevention of Disclosure)	F.CRYPTO F.TRSTCOMM	Communications between the TOE and the FortiGuard Distribution Server use a trusted communication channel in order to transfer updates of IPS attack signatures and virus definitions from the FortiGuard Distribution Server to the TOE. The trusted communication channel may be invoked either by the TOE (to pull an update) or by the FortiGuard Distribution Server (to push an update). The TOE is delivered with a preset value for the public key of the FortiGuard Distribution Server. Therefore no administrative activity is required to configure or maintain the trusted channel. Once established, the trusted communication channel is uniquely identified by the source and destination. Data transmitted via the channel is protected from disclosure through the use of FIPS 140-2 validated encryption. The integrity of the data transmitted via the channel is ensured through the use of FIPS 140-2 validated cryptographic signatures.
FTP_ITC.1(2) - Inter-TSF trusted channel (Detection of Modification)	F.CRYPTO F.TRSTCOMM	Communications between the TOE and the FortiGuard Distribution Server use a trusted communication channel in order to transfer updates of IPS attack signatures and virus definitions from the FortiGuard Distribution Server to the TOE. The trusted communication channel may be invoked either by the TOE (to pull an update) or by the FortiGuard Distribution Server (to push an update). The TOE is delivered with a preset value for the public key of the FortiGuard Distribution Server. Therefore no administrative activity is required to configure or maintain the trusted channel. Once established, the trusted communication channel is uniquely identified by the source and destination. Data transmitted via the channel is protected from disclosure through the use of FIPS 140-2 validated encryption. The integrity of the data transmitted via the channel is ensured through the use of FIPS 140-2 validated cryptographic signatures.
FTP_TRP.1(1) - Trusted path (Prevention of Disclosure)	F.CRYPTO F.TRSTCOMM	The TOE supports remote administrator connections via the Network GUI over HTTPS and Network CLI sessions over SSH. In order to establish the session, the

Security Functional Requirement	TOE Security Function	Rationale
		administrator credentials are required (user id, password) and the session must be established from an authorized host. All communications between the remote administrator and the TOE are via the established trusted communications path. Once established, the trusted communication path is uniquely identified by the source and destination. Data transmitted via the path is protected from disclosure through the use of FIPS 140-2 validated encryption. The integrity of data transmitted via the path is ensured through the use of FIPS 140-2 validated cryptographic signatures. Proxy users are required to authenticate to the TOE (via HTTPS) by providing appropriate credentials (user id and password) before using defined proxy services. Once a proxy user has been successfully authenticated, further encryption of the communications between the user and the TOE is dependent on the requested proxy service.
FTP_TRP.1(2) - Trusted path (Detection of Modification)	F.CRYPTO F.TRSTCOMM	The TOE supports remote administrator connections via the Network GUI over HTTPS and Network CLI sessions over SSH. In order to establish the session, the administrator credentials are required (user id, password) and the session must be established from an authorized host. All communications between the remote administrator and the TOE are via the established trusted communications path. Once established, the trusted communication path is uniquely identified by the source and destination. Data transmitted via the path is protected from disclosure through the use of FIPS 140-2 validated encryption. The integrity of data transmitted via the path is ensured through the use of FIPS 140-2 validated cryptographic signatures. Proxy users are required to authenticate to the TOE (via HTTPS) by providing appropriate credentials (user id and password) before using defined proxy services. Once a proxy user has been successfully authenticated, further encryption of the communications between the user and the TOE is dependent on the requested proxy service.
IDS_COL_EXP.1 – Sensor data collection	F.IPS	The TOE is capable of acting as an IDS sensor by collecting network traffic (protocol, source address, destination address) according to criteria established by the Security Administrator (who acts as the IDS administrator)
IDS_RDR_EXP.1 – Restricted	F.IPS	All administrators have the ability to read the IDS

Security Functional Requirement	TOE Security Function	Rationale
data review		information collected by the TOE.
IDS_STG_EXP.2 – Guarantee of sensor data availability	F.IPS	The TOE prevents modification to the IDS sensor information and also prevents unauthorized deletion of this information. The Security Admistrator (acting as the IDS administrator) may specify how the TOE responds when data storage for IDS sensor data is exhausted.
IDS_STG_EXP.2 – Prevention of sensor data loss	F.IPS	The Security Administrator (acting as the IDS administrator) may elect to overwrite the oldest IDS Sensor data or prevent events that would result in the creation of new IDS sensor data in the event of reaching the storage capacity for IDS Sensor data.

Table 22 - TOE Security Functions Rationale

8.7.2 TOE Assurance Measures Rationale

Table 23 provides a bi-directional mapping of Assurance Measures to Assurance Requirements. It shows that each of the Assurance Requirements is addressed by at least one of the Assurance Measures and that each of the Assurance Measures addresses at least one of the Assurance Requirements. The table is followed by a short discussion of how the Assurance Requirements are addressed by the corresponding Assurance Measures.

	ACM_AUT.1	ACM_CAP.4	ACM_SCP.2	ADO_DEL.2	ADO_IGS.1	ADV_FSP.2	ADV_HLD.2	ADV_IMP.1	ADV_LLD.1	ADV_RCR.1	ADV_SPM.1	AGD_ADM.1	AGD_USR.1	ALC_DVS.1	ALC_FLR.3	ALC_LCD.1	ALC_TAT.1	ATE_COV.2	ATE_DPT.1	ATE_FUN.1	ATE_IND.2	AVA_MSU.2	AVA_SOF.1	AVA_VLA.2
M.ID		X																						
M.CMSYS	X	X	X																					
M.GETTOE				X																				
M.SETUP					X																			
M.SPEC						X	X		X															
M.IMPREP								X																
M.TRACE										X														
M.TOESPM											X													
M.DOCS												X	X		X									

	ACM_AUT.1	ACM_CAP.4	ACM_SCP.2	ADO_DEL.2	ADO_IGS.1	ADV_FSP.2	ADV_HLD.2	ADV_IMP.1	ADV_LLD.1	ADV_RCR.1	ADV_SPM.1	AGD_ADM.1	AGD_USR.1	ALC_DVS.1	ALC_FLR.3	ALC_LCD.1	ALC_TAT.1	ATE_COV.2	ATE_DPT.1	ATE_FUN.1	ATE_IND.2	AVA_MSU.2	AVA_SOF.1	AVA_VLA.2
M.DEVSEC														X										
M.FLAWREM			X												X									
M.LIFECYCLE																X								
M.DEVTOOLS																	X							
M.TESTCOV																		X						
M.TESTDPT																			X					
M.DEVTEST																				X				
M.INDTEST																					X			
M.VALIDANAL																						X		
M.SOFASS																							X	
M.VULANAL																								X

Table 23 - Mapping of Assurance Measures to Assurance Requirements

ACM_AUT.1 Partial CM automation

M.CMSYS satisfies the requirement for a CM system with automation support for change control and for TOE generation.

ACM_CAP.4 Generation support and acceptance procedures

M.ID and M.CMSYS combine to satisfy the requirement for a CM system that supports controlled generation of the TOE and acceptance of new or changed configuration items into the TOE.

ACM_SCP.2 Problem tracking CM coverage

M.CMSYS and M.FLAWREM combine to satisfy the requirement for controlling security flaws and tracking them to their resolution.

ADO_DEL.2 Detection of modification

M.GETTOE satisfies the requirement for defined delivery procedures with the ability to detect modifications to the TOE while in transit.

ADO_IGS.1 Installation, generation, and start-up procedures

M.SETUP satisfies the requirement for installation, generation and start-up procedures.

ADV_FSP.2 Fully defined external interfaces

M.SPEC satisfies the requirement for a functional specification with fully defined external interfaces.

ADV_HLD.2 Security-enforcing high-level design

M.SPEC satisfies the requirement for a security-enforcing high-level design.

ADV_IMP.1 Subset of the implementation of the TSF

M.IMPREP satisfies the requirement to provide a subset of the implementation of the TSF for review.

ADV_LLD.1 Descriptive low-level design

M.SPEC satisfies the requirement for a descriptive low-level design.

ADV_RCR.1 Informal correspondence demonstration

M.TRACE satisfies the requirement to informally demonstrate that more abstract TSF representations are correctly and completely refined into less abstract TSF representations.

ADV_SPM.1 Informal TOE security policy model

M.TOESPM satisfies the requirement for a model of the TSP.

AGD_ADM.1 Administrator guidance

M.DOCS satisfies the requirement for administrator guidance documentation.

AGD_USR.1 User guidance

M.DOCS satisfies the requirement for user guidance documentation.

ALC_DVS.1 Identification of security measures

M.DEVSEC satisfies the requirement to identify and documental developmental security measures.

ALC_FLR.3 Systematic flaw remediation

M.FLAWREM satisfies the requirement for systematically accepting and remediating security flaws. M.DOCS provides the documentation required to enable users to interact with the developers to report flaws and obtain corrections.

ALC_LCD.1	Developer defined life-cycle model

M.LIFECYCLE satisfies the requirement to establish and document a life-cycle model for TOE development and maintenance.

ALC_TAT.1	Well-defined development tools

M.DEVTOOLS satisfies the requirement for identification and documentation of the development tools being used for the TOE.

ATE_COV.2	Analysis of coverage

M.TESTCOV satisfies the requirement to provide an analysis of test coverage.

ATE_DPT.1	Testing: high-level design

M.TESTDPT satisfies the requirement to provide an analysis of the depth of testing to demonstrate that the TSF operates in accordance with its high-level design.

ATE_FUN.1	Functional Testing

M.DEVTEST satisfies the requirement to test the TSF and document the results.

ATE_IND.2	Independent testing – sample

M.INDTEST satisfies the requirement to support independent testing of a selected sample of the developer tests.

AVA_MSU.2	Validation of analysis

M.VALIDANAL satisfies the requirement to document an analysis of the completeness of the guidance documentation.

AVA_SOF.1	Strength of TOE security function evaluation

M.SOFASS satisfies the requirement for evidence that all TOE security functions have been examined to ensure their strengths against threats.

AVA_VLA.2	Independent vulnerability analysis

M.VULANAL satisfies the requirement to perform and document a vulnerability analysis.

9 REFERENCES

1) Common Criteria for Information Technology Security Evaluation, *CCIB-98-031 Version 2.1, August 1999.*

2) *Department of Defense Chief Information Officer Guidance and Policy Memorandum No. 6-8510*, Guidance and Policy for the Department of Defense Global Information Grid Information Assurance (GIG), *June 2000.*

3) U.S. Government Traffic-Filter Firewall Protection Profile for Medium Robustness Environments, *Version 1.4, May 1, 2000.*

4) U.S. Department of Defense Application-Level Firewall Protection Profile for Medium Robustness Environments, *Version 1.1, December 2001.*

5) Information Assurance Technical Framework, *Version 3.0, September 2000.*

6) *Federal Information Processing Standard Publication (FIPS-PUB) 46-3,* Data Encryption Standard (DES), *October 1999.*

7) *Federal Information Processing Standard Publication (FIPS-PUB) 140-2,* Security Requirements for Cryptographic Modules, *May 25, 2001.*

8) *Internet Engineering Task Force,* IP Encapsulating Security Payload (ESP), *RFC 2406, November 1998.*

9) *Internet Engineering Task Force,* Internet Key Exchange (IKE), *RFC 2409, November 1998.*

10) *Internet Engineering Task Force,* ESP CBC-Mode Cipher Algorithms, *RFC 2451, November 1998.*

11) *Internet Engineering Task Force,* Use of HMAC-SHA-1-96 within ESP and AH, *RFC 2404, November 1998.*

12) *Department of Defense Directive, Information Assurance,* 8500.1, October 24, 2002.

13) *Department of Defense Instruction, Information Assurance Implementation,* 8500.2, February 6, 2003.

14) *The AES Cipher Algorithm and Its Use with IPSec* <draft-ietf-ipsec-ciph-aes-cbc.03.txt>, Internet draft, November 2001.

15) *Federal Information Processing Standard Publication (FIPS-PUB) 197,* Specification for the Advanced Encryption Standard (AES), *November 26, 2001*

16) *NSA Glossary of Terms Used in Security and Intrusion Detection,* Greg Stocksdale, NSA Information Systems Security Organization, April 1998.

10 TERMINOLOGY

In the Common Criteria, many terms are defined in Section 2.3 of Part 1. The following are a definitions of terms used in this ST and common to the U.S. Government Traffic Filter Firewall Protection Profile for Medium Robustness Environments, as well as other DoD PPs.

Access -- Interaction between an entity and an object that results in the flow or modification of data.

Access Control -- Security service that controls the use of resources[70] and the disclosure and modification of data[71].

Accountability --Property that allows activities in an IT system to be traced to the entity responsible for the activity.

Administrator -- A user who has been specifically granted the authority to manage some portion or all of the TOE and whose actions may affect the TSP. Administrators may possess special privileges that provide capabilities to override portions of the TSP.

Assurance -- A measure of confidence that the security features of an IT system are sufficient to enforce its' security policy.

Asymmetric Cryptographic System -- A system involving two related transformations; one determined by a public key (the public transformation), and another determined by a private key (the private transformation) with the property that it is computationally infeasible to determine the private transformation (or the private key) from knowledge of the public transformation (and the public key).

Asymmetric Key -- The corresponding public/private key pair needed to determine the behavior of the public/private transformations that comprise an asymmetric cryptographic system.

Attack -- An intentional act attempting to violate the security policy of an IT system.

Authentication -- Security measure that verifies a claimed identity.

Authentication data -- Information used to verify a claimed identity.

Authorization -- Permission, granted by an entity authorized to do so, to perform functions and access data.

[70] Hardware and software.

[71] Stored or communicated

Authorized user -- An authenticated user who may, in accordance with the TSP, perform an operation.

Availability -- Timely[72], reliable access to IT resources.

Compromise -- Violation of a security policy.

Confidentiality -- A security policy pertaining to disclosure of data.

Critical Security Parameters (CSP) -- Security-related information (e.g., cryptographic keys, authentication data such as passwords and pins, and cryptographic seeds) appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a cryptographic module or the security of the information protected by the module.

Cryptographic Administrator -- An authorized user who has been granted the authority to perform cryptographic initialization and management functions. These users are expected to use this authority only in the manner prescribed by the guidance given to them.

Cryptographic boundary -- An explicitly defined contiguous perimeter that establishes the physical bounds (for hardware) or logical bounds (for software) of a cryptographic module.

Cryptographic key (key) -- A parameter used in conjunction with a cryptographic algorithm that determines [7]:

- the transformation of plaintext data into ciphertext data,
- the transformation of cipher text data into plaintext data,
- a digital signature computed from data,
- the verification of a digital signature computed from data, or
- a data authentication code computed from data.

Cryptographic Module -- The set of hardware, software, firmware, or some combination thereof that implements cryptographic logic or processes, including cryptographic algorithms, and is contained within the cryptographic boundary of the module.

Cryptographic Module Security Policy -- A precise specification of the security rules under which a cryptographic module must operate, including the rules derived from the requirements of this ST and additional rules imposed by the vendor.

[72] According to a defined metric.

Defense-in-Depth (DID) --A security design strategy whereby layers of protection are utilized to establish an adequate security posture for an IT system.

Discretionary Access Control (DAC) -- A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. These controls are discretionary in the sense that a subject with certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.

DMZ --A Demilitarized Zone (DMZ) is a network that is mediated by the TOE but, as a result of less stringent access controls, provides access to publicly available services, such as web servers.

Embedded Cryptographic Module -- One that is built as an integral part of a larger and more general surrounding system (i.e., one that is not easily removable from the surrounding system).

Enclave -- A collection of entities under the control of a single authority and having a homogeneous security policy. They may be logical, or may be based on physical location and proximity.

Entity -- A subject, object, user or another IT device, which interacts with TOE objects, data, or resources.

External IT entity -- Any trusted Information Technology (IT) product or system, outside of the TOE, which may, in accordance with the TSP, perform an operation.

Identity -- A representation (e.g., a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.

Integrity -- A security policy pertaining to the corruption of data and TSF mechanisms.

Integrity label --A security attribute that represents the integrity level of a subject or an object. Integrity labels are used by the TOE as the basis for mandatory integrity control decisions.

Integrity level -- The combination of a hierarchical level and an optional set of non-hierarchical categories that represent the integrity of data.

Mandatory Access Control (MAC) -- A means of restricting access to objects based on subject and object sensitivity labels[73].

Mandatory Integrity Control (MIC) -- A means of restricting access to objects based on subject and object integrity labels.

[73] The Bell LaPadula model is an example of Mandatory Access Control

Multilevel -- The ability to simultaneously handle (e.g., share, process) multiple levels of data, while allowing users at different sensitivity levels to access the system concurrently. The system permits each user to access only the data to which they are authorized access.

Named Object[74] -- An object that exhibits all of the following characteristics:

- The object may be used to transfer information between subjects of differing user identities within the TSF.
- Subjects in the TOE must be able to request a specific instance of the object.
- The name used to refer to a specific instance of the object must exist in a context that potentially allows subjects with different user identities to request the same instance of the object.

Non-Repudiation -- A security policy pertaining to providing one or more of the following:

- To the sender of data, proof of delivery to the intended recipient,
- To the recipient of data, proof of the identity of the user who sent the data.

Object -- An entity within the TSC that contains or receives information and upon which subjects perform operations.

Operating Environment --The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls.

Operating System (OS) -- An entity within the TSC that causes operations to be performed. Subjects can come in two forms: trusted and untrusted. Trusted subjects are exempt from part or all of the TOE security policies. Untrusted subjects are bound by all TOE security policies.

Operational key -- Key intended for protection of operational information or for the production or secure electrical transmissions of key streams.

Peer TOEs -- Mutually authenticated TOEs that interact to enforce a common security policy.

Public Object -- An object for which the TSF unconditionally permits all entities “read” access. Only the TSF or authorized administrators may create, delete, or modify the public objects.

[74] The only named objects in this ST, are operating system controlled files.

Robustness -- A characterization of the strength of a security function, mechanism, service or solution, and the assurance (or confidence) that it is implemented and functioning correctly. DoD has three levels of robustness:

- **Basic:** Security services and mechanisms that equate to good commercial practices.
- **Medium:** Security services and mechanisms that provide for layering of additional safeguards above good commercial practices.
- **High:** Security services and mechanisms that provide the most stringent protection and rigorous security countermeasures.

Secure State -- Condition in which all TOE security policies are enforced.

Security attributes -- TSF data associated with subjects, objects, and users that is used for the enforcement of the TSP.

Security level -- The combination of a hierarchical classification and a set of non-hierarchical categories that represent the sensitivity on the information [10].

Sensitivity label -- A security attribute that represents the security level of an object and that describes the sensitivity (e.g. Classification) of the data in the object. Sensitivity labels are used by the TOE as the basis for mandatory access control decisions [10].

Split key -- A variable that consists of two or more components that must be combined to form the operational key variable. The combining process excludes concatenation or interleaving of component variables.

Subject -- An entity within the TSC that causes operations to be performed.

Symmetric key -- A single, secret key used for both encryption and decryption in symmetric cryptographic algorithms.

Threat -- Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy.

Threat Agent - Any human user or Information Technology (IT) product or system which may attempt to violate the TSP and perform an unauthorized operation with the TOE.

User --Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

Vulnerability -- A weakness that can be exploited to violate the TOE security policy.

11 ACRONYMS, ABBREVIATIONS, AND INITIALIZATIONS

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

AES	Advanced Encryption Standard
AH	Authenticating Header
ANSI	American National Standards Institute
ARP	Address Resolution Protocol
ASIC	Application Specific Integrated Circuit
AV	Anti-Virus
BGP	Border Gateway Protocol
CBC	Cipher Block Chaining
CC	Common Criteria for Information Technology Security Evaluation
CCEVS	Common Criteria Evaluation and Validation Scheme
CEM	Common Evaluation Methodology
CLI	Command Line Interface
CM	Configuration Management
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
DAC	Discretionary Access Control
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DID	Defense In Depth
DMZ	Demilitarized zone
DNS	Domain Name System
DoD	Department of Defense

DoS	Denial of Service
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
ECDSA	Elliptic Curve Digital Signature Algorithm
ESP	Encapsulating Security Payload
FGCP	FortiGate Clustering Protocol
FIPS	Federal Information Processing Standard
FIPS PUB	Federal Information Processing Standard Publication
FTP	File Transfer Protocol
FW	Firewall
GIG	Global Information Grid
GUI	Graphical user interface
HA	High Availability
HMI	Human-Machine Interface
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol (Secure)
I&A	Identification and Authentication
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IDSS	Intrusion Detection System Sensor
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IM	Instant Messaging
IMAP	Internet Message Access Protocol

IP	Internet Protocol
IPS	Intrusion Prevention System
IPSEC	Internet Protocol Security
IT	Information Technology
LCD	Liquid Crystal Display
LDAP	Lightweight Directory Access Protocol
MAC	Mandatory Access Control
MIC	Mandatory Integrity Control
MIME	Multipurpose Internet Mail Extensions
N/A	Not Applicable
NAT	Network Address Translation
NBIAT&S	Network Boundary Information Assurance Technologies and Solutions Support
NIAP	National Information Assurance Partnership
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NSA	National Security Agency
NTP	Network Time Protocol
OS	Operating System
P2P	Peer to Peer
OSPF	Open Shortest Path First
PIN	Private Identification Number
POP3	Post Office Protocol Version 3
PKI	Public Key Infrastructure

RADIUS	Remote Authentication Dial In User Service
PP	Protection Profile
RFC	Request for Comments
RIP	Routing Information Protocol
RNG	Random Number Generator
ROBO	Remote Office or Branch Office
SFP	Security Function Policy
SFR	Security Functional Requirement
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SOF	Strength of Function
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TBD	To Be Determined
TCP	Transmission Control Protocol
TDEA	Triple Data Encryption Algorithm
TFFW	Traffic Filter Firewall
TFS	Terminal Final State
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TP	Transparent (Mode)
TSC	TSF Scope of Control

TSF	TOE Security Function
TSFI	TSF Interface
TSP	TOE Security Policy
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USB	Universal Serial Bus
VDOM	Virtual Domain
VLAN	Virtual Local Area Network
VPN	Virtual Private Network

--- End of Document ---