

**Assurance Activity Report for  
SonicWall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv  
Appliances**

Sonicwall SonicOS/X v7.0.1 with VPN and IPS on TZ, NSa, NSsp, and NSv  
Appliances Security Target Version 1.2

collaborative Protection Profile for Network Devices, Version 2.2e (CPP\_ND\_V2.2E)  
PP-Module for Intrusion Protection Systems (IPS), Version 1.0 (MOD\_IPS\_V1.0)  
PP-Module for Stateful Traffic Filter Firewalls, Version 1.4 + Errata 20200625 (MOD\_FW\_1.4E)  
PP-Module for Virtual Private Network (VPN) Gateways, Version 1.3 (MOD\_VPNGW\_1.3)

AAR Version 1.1, 24 December 2024

**Evaluated by:**



**2400 Research Blvd, Suite 395  
Rockville, MD 20850**

**Prepared for:**



**National Information Assurance Partnership  
Common Criteria Evaluation and Validation Scheme**

**The Developer of the TOE:**  
SonicWall, Inc.

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

**The TOE Evaluation was Sponsored by:**  
SonicWall, Inc.

**Evaluation Personnel:**  
Rupal Gupta  
Reema Nagwekar  
Halil Tosunoglu  
Yogita Kore

**Common Criteria Version**  
Common Criteria Version 3.1 Revision 5

**Common Evaluation Methodology Version**  
CEM Version 3.1 Revision 5

## REVISION HISTORY

VERSION	DATE	CHANGES
1.0	14 November, 2024	Initial Release
1.1	24 December, 2024	Initial Release

## CONTENTS

<b>1</b>	<b>TOE OVERVIEW.....</b>	<b>16</b>
<b>2</b>	<b>ASSURANCE ACTIVITIES IDENTIFICATION .....</b>	<b>17</b>
<b>3</b>	<b>TEST EQUIVALENCY JUSTIFICATION.....</b>	<b>18</b>
3.1	DIFFERENCES BETWEEN MODELS OF THE TOE.....	18
3.1.1	<i>Platform/Hardware Differences.....</i>	<i>18</i>
3.1.2	<i>Software/OS Dependencies: .....</i>	<i>18</i>
3.1.3	<i>Differences in Libraries Used to Provide TOE Functionality .....</i>	<i>19</i>
3.1.4	<i>TOE Management Interface Differences.....</i>	<i>19</i>
3.1.5	<i>TOE Functional Differences .....</i>	<i>19</i>
3.2	EQUIVALENCY CONCLUSIONS .....	21
<b>4</b>	<b>TEST BED DESCRIPTIONS .....</b>	<b>23</b>
4.1	AUDIT/FIREWALL/IPS/IPSEC/VPNGW/X509-REV.....	23
4.1.1	<i>Physical TOE.....</i>	<i>23</i>
4.1.2	<i>Virtual TOE.....</i>	<i>24</i>
4.2	AUTH/CRYPTO/UPDATE/TLSS.....	25
4.2.1	<i>Physical TOE.....</i>	<i>25</i>
4.2.2	<i>Virtual TOE.....</i>	<i>26</i>
4.3	CONFIGURATION INFORMATION .....	27
4.3.1	<i>Physical TOE.....</i>	<i>27</i>
4.3.2	<i>Virtual TOE.....</i>	<i>29</i>
4.4	TEST TIME AND LOCATION.....	30
<b>5</b>	<b>DETAILED TEST CASES (TSS AND AGD ACTIVITIES).....</b>	<b>31</b>
5.1	MANDATORY REQUIREMENTS.....	31
5.1.1	<i>Security Audit (FAU).....</i>	<i>31</i>
5.1.1.1	FAU_GEN.1 Audit Data Generation.....	31
5.1.1.1.1	FAU_GEN.1 TSS.....	31
5.1.1.1.2	FAU_GEN.1 AGD.....	31
5.1.1.2	FAU_GEN.1 Audit Data Generation (FFW) .....	34
5.1.1.2.1	FAU_GEN.1 TSS.....	34
5.1.1.2.2	FAU_GEN.1 AGD.....	34
5.1.1.3	FAU_GEN.1/VPN Audit Data Generation (VPN Gateway) .....	35
5.1.1.3.1	FAU_GEN.1/VPN TSS .....	35
5.1.1.3.2	FAU_GEN.1/VPN Audit Data Generation (VPN Gateway) AGD .....	37
5.1.1.4	Security Audit Data Generation for IPS Refinement (FAU_GEN) .....	37
5.1.1.4.1	FAU_GEN.1/IPS Audit Data Generation (IPS) TSS .....	37
5.1.1.4.2	FAU_GEN.1/IPS Audit Data Generation (IPS) AGD .....	39
5.1.1.5	FAU_GEN.2 User Identity Association.....	41
5.1.1.5.1	TSS & AGD .....	41
5.1.1.6	FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE .....	41
5.1.1.6.1	FAU_STG_EXT.1 TSS .....	41
5.1.1.6.2	FAU_STG_EXT.1 AGD.....	44

5.1.2	<i>Cryptographic Support (FCS)</i> .....	46
5.1.2.1	FCS_CKM.1 Cryptographic Key Generation .....	46
5.1.2.1.1	FCS_CKM.1 TSS .....	46
5.1.2.1.2	FCS_CKM.1 AGD .....	47
5.1.2.2	FCS_CKM.1/IKE Cryptographic Key Generation (for IKE Peer Authentication) .....	47
5.1.2.2.1	FCS_CKM.1/IKE TSS .....	47
5.1.2.2.2	FCS_CKM.1/IKE AGD .....	49
5.1.2.3	FCS_CKM.2 Cryptographic Key Establishment .....	50
5.1.2.3.1	FCS_CKM.2 TSS <b>[TD0580]</b> .....	50
5.1.2.3.2	FCS_CKM.2 AGD .....	52
5.1.2.4	FCS_CKM.4 Cryptographic Key Destruction .....	52
5.1.2.4.1	FCS_CKM.4 TSS .....	52
5.1.2.4.2	FCS_CKM.4 AGD .....	56
5.1.2.5	FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption).....	57
5.1.2.5.1	FCS_COP.1/DataEncryption TSS .....	57
5.1.2.5.2	FCS_COP.1/DataEncryption AGD.....	58
5.1.2.6	FCS_COP.1/ DataEncryption Cryptographic Operation (AES Data Encryption/Decryption) (VPNGW).....	58
5.1.2.6.1	FCS_COP.1/DataEncryption.....	58
5.1.2.7	FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification).....	59
5.1.2.7.1	FCS_COP.1/SigGen TSS .....	59
5.1.2.7.2	FCS_COP.1/SigGen AGD .....	59
5.1.2.8	FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm) .....	60
5.1.2.8.1	FCS_COP.1/Hash TSS .....	60
5.1.2.8.2	FCS_COP.1/Hash AGD.....	60
5.1.2.9	FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm) .....	61
5.1.2.9.1	FCS_COP.1/KeyedHash TSS .....	61
5.1.2.9.2	FCS_COP.1/KeyedHash AGD.....	62
5.1.2.10	FCS_RBG_EXT.1 Extended: Cryptographic Operation (Random Bit Generation).....	62
5.1.2.10.1	FCS_RBG_EXT.1 TSS.....	62
5.1.2.10.2	FCS_RBG_EXT.1 AGD .....	63
5.1.3	<i>User Data Protection (FDP)</i> .....	63
5.1.3.1	FDP_RIP.2 Full Residual Information Protection .....	63
5.1.3.1.1	FDP_RIP.2 TSS.....	63
5.1.4	<i>Identification and Authentication (FIA)</i> .....	64
5.1.4.1	FIA_AFL.1 Authentication Failure Management .....	64
5.1.4.1.1	FIA_AFL.1 TSS .....	64
5.1.4.1.2	FIA_AFL.1 AGD.....	65
5.1.4.2	FIA_PMG_EXT.1 Password Management.....	66
5.1.4.2.1	FIA_PMG_EXT.1 TSS <b>[TD0792]</b> .....	66
5.1.4.2.2	FIA_PMG_EXT.1 AGD.....	67
5.1.4.3	FIA_UIA_EXT.1 User Identification and Authentication .....	69
5.1.4.3.1	FIA_UIA_EXT.1 TSS .....	69
5.1.4.3.2	FIA_UIA_EXT.1 AGD.....	70
5.1.4.4	FIA_UAU_EXT.2 Password-based Authentication Mechanism.....	72
5.1.4.5	FIA_UAU.7 Protected Authentication Feedback .....	72
5.1.4.5.1	FIA_UAU.7 TSS.....	73
5.1.4.5.2	FIA_UAU.7 AGD .....	73
5.1.5	<i>Security Management (FMT)</i> .....	73

5.1.5.1	FMT_MOF.1/ManualUpdate	73
5.1.5.1.1	FMT_MOF.1/ManualUpdate TSS	73
5.1.5.1.2	FMT_MOF.1/ManualUpdate AGD	73
5.1.5.2	FMT_MTD.1/CoreData Management of TSF Data	75
5.1.5.2.1	FMT_MTD.1/CoreData TSS	75
5.1.5.2.2	FMT_MTD.1/CoreData AGD	76
5.1.5.3	FMT_SMF.1 Specification of Management Functions	78
5.1.5.3.1	FMT_SMF.1 TSS (containing also requirements on guidance documentation and tests)	78
5.1.5.3.2	FMT_SMF.1 AGD	81
5.1.5.4	FMT_SMF.1/FFW Specification of Management Functions	81
5.1.5.4.1	FMT_SMF.1/FFW	81
5.1.5.5	FMT_SMF.1/VPN Specification of Management Functions	82
5.1.5.5.1	FMT_SMF.1/VPN TSS	82
5.1.5.5.2	FMT_SMF.1/VPN AGD	83
5.1.5.6	Specification of Management Functions (FMT_SMF)	83
5.1.5.6.1	FMT_SMF.1/IPS Specification of Management Functions (IPS) TSS	83
5.1.5.6.2	FMT_SMF.1/IPS Specification of Management Functions (IPS) AGD	84
5.1.5.7	FMT_SMR.2 Restrictions on Security Roles	85
5.1.5.7.1	FMT_SMR.2 TSS	85
5.1.5.7.2	FMT_SMR.2 AGD	85
5.1.6	<i>Protection of the TSF (FPT)</i>	87
5.1.6.1	FPT_APW_EXT.1 Protection of Administrator Passwords	87
5.1.6.1.1	FPT_APW_EXT.1 TSS	87
5.1.6.2	FPT_SKP_EXT.1 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)	88
5.1.6.2.1	FPT_SKP_EXT.1 TSS	88
5.1.6.3	FPT_STM_EXT.1 Reliable Time Stamps	89
5.1.6.3.1	FPT_STM_EXT.1 TSS <b>[TD0632]</b>	89
5.1.6.3.2	FPT_STM_EXT.1 AGD <b>[TD0632]</b>	90
5.1.6.4	FPT_TST_EXT.1 TSF Testing	90
5.1.6.4.1	FPT_TST_EXT.1 TSS	90
5.1.6.4.2	FPT_TST_EXT.1 AGD	92
5.1.6.4.3	FPT_TST_EXT.1 (VPNGW)	93
5.1.6.5	FPT_TST_EXT.3 Self-Test with Defined Methods	93
5.1.6.5.1	FPT_TST_EXT.3 TSS	93
5.1.6.5.2	FPT_TST_EXT.3 AGD	94
5.1.6.6	FPT_TUD_EXT.1 Trusted Update	94
5.1.6.6.1	FPT_TUD_EXT.1 TSS	94
5.1.6.6.2	FPT_TUD_EXT.1 AGD	97
5.1.6.6.3	FPT_TUD_EXT.1 (VPNGW)	99
5.1.6.7	FPT_FLS.1/SelfTest Failure with Preservation of Secure State (Self-Test Failures)	99
5.1.6.7.1	FPT_FLS.1/SelfTest TSS	99
5.1.6.7.2	FPT_FLS.1/SelfTest AGD	100
5.1.7	<i>TOE Access (FTA)</i>	100
5.1.7.1	FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING	100
5.1.7.1.1	FTA_SSL_EXT.1 TSS	100
5.1.7.1.2	FTA_SSL_EXT.1 AGD	101
5.1.7.2	FTA_SSL.3 TSF-Initiated Termination	101
5.1.7.2.1	FTA_SSL.3 TSS	102

5.1.7.2.2	FTA_SSL.3 AGD .....	102
5.1.7.3	FTA_SSL.4 User-Initiated Termination .....	103
5.1.7.3.1	FTA_SSL.4 TSS .....	103
5.1.7.3.2	FTA_SSL.4 AGD .....	103
5.1.7.4	FTA_TAB.1 Default TOE Access Banners .....	103
5.1.7.4.1	FTA_TAB.1 TSS .....	104
5.1.7.4.2	FTA_TAB.1 AGD .....	104
5.1.8	<i>Trusted Path (FTP)</i> .....	105
5.1.8.1	FTP_ITC.1 Inter-TSF Trusted Channel .....	105
5.1.8.1.1	FTP_ITC.1 TSS .....	105
5.1.8.1.2	FTP_ITC.1 AGD.....	106
5.1.8.2	FTP_ITC.1/VPN Inter-TSF Trusted Channel (VPN Communications).....	108
5.1.8.2.1	FTP_ITC.1/VPN TSS .....	108
5.1.8.2.1.1	FTP_ITC.1/VPN AGD.....	108
5.1.8.3	FTP_TRP.1/Admin Trusted Path .....	108
5.1.8.3.1	FTP_TRP.1/Admin TSS .....	108
5.1.8.3.2	FTP_TRP.1/Admin AGD.....	109
5.1.9	<i>Firewall (FFW)</i> .....	110
5.1.9.1	FFW_RUL_EXT.1 Stateful Traffic Filtering .....	110
5.1.9.1.1	FFW_RUL_EXT.1 TSS.....	110
5.1.9.1.2	FFW_RUL_EXT.1 AGD .....	112
5.1.9.1.3	FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 TSS .....	113
5.1.9.1.4	FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 AGD.....	116
5.1.9.1.5	FFW_RUL_EXT.1.5 TSS.....	118
5.1.9.1.6	FFW_RUL_EXT.1.5 AGD .....	122
5.1.9.1.7	FFW_RUL_EXT.1.6 TSS.....	123
5.1.9.1.8	FFW_RUL_EXT.1.6 AGD .....	124
5.1.9.1.9	FFW_RUL_EXT.1.7 TSS.....	125
5.1.9.1.10	FFW_RUL_EXT.1.7 AGD .....	126
5.1.9.1.11	FFW_RUL_EXT.1.8 TSS [TD0545] .....	127
5.1.9.1.12	FFW_RUL_EXT.1.8 AGD .....	128
5.1.9.1.13	FFW_RUL_EXT.1.9 TSS.....	129
5.1.9.1.14	FFW_RUL_EXT.1.9 AGD .....	130
5.1.9.1.15	FFW_RUL_EXT.1.10 TSS.....	131
5.1.9.1.16	FFW_RUL_EXT.1.10 AGD .....	132
5.1.10	<i>Packet Filtering (FPF)</i> .....	133
5.1.10.1	FPF_RUL_EXT.1 Packet Filtering Rules TSS .....	133
5.1.10.1.1	FPF_RUL_EXT.1.1 TSS .....	133
5.1.10.1.2	FPF_RUL_EXT.1.1 AGD.....	135
5.1.10.1.3	FPF_RUL_EXT.1.2.....	135
5.1.10.1.4	FPF_RUL_EXT.1.3.....	135
5.1.10.1.5	FPF_RUL_EXT.1.4 TSS .....	135
5.1.10.1.6	FPF_RUL_EXT.1.4 AGD.....	138
5.1.10.1.7	FPF_RUL_EXT.1.5 TSS .....	140
5.1.10.1.8	FPF_RUL_EXT.1.5 AGD.....	141
5.1.10.1.9	FPF_RUL_EXT.1.6 TSS .....	142
5.1.10.1.10	FPF_RUL_EXT.1.6 AGD.....	143
5.1.11	<i>Intrusion Prevention System (IPS)</i> .....	144

5.1.11.1	Anamoly-Based IPS Functionality (IPS_ABD_EXT).....	144
5.1.11.1.1	IPS_ABD_EXT.1 Anamoly-Based IPS Functionality TSS .....	144
5.1.11.1.2	IPS_ABD_EXT.1 Anamoly-Based IPS Functionality AGD.....	146
5.1.11.2	IP Blocking (IPS_IPB_EXT) .....	148
5.1.11.2.1	IPS_IPB_EXT.1 IP Blocking TSS .....	148
5.1.11.2.2	IPS_IPB_EXT.1 IP Blocking AGD .....	150
5.1.11.3	Network Traffic Analysis (IPS_NTA_EXT) .....	150
5.1.11.3.1	IPS_NTA_EXT.1.1 Network Traffic Analysis TSS .....	150
5.1.11.3.2	IPS_NTA_EXT.1.1 Network Traffic Analysis AGD .....	151
5.1.11.3.3	IPS_NTA_EXT.1.2 Network Traffic Analysis TSS .....	152
5.1.11.3.4	IPS_NTA_EXT.1.2 Network Traffic Analysis AGD .....	153
5.1.11.3.5	IPS_NTA_EXT.1.3 Network Traffic Analysis TSS .....	153
5.1.11.3.6	IPS_NTA_EXT.1.3 Network Traffic Analysis AGD .....	154
5.1.11.4	Signature-Based IPS Functionality (IPS_SBD_EXT) .....	156
5.1.11.4.1	IPS_SBD_EXT.1.1 Signature-Based IPS Functionality TSS.....	156
5.1.11.4.2	IPS_SBD_EXT.1.1 Signature-Based IPS Functionality AGD <b>[TD0722]</b> .....	158
5.1.11.4.3	IPS_SBD_EXT.1.2 Signature-Based IPS Functionality TSS.....	160
5.1.11.4.4	IPS_SBD_EXT.1.2 Signature-Based IPS Functionality AGD .....	162
5.1.11.4.5	IPS_SBD_EXT.1.3 Signature-Based IPS Functionality TSS.....	163
5.1.11.4.6	IPS_SBD_EXT.1.3 Signature-Based IPS Functionality AGD .....	164
5.1.11.4.7	IPS_SBD_EXT.1.4 Signature-Based IPS Functionality TSS.....	165
5.1.11.4.8	IPS_SBD_EXT.1.4 Signature-Based IPS Functionality AGD .....	166
5.1.11.4.9	IPS_SBD_EXT.1.5 Signature-Based IPS Functionality TSS.....	167
5.1.11.4.10	IPS_SBD_EXT.1.5 Signature-Based IPS Functionality AGD .....	167
5.1.11.4.11	IPS_SBD_EXT.1.6 Signature-Based IPS Functionality TSS.....	167
5.1.11.4.12	IPS_SBD_EXT.1.6 Signature-Based IPS Functionality AGD .....	167
5.2	SELECTION-BASED REQUIREMENTS .....	171
5.2.1	<i>Cryptographic Support (FCS)</i> .....	171
5.2.1.1	FCS_HTTPS_EXT.1 HTTPS Protocol .....	171
5.2.1.1.1	FCS_HTTPS_EXT.1 TSS .....	171
5.2.1.1.2	FCS_HTTPS_EXT.1 AGD.....	171
5.2.1.2	FCS_IPSEC_EXT.1 Ipsec Protocol .....	173
5.2.1.2.1	FCS_IPSEC_EXT.1.1 TSS.....	173
5.2.1.2.2	FCS_IPSEC_EXT.1 TSS (VPNGW).....	174
5.2.1.2.3	FCS_IPSEC_EXT.1.3 TSS.....	174
5.2.1.2.4	FCS_IPSEC_EXT.1.4 TSS.....	175
5.2.1.2.5	FCS_IPSEC_EXT.1.5 TSS.....	175
5.2.1.2.6	FCS_IPSEC_EXT.1.6 TSS.....	176
5.2.1.2.7	FCS_IPSEC_EXT.1.7 TSS.....	176
5.2.1.2.8	FCS_IPSEC_EXT.1.8 TSS.....	177
5.2.1.2.9	FCS_IPSEC_EXT.1.9 TSS.....	178
5.2.1.2.10	FCS_IPSEC_EXT.1.10 TSS.....	179
5.2.1.2.11	FCS_IPSEC_EXT.1.11 TSS.....	180
5.2.1.2.12	FCS_IPSEC_EXT.1.12 TSS.....	180
5.2.1.2.13	FCS_IPSEC_EXT.1.13 TSS.....	181
5.2.1.2.14	FCS_IPSEC_EXT.1.14 TSS.....	182
5.2.1.2.15	FCS_IPSEC_EXT.1.1 AGD .....	182
5.2.1.2.16	FCS_IPSEC_EXT.1 AGD (VPNGW) .....	184



5.2.1.2.17	FCS_IPSEC_EXT.1.3 AGD .....	184
5.2.1.2.18	FCS_IPSEC_EXT.1.4 AGD .....	185
5.2.1.2.19	FCS_IPSEC_EXT.1.5 AGD .....	186
5.2.1.2.20	FCS_IPSEC_EXT.1.6 AGD .....	187
5.2.1.2.21	FCS_IPSEC_EXT.1.7 AGD [TD0800] .....	188
5.2.1.2.22	FCS_IPSEC_EXT.1.8 AGD [TD0800] .....	190
5.2.1.2.23	FCS_IPSEC_EXT.1.11 AGD .....	191
5.2.1.2.24	FCS_IPSEC_EXT.1.13 AGD .....	191
5.2.1.2.25	FCS_IPSEC_EXT.1.14 AGD .....	195
5.2.1.3	FCS_TLSS_EXT.1 Extended: TLS Server Protocol Without Mutual Authentication .....	196
5.2.1.3.1	FCS_TLSS_EXT.1.1 TSS .....	196
5.2.1.3.2	FCS_TLSS_EXT.1.2 TSS .....	197
5.2.1.3.3	FCS_TLSS_EXT.1.3 TSS [TD0635] .....	198
5.2.1.3.4	FCS_TLSS_EXT.1.4 TSS [TD0569] .....	198
5.2.1.3.5	FCS_TLSS_EXT.1.1 AGD .....	201
5.2.1.3.6	FCS_TLSS_EXT.1.2 AGD .....	202
5.2.1.3.7	FCS_TLSS_EXT.1.3 AGD .....	202
5.2.1.3.8	FCS_TLSS_EXT.1.4 AGD [TD0569] .....	203
5.2.2	<i>Identification and Authentication (FIA)</i> .....	203
5.2.2.1	FIA_X509_EXT.1/Rev X.509 Certificate Validation .....	203
5.2.2.1.1	FIA_X509_EXT.1/Rev TSS .....	203
5.2.2.1.2	FIA_X509_EXT.1/Rev AGD .....	204
5.2.2.1.3	FIA_X509_EXT.1/Rev (VPNGW) .....	206
5.2.2.2	FIA_X509_EXT.2 X.509 Certificate Authentication .....	206
5.2.2.2.1	FIA_X509_EXT.2 TSS .....	206
5.2.2.2.2	FIA_X509_EXT.2 AGD .....	207
5.2.2.2.3	FIA_X509_EXT.2 (VPNGW) .....	209
5.2.2.3	FIA_X509_EXT.3 Extended: X509 Certificate Requests .....	209
5.2.2.3.1	FIA_X509_EXT.3 TSS .....	209
5.2.2.3.2	FIA_X509_EXT.3 AGD .....	209
5.2.2.3.3	FIA_X509_EXT.3 (VPNGW) .....	212
5.2.3	<i>Security Management (FMT)</i> .....	212
5.2.3.1	FMT_MOF.1/Services Management of Security Functions Behaviour .....	212
5.2.3.1.1	FMT_MOF.1/Services TSS .....	212
5.2.3.1.2	FMT_MOF.1/Services AGD .....	213
5.2.3.2	FMT_MTD.1/CryptoKeys Management of TSF Data .....	214
5.2.3.2.1	FMT_MTD.1/CryptoKeys TSS .....	214
5.2.3.2.2	FMT_MTD.1/CryptoKeys AGD .....	214
5.2.3.2.3	FMT_MTD.1/CryptoKeys (VPNGW) .....	215
<b>6</b>	<b>SECURITY ASSURANCE REQUIREMENTS</b> .....	<b>216</b>
6.1	ADV: DEVELOPMENT .....	216
6.1.1	<i>Basic Functional Specification (ADV_FSP.1)</i> .....	216
6.1.1.1	(5.2.1.1) Evaluation Activity .....	216
6.1.1.2	(5.2.1.2) Evaluation Activity .....	217
6.1.1.3	(5.2.1.3) Evaluation Activity .....	217
6.2	AGD: GUIDANCE DOCUMENTS .....	218
6.2.1	<i>Operational User Guidance (AGD_OPE.1)</i> .....	218

6.2.1.1	(5.3.1.1) Evaluation Activity .....	218
6.2.1.2	(5.3.1.2) Evaluation Activity .....	218
6.2.1.3	(5.3.1.3) Evaluation Activity .....	219
6.2.1.4	(5.3.1.4) Evaluation Activity .....	220
6.2.1.5	(5.3.1.5) Evaluation Activity <b>[TD0536]</b> .....	220
6.2.2	<i>Preparative Procedures (AGD_PRE.1)</i> .....	222
6.2.2.1	(5.3.2.1) Evaluation Activity .....	222
6.2.2.2	(5.3.2.2) Evaluation Activity .....	223
6.2.2.3	(5.3.2.3) Evaluation Activity .....	224
6.2.2.4	(5.3.2.4) Evaluation Activity .....	225
6.2.2.5	(5.3.2.5) Evaluation Activity .....	225
6.3	AVA: VULNERABILITY ASSESSMENT.....	225
6.3.1	<i>Vulnerability Survey (AVA_VAN.1)</i> .....	225
6.3.1.1	(5.6.1.1) Evaluation Activity (Documentation) <b>[TD0547]</b> .....	225
6.3.1.2	(5.6.1.2) Evaluation Activity .....	226
<b>7</b>	<b>DETAILED TEST CASES (TEST ACTIVITIES) .....</b>	<b>228</b>
7.1	AUDIT .....	228
7.1.1	<i>FAU_GEN.1 Test #1</i> .....	228
7.1.2	<i>FAU_GEN.1 Test #2</i> .....	229
7.1.3	<i>FAU_GEN.2 Test #1</i> .....	229
7.1.4	<i>FAU_GEN.2 Test #2</i> .....	230
7.1.5	<i>FAU_STG_EXT.1 Test #1</i> .....	230
7.1.6	<i>FAU_STG_EXT.1 Test #2 (a)</i> .....	231
7.1.7	<i>FAU_STG_EXT.1 Test #2 (b)</i> .....	231
7.1.8	<i>FAU_STG_EXT.1 Test #2 (c)</i> .....	232
7.1.9	<i>FAU_STG_EXT.1 Test #3</i> .....	233
7.1.10	<i>FAU_STG_EXT.1 Test #4</i> .....	233
7.1.11	<i>FPT_STM_EXT.1 Test #1</i> .....	234
7.1.12	<i>FPT_STM_EXT.1 Test #2</i> .....	234
7.1.13	<i>FPT_STM_EXT.1 Test #3</i> <b>[TD0632]</b> .....	235
7.1.14	<i>FTP_ITC.1 Test #1</i> .....	235
7.1.15	<i>FTP_ITC.1 Test #2</i> .....	236
7.1.16	<i>FTP_ITC.1 Test #3</i> .....	237
7.1.17	<i>FTP_ITC.1 Test #4</i> .....	238
7.2	AUTH .....	240
7.2.1	<i>FCS_HTTPS_EXT.1 Test #1</i> .....	240
7.2.2	<i>FIA_AFL.1 Test #1</i> .....	240
7.2.3	<i>FIA_AFL.1 Test #2a</i> .....	241
7.2.4	<i>FIA_AFL.1 Test #2b</i> .....	242
7.2.5	<i>FIA_PMG_EXT.1 Test #1</i> .....	243
7.2.6	<i>FIA_PMG_EXT.1 Test #2</i> .....	245
7.2.7	<i>FIA_UIA_EXT.1 Test #1</i> .....	246
7.2.8	<i>FIA_UIA_EXT.1 Test #2</i> .....	246
7.2.9	<i>FIA_UIA_EXT.1 Test #3</i> .....	247

7.2.10	FIA_UIA_EXT.1 Test #4.....	248
7.2.11	FIA_UAU_EXT.2 Test #1.....	248
7.2.12	FIA_UAU.7 Test #1.....	249
7.2.13	FMT_MOF.1/ManualUpdate Test #1.....	249
7.2.14	FMT_MOF.1/ManualUpdate Test #2.....	250
7.2.15	FMT_MOF.1/Services Test #1.....	250
7.2.16	FMT_MOF.1/Services Test #2.....	251
7.2.17	FMT_MTD.1/CoreData Test #1.....	251
7.2.18	FMT_MTD.1/CryptoKeys Test #1.....	251
7.2.19	FMT_MTD.1/CryptoKeys Test #2.....	252
7.2.20	FMT_SMF.1 Test #1.....	252
7.2.21	FMT_SMR.2 Test #1.....	253
7.2.22	FTA_SSL.3 Test #1.....	253
7.2.23	FTA_SSL.4 Test #1.....	254
7.2.24	FTA_SSL.4 Test #2.....	255
7.2.25	FTA_SSL_EXT.1 Test #1.....	255
7.2.26	FTA_TAB.1 Test #1.....	256
7.2.27	FTP_TRP.1/Admin Test #1.....	257
7.2.28	FTP_TRP.1/Admin Test #2.....	257
7.3	CRYPTO.....	259
7.3.1	FCS_CKM.1 RSA.....	259
7.3.2	FCS_CKM.1 ECC.....	260
7.3.3	FCS_CKM.1 FFC – FIPS PUB 186-4.....	260
7.3.4	FCS_CKM.1 FFC – “safe-prime” groups <b>[TD0580]</b> .....	262
7.3.5	FCS_CKM.2 RSA.....	262
7.3.6	FCS_CKM.2 SP800-56A - ECC.....	263
7.3.7	FCS_CKM.2 SP800-56A - FFC.....	264
7.3.8	FCS_CKM.2 FCC safe-prime.....	266
7.3.9	FCS_CKM.4.....	267
7.3.10	FCS_COP.1/DataEncryption AES-CBC.....	267
7.3.11	FCS_COP.1/DataEncryption AES-GCM.....	270
7.3.12	FCS_COP.1/DataEncryption AES-CTR.....	271
7.3.13	FCS_COP.1/SigGen ECDSA.....	272
7.3.14	FCS_COP.1/SigGen RSA.....	273
7.3.15	FCS_COP.1/Hash.....	274
7.3.16	FCS_COP.1/KeyedHash.....	275
7.3.17	FCS_RBG_EXT.1.....	275
7.4	TLSS.....	278
7.4.1	FCS_TLSS_EXT.1.1 Test #1.....	278
7.4.2	FCS_TLSS_EXT.1.1 Test #2.....	279
7.4.3	FCS_TLSS_EXT.1.1 Test #3a.....	280
7.4.4	FCS_TLSS_EXT.1.1 Test #3b.....	280
7.4.5	FCS_TLSS_EXT.1.2 Test #1.....	282
7.4.6	FCS_TLSS_EXT.1.3 Test #1a.....	282

7.4.7	FCS_TLSS_EXT.1.3 Test #1b	283
7.4.8	FCS_TLSS_EXT.1.3 Test #2	284
7.4.9	FCS_TLSS_EXT.1.3 Test #3	284
7.4.10	FCS_TLSS_EXT.1.4 Test #1 [TD0569]	284
7.4.11	FCS_TLSS_EXT.1.4 Test #2a [TD0569]	285
7.4.12	FCS_TLSS_EXT.1.4 Test #2b [TD0569]	286
7.4.13	FCS_TLSS_EXT.1.4 Test #3a [TD0556, TD0569]	287
7.4.14	FCS_TLSS_EXT.1.4 Test #3b [TD0569]	288
7.5	UPDATE	290
7.5.1	FPT_TST_EXT.1 Test #1	290
7.5.2	FPT_TUD_EXT.1 Test #1	290
7.5.3	FPT_TUD_EXT.1 Test #2 (a)	292
7.5.4	FPT_TUD_EXT.1 Test #2 (b)	293
7.5.5	FPT_TUD_EXT.1 Test #2 (c)	294
7.5.6	FPT_TUD_EXT.1 Test #3 (a)	295
7.5.7	FPT_TUD_EXT.1 Test #3 (b)	296
7.6	X509-REV	298
7.6.1	FIA_X509_EXT.1.1/Rev Test #1a	298
7.6.2	FIA_X509_EXT.1.1/Rev Test #1b	299
7.6.3	FIA_X509_EXT.1.1/Rev Test #2	300
7.6.4	FIA_X509_EXT.1.1/Rev Test #3	301
7.6.5	FIA_X509_EXT.1.1/Rev Test #4	303
7.6.6	FIA_X509_EXT.1.1/Rev Test #5	304
7.6.7	FIA_X509_EXT.1.1/Rev Test #6	305
7.6.8	FIA_X509_EXT.1.1/Rev Test #7	306
7.6.9	FIA_X509_EXT.1.1/Rev Test #8a [TD0527]	307
7.6.10	FIA_X509_EXT.1.1/Rev Test #8b [TD0527]	307
7.6.11	FIA_X509_EXT.1.1/Rev Test #8c [TD0527]	308
7.6.12	FIA_X509_EXT.1.2/Rev Test #1	309
7.6.13	FIA_X509_EXT.1.2/Rev Test #2	310
7.6.14	FIA_X509_EXT.2 Test #1	311
7.6.15	FIA_X509_EXT.3 Test #1	312
7.6.16	FIA_X509_EXT.3 Test #2	313
7.7	FIREWALL	314
7.7.1	FAU_GEN.1 Test #1	314
7.7.2	FFW_RUL_EXT.1 Test #1	314
7.7.3	FFW_RUL_EXT.1 Test #2	315
7.7.4	FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #1	316
7.7.5	FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #2	320
7.7.6	FFW_RUL_EXT.1.5 Test #1	321
7.7.7	FFW_RUL_EXT.1.5 Test #2	323
7.7.8	FFW_RUL_EXT.1.5 Test #3	324
7.7.9	FFW_RUL_EXT.1.5 Test #4	325
7.7.10	FFW_RUL_EXT.1.5 Test #5	328

7.7.11	FFW_RUL_EXT.1.5 Test #6 .....	328
7.7.12	FFW_RUL_EXT.1.5 Test #7 .....	329
7.7.13	FFW_RUL_EXT.1.5 Test #8 .....	329
7.7.14	FFW_RUL_EXT.1.6 Test #1 .....	329
7.7.15	FFW_RUL_EXT.1.6 Test #2 .....	333
7.7.16	FFW_RUL_EXT.1.7 Test #1 .....	333
7.7.17	FFW_RUL_EXT.1.7 Test #2 .....	334
7.7.18	FFW_RUL_EXT.1.8 Test #1 <b>[TD0545]</b> .....	335
7.7.19	FFW_RUL_EXT.1.8 Test #2 .....	336
7.7.20	FFW_RUL_EXT.1.9 Test #1 .....	337
7.7.21	FFW_RUL_EXT.1.10 Test #1 .....	337
7.7.22	FMT_SMF.1/FFW Test #1 .....	338
7.8	IPS .....	339
7.8.1	FAU_GEN.1/IPS .....	339
7.8.2	FMT_SMF.1/IPS Test #1 .....	339
7.8.3	FMT_SMF.1/IPS Test #2 .....	340
7.8.4	FMT_SMF.1/IPS Test #3 .....	341
7.8.5	IPS_ABD_EXT.1 Test #1 .....	341
7.8.6	IPS_ABD_EXT.1 Test #2 .....	344
7.8.7	IPS_IPB_EXT.1 Test #1 .....	346
7.8.8	IPS_IPB_EXT.1 Test #2 .....	347
7.8.9	IPS_IPB_EXT.1 Test #3 .....	348
7.8.10	IPS_NTA_EXT.1.1 Test #1 .....	349
7.8.11	IPS_NTA_EXT.1.2 Test #1 .....	349
7.8.12	IPS_NTA_EXT.1.3 Test #1 .....	349
7.8.13	IPS_SBD_EXT.1.1 Test #1 <b>[TD0722]</b> .....	350
7.8.14	IPS_SBD_EXT.1.1 Test #2 .....	362
7.8.15	IPS_SBD_EXT.1.2 Test #1 .....	364
7.8.16	IPS_SBD_EXT.1.2 Test #2 .....	367
7.8.17	IPS_SBD_EXT.1.3 Test #1 .....	368
7.8.18	IPS_SBD_EXT.1.4 Test #1 .....	370
7.8.19	IPS_SBD_EXT.1.5 Test #1 .....	372
7.8.20	IPS_SBD_EXT.1.6 Test #1 .....	372
7.9	IPSEC .....	373
7.9.1	FCS_IPSEC_EXT.1.1 Test #1 .....	373
7.9.2	FCS_IPSEC_EXT.1.1 Test #2 .....	374
7.9.3	FCS_IPSEC_EXT.1.2 Test #1 .....	375
7.9.4	FCS_IPSEC_EXT.1.3 Test #1 .....	376
7.9.5	FCS_IPSEC_EXT.1.3 Test #2 .....	377
7.9.6	FCS_IPSEC_EXT.1.4 Test #1 .....	378
7.9.7	FCS_IPSEC_EXT.1.5 Test #1 .....	379
7.9.8	FCS_IPSEC_EXT.1.5 Test #2 .....	379
7.9.9	FCS_IPSEC_EXT.1.6 Test #1 .....	380
7.9.10	FCS_IPSEC_EXT.1.7 Test #1 .....	381

7.9.11	FCS_IPSEC_EXT.1.7 Test #2 [TD0800]	382
7.9.12	FCS_IPSEC_EXT.1.8 Test #1	383
7.9.13	FCS_IPSEC_EXT.1.8 Test #2 [TD0800]	384
7.9.14	FCS_IPSEC_EXT.1.10 Test #1	385
7.9.15	FCS_IPSEC_EXT.1.10 Test #2	385
7.9.16	FCS_IPSEC_EXT.1.11 Test #1	385
7.9.17	FCS_IPSEC_EXT.1.12 Test #1	387
7.9.18	FCS_IPSEC_EXT.1.12 Test #2	387
7.9.19	FCS_IPSEC_EXT.1.12 Test #3	388
7.9.20	FCS_IPSEC_EXT.1.12 Test #4	388
7.9.21	FCS_IPSEC_EXT.1.13 Test #1	389
7.9.22	FCS_IPSEC_EXT.1.14 Test #1	389
7.9.23	FCS_IPSEC_EXT.1.14 Test #2	390
7.9.24	FCS_IPSEC_EXT.1.14 Test #3	391
7.9.25	FCS_IPSEC_EXT.1.14 Test #4	392
7.9.26	FCS_IPSEC_EXT.1.14 Test #5	393
7.9.27	FCS_IPSEC_EXT.1.14 Test #6a	394
7.9.28	FCS_IPSEC_EXT.1.14 Test #6b	395
7.10	VPNGW	396
7.10.1	FCS_COP.1/DATAENCRYPTION TEST #1	396
7.10.2	FCS_IPSEC_EXT.1 TEST #1	396
7.10.3	FIA_X509_EXT.1/REV TEST #1	396
7.10.4	FIA_X509_EXT.2 TEST #1	397
7.10.5	FIA_X509_EXT.3 TEST #1	397
7.10.6	FMT_MTD.1/CRYPTOKEYS TEST #1	397
7.10.7	FPT_TST_EXT.1 TEST #1	397
7.10.8	FPT_TUD_EXT.1 TEST #1	398
7.10.9	FAU_GEN.1/VPN Test #1	398
7.10.10	FCS_CKM.1/IKE Test #1	399
7.10.11	FMT_SMF.1/VPN Test #1	399
7.10.12	FPF_RUL_EXT.1.1 Test #1	399
7.10.13	FPF_RUL_EXT.1.1 Test #2	400
7.10.14	FPF_RUL_EXT.1.2 Test #1	400
7.10.15	FPF_RUL_EXT.1.3 Test #1	400
7.10.16	FPF_RUL_EXT.1.4 Test #1	401
7.10.17	FPF_RUL_EXT.1.4 Test #2	401
7.10.18	FPF_RUL_EXT.1.5 Test #1	402
7.10.19	FPF_RUL_EXT.1.5 Test #2	402
7.10.20	FPF_RUL_EXT.1.6 Test #1	403
7.10.21	FPF_RUL_EXT.1.6 Test #2	404
7.10.22	FPF_RUL_EXT.1.6 Test #3	406
7.10.23	FPF_RUL_EXT.1.6 Test #4	408
7.10.24	FPF_RUL_EXT.1.6 Test #5	409
7.10.25	FPF_RUL_EXT.1.6 Test #6	411

7.10.26	<i>FPF_RUL_EXT.1.6 Test #7</i> .....	413
7.10.27	<i>FPF_RUL_EXT.1.6 Test #8</i> .....	414
7.10.28	<i>FPF_RUL_EXT.1.6 Test #9</i> .....	415
7.10.29	<i>FPF_RUL_EXT.1.6 Test #10</i> .....	416
7.10.30	<i>FPT_FLS.1/SelfTest Test #1</i> .....	417
7.10.31	<i>FPT_TST_EXT.3 Test #1</i> .....	418
7.10.32	<i>FTP_ITC.1/VPN Test #1</i> .....	418
<b>8</b>	<b>CAVP MAPPING</b> .....	<b>419</b>
8.1	TOE MODELS AND CRYPTOGRAPHIC OPERATIONAL ENVIRONMENT.....	419
8.2	OPERATIONAL ENVIRONMENT OF THE ALGORITHM IMPLEMENTATION.....	419
8.3	CERTIFICATE(S) TABLE .....	423
<b>9</b>	<b>CONCLUSION</b> .....	<b>428</b>

## 1 TOE OVERVIEW

The TOE is comprised of the SonicWall SonicOS/X v7.0.1 software running either on purpose built TZ, NSa, NSsp, series hardware appliance platforms and NSv virtual appliances running on purpose built ESXi hardware appliances.

The appliance next generation firewall capabilities include stateful packet inspection. Stateful packet inspection maintains the state of network connections, such as Transmission Control Protocol (TCP) streams and User Datagram Protocol (UDP) communication, traveling across the firewall. The firewall distinguishes between legitimate packets and illegitimate packets for the given network deployment. Only packets adhering to the administrator-configured access rules are permitted to pass through the firewall; all others are rejected.

The appliance capabilities include deep-packet inspection (DPI) used for intrusion prevention and detection. These services employ stream-based analysis wherein traffic traversing the product is parsed and interpreted so that its content might be matched against a set of signatures to determine the acceptability of the traffic. Only traffic adhering to the administrator-configured policies is permitted to pass through the TOE.

The appliances support Virtual Private Network (VPN) functionality, which provides a secure connection between the device and the audit server. The appliances support authentication and protect data from disclosure or modification during transfer.

The appliances are managed through a web based Graphical User Interface (GUI). All management activities may be performed through the web management GUI via a hierarchy of menu buttons. Administrators may configure policies and manage network traffic, users, and system logs. The appliances also have local console access where limited administrative functionality to configure the network, perform system updates, and view logs.



## 2 ASSURANCE ACTIVITIES IDENTIFICATION

The Assurance Activities contained within this document include all those defined within collaborative Protection Profile for Network Devices, Version 2.2e , Intrusion Protection Systems (IPS) Version 1.0, Stateful Traffic Filter Firewalls Version 1.4 + Errata 20200625, Virtual Private Network (VPN) Gateways Version 1.3 based upon the core SFRs and those implemented based on selections within the PPs/Modules.

### 3 TEST EQUIVALENCY JUSTIFICATION

The following equivalency analysis provides a per category analysis of key areas of differentiation for each hardware model to determine the minimum subset to be used in testing. The areas examined will use the areas and analysis description provided in the supporting documentation for the NDcPP. Additionally, a comparison of the data presented in section 3 is provided to identify a testing subset that will exercise each of the differences in TOE models.

#### 3.1 DIFFERENCES BETWEEN MODELS OF THE TOE

##### 3.1.1 PLATFORM/HARDWARE DIFFERENCES

The TOE boundary is inclusive of all hardware required by the TOE. The hardware platforms do not provide any of the TSF functionality. All security functionality is implemented in Platform Independent code which is line-by-line identical across hardware models.

**TZ, NSa 2700, and NSa 3700 devices** – These models have Marvell Processors which have the same ‘Quad core Armv8 Cortex-A72’ micro architecture. Hence, they are equivalent and just one out of the 12 models will be tested.

**NSa 4700, NSa 5700, NSa 6700, NSsp 10700, NSsp 11700, and NSsp 13700 devices** – All of these models have Intel Xeon D-21XX processors that have the same ‘Skylake’ microarchitecture. Hence, they are equivalent and just one out of the 6 models will we tested.

**NSv 270, NSv 470, and NSv 870 virtual devices** - The NSv models run on ESXi 7.0 and ESXi 8.0 on same Dell PowerEdge 640 hardware with Intel Xeon Silver 4208 processor and Cascade Lake microarchitecture. Hence, they are equivalent and just one out of two will be tested.

Result: All TOE platforms are not equivalent.

##### 3.1.2 SOFTWARE/OS DEPENDENCIES:

This category of differences is only applicable if the TOE is installed on an OS outside of the TOE boundary. In this case, all software including the OS is included in SonicWall and within the TOE boundary. There are no specific dependencies on the OS since the TOE will not be installed on different OSs.

Result: All TOE platforms are equivalent.

---

### 3.1.3 DIFFERENCES IN LIBRARIES USED TO PROVIDE TOE FUNCTIONALITY

All software binaries compiled in the TOE software are identical and have the same version numbers. There are no differences between the included libraries. A set of CAVP certificates will be provided for the cryptographic functionality as tested in the TOE’s operational environment.

Table 01 – CAVP Certificate to Operational Environment Mapping

Operational Environment	Certificate
Marvell 88F7040	A5110
Marvell CN9130	
Intel Xeon D-2123IT	A2583
Intel Xeon D-2166NT	
Intel Xeon D-2178NT	
SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)	A4982
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)	

Result: CAVP algorithm testing provides valid coverage for the platforms. The libraries are identical.

---

### 3.1.4 TOE MANAGEMENT INTERFACE DIFFERENCES

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

Result: All TOE platforms are equivalent.

---

### 3.1.5 TOE FUNCTIONAL DIFFERENCES

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v2.2e or NDcPP, collaborative Protection Profile Module for Stateful Traffic Filter Firewall, hereafter referred to as MOD\_FW v1.4e or MOD\_FW, PP-Module for Virtual Private Network (VPN) Gateways Version 1.2 hereafter referred to as MOD\_VPNGW v1.2 or MOD\_VPNGW, PP-Module for Intrusion Protection Systems (IPS) Version 1.0, hereafter referred to as MOD\_IPS v1.0 or MOD\_IPS.

## **Security Audit**

The TOE generates audit records for administrative activity, security related configuration changes, cryptographic key changes and startup and shutdown of the audit functions. The audit events are associated with the administrator who performs them, if applicable. The audit records are transmitted over an IPsec VPN tunnel to an external audit server in the IT environment for storage.

## **Cryptographic Support**

The TOE provides cryptographic functions (key generation, key establishment, key destruction, cryptographic operation) to secure remote administrative sessions over Hypertext Transfer Protocol Secure (HTTPS)/Transport Layer Security (TLS), and to support Internet Protocol Security (IPsec) to provide VPN functionality and to protect the connection to the audit server.

## **Residual Data Protection**

The TOE ensures that data cannot be recovered once deallocated.

## **Identification and Authentication**

The TOE provides a password-based logon mechanism. This mechanism enforces minimum strength requirements and ensures that passwords are obscured when entered. The TOE also validates and authenticates X.509 certificates for all certificate use.

## **Security Management**

The TOE provides management capabilities via a Web-based GUI, accessed over HTTPS. Management functions allow the administrators to configure and update the system, manage users, and configure the Virtual Private Network (VPN) and Intrusion Prevention System (IPS) functionality.

## **Protection of the TSF**

The TOE prevents the reading of plaintext passwords and keys. The TOE provides a reliable timestamp for its own use. To protect the integrity of its security functions, the TOE implements a suite of self-tests at startup and shuts down if a critical failure occurs. The TOE verifies the software image when it is loaded. The TOE ensures that updates to the TOE software can be verified using a digital signature.

## **TOE Access**

The TOE monitors local and remote administrative sessions for inactivity and either locks or terminates the session when a threshold time period is reached. An advisory notice is displayed at the start of each session.

### Trusted Path/Channels

The TSF provides IPsec VPN tunnels for trusted communication between itself and an audit server. The TOE implements HTTPS for protection of communications between itself and the Management Console.

### Intrusion Prevention

The TOE performs analysis of IP-based network traffic and detects violations of administratively defined IPS policies. The TOE inspects each packet header and payload for anomalies and known signature-based attacks and determines whether to allow traffic to traverse the TOE.

### Stateful Traffic Filtering and Packet Filtering

The TOE restricts the flow of network traffic between protected networks and other attached networks based on addresses and ports of the network nodes originating (source) and/or receiving (destination) applicable network traffic, as well as on established connection information.

The TOE performs packet filtering on network packets.

Each hardware model within the TOE boundary provides identical functionality. There is no difference in the way the user interacts with each of the devices or the services that are available to the user in each of these devices. Each device can be run with the same identical version of SonicWall software. For TOE software, differences in the provided functionality are denoted by a different version of the software. If there had been differences in the functionality provided by the software, the actual release version would have been different for the platform.

Result: All TOE platforms are equivalent

## 3.2 EQUIVALENCY CONCLUSIONS

The above analysis shows that the TOE models support 3 different microarchitectures:

1. Quad-core ARMv8 Cortex-A53 microarchitecture on Marvell Processors
2. Skylake microarchitecture on Intel Xeon D-21XX processors
3. Cascade Lake microarchitecture on Intel Xeon Silver 4208 processors

Hence, complete testing on 3 models covering the 3 different microarchitecture is sufficient. All other models listed above are included by equivalency. The following platforms are tested for this evaluation.

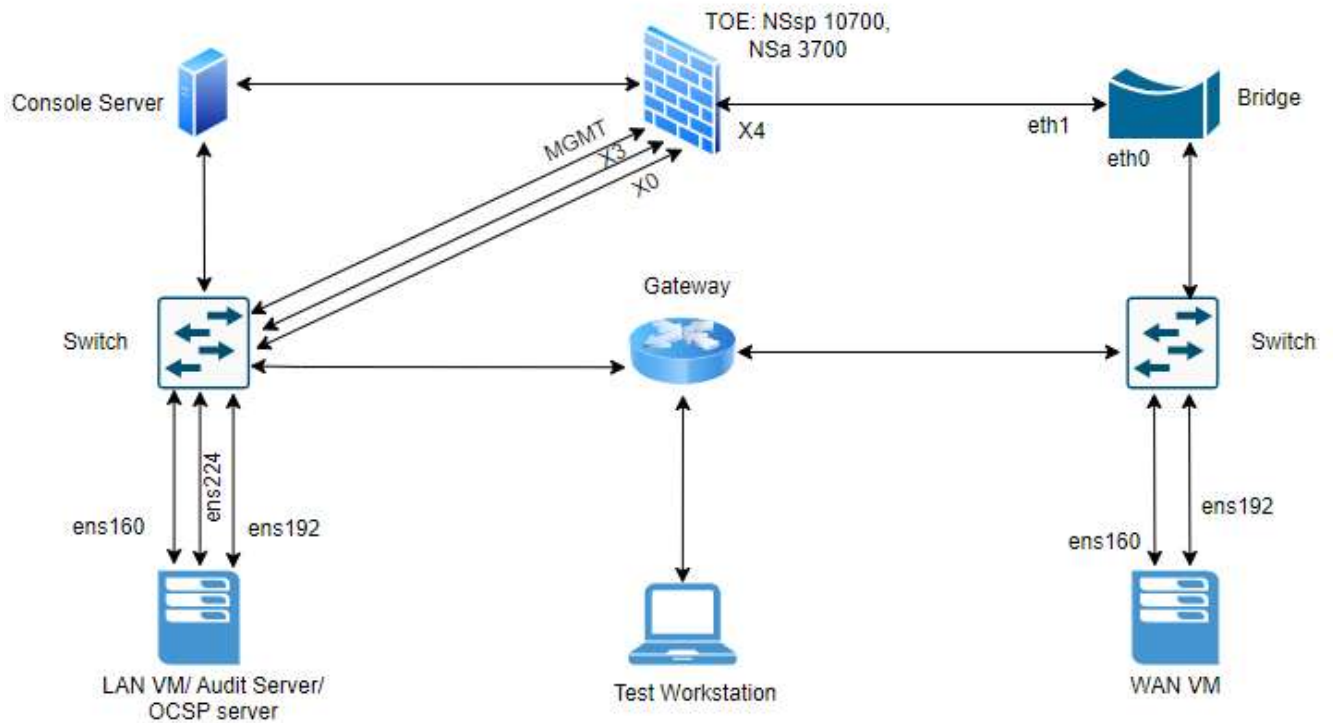
- NSa 3700 running SonicOS/X 7.0.1
- NSsp 10700 running SonicOS/X 7.0.1

- NSv 870 running SonicOS/X 7.0.1 on ESXi 7.0.

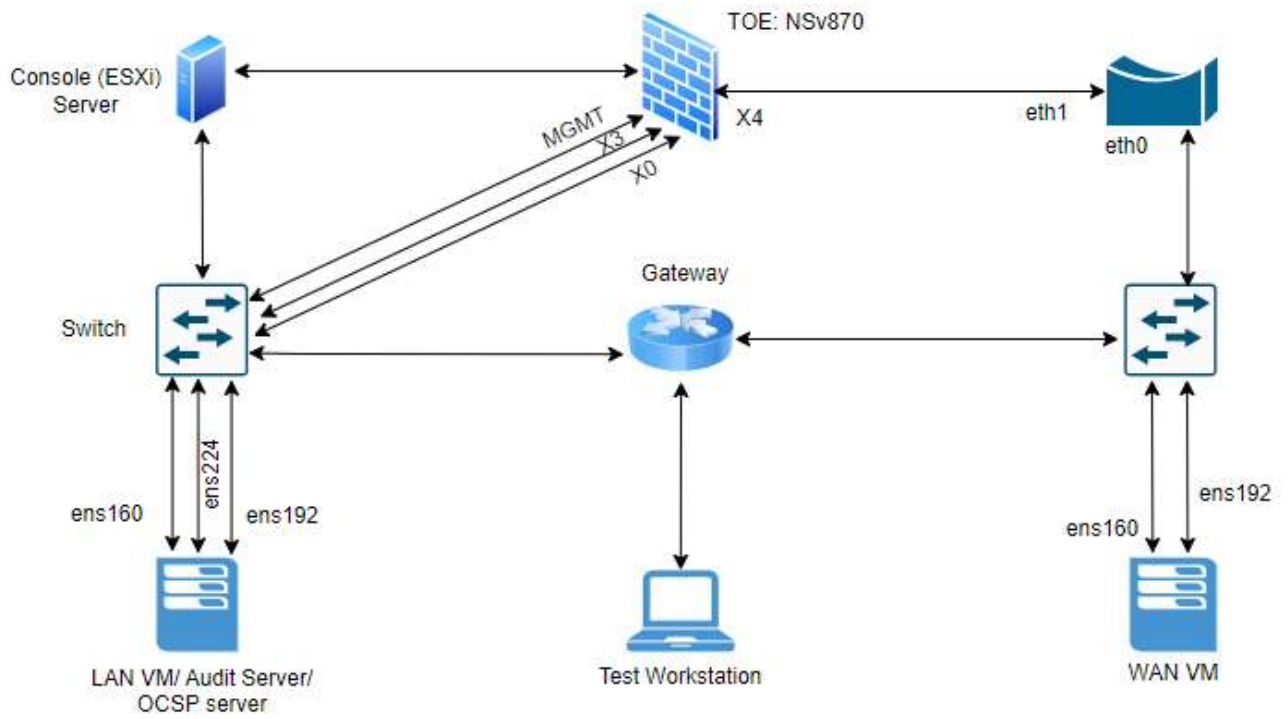
## 4 TEST BED DESCRIPTIONS

### 4.1 AUDIT/FIREWALL/IPS/IPSEC/VPNGW/X509-REV

#### 4.1.1 PHYSICAL TOE



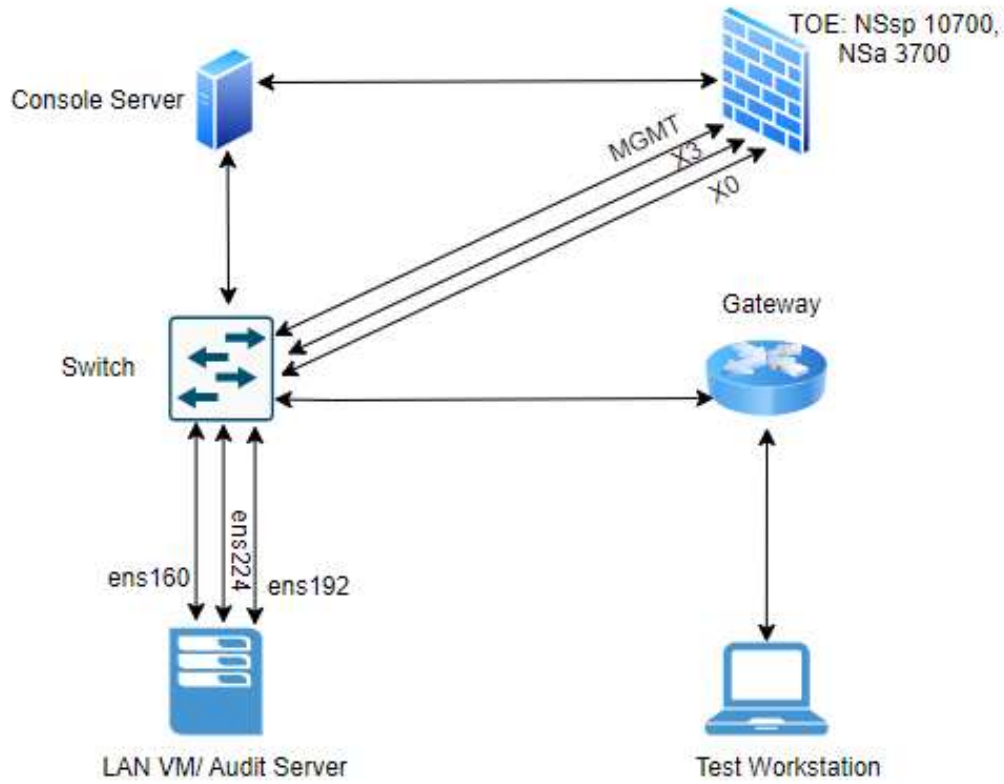
#### 4.1.2 VIRTUAL TOE



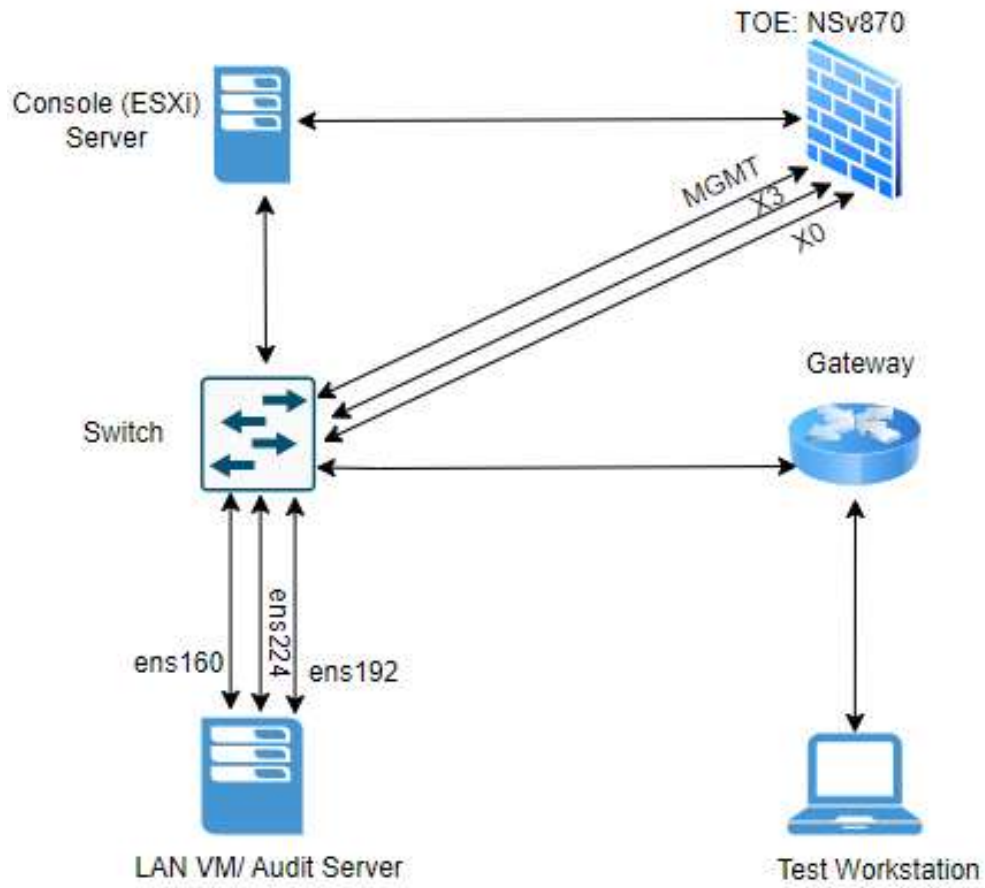


## 4.2 AUTH/CRYPTO/UPDATE/TLSS

### 4.2.1 PHYSICAL TOE



#### 4.2.2 VIRTUAL TOE



## 4.3 CONFIGURATION INFORMATION

### 4.3.1 PHYSICAL TOE

The following table provides configuration information about each device in the test environment.

Device Details Table					
Device Details		Network Details	System Details		
Device Name	Function	Protocols	OS, including version	Timing Source	Software & Tools, including version
NSsp 10700	TOE	HTTPS, TLS IPsec	SonicOS 7.0.1-5163	Manually verified and synced	N/A
NSa 3700	TOE	HTTPS, TLS IPsec	SonicOS 7.0.1-5163	Manually verified and synced	N/A
Console Server	Console Server	SSH	N/A	Manually verified and synced	N/A
Test Workstation	Tester's Laptop	SSH	Windows 10	Manually verified and synced	Wireshark Version 3.6.7, Windows Terminal Version: 1.20.11781.0, Putty v 0.77, Chrome (Version 129.0.6668.60), Microsoft Edge (Version 129.0.2792.65), XCA (v2.7.0)
WAN VM	WAN VM	SSH/IPsec	Ubuntu 22.04.4 LTS	Manually verified and synced	strongSwan (5.9.11), rsyslogd (8.2302.0), OpenSSL (3.0.2), tcpdump (4.99.1), netcat (1.218-4ubuntu1), hping3 (3.0.0-alpha-2), Scapy(2.5.0), filezilla (3.58.0), apache2 (2.4.52), acumen-tls-tool(2.1.0), acumen- firewall-vpn-tool(1.0), x509-mod tool(v1.1)
LAN VM/ Audit Server/OC SP Server	LAN VM/ Audit Server/OCS P Server	SSH/IPsec	Ubuntu 22.04.4 LTS	Manually verified and synced	strongSwan (5.9.11), rsyslogd (8.2302.0), OpenSSL (3.0.2), tcpdump (4.99.1), netcat (1.218-4ubuntu1),

Device Details Table					
Device Details		Network Details	System Details		
Device Name	Function	Protocols	OS, including version	Timing Source	Software & Tools, including version
					hping3 (3.0.0-alpha-2), Scapy(2.5.0), filezilla (3.58.0), apache2 (2.4.52), acumen-tls-tool(2.1.0), acumen- firewall-vpn-tool(1.0), x509-mod tool(v1.1), acumen-tlss tool(v1.1)
Bridge	Bridge, Tcpdump	SSH	Raspbian GNU/Linux 9 (stretch) Linux 4.14.71-v7+	Manually verified and synced	tcpdump (4.99.1)
Switch	Switch	N/A	N/A	Manually verified and synced	N/A
Gateway	Gateway	N/A	N/A	Manually verified and synced	N/A

### 4.3.2 VIRTUAL TOE

The following table provides configuration information about each device in the test environment.

Device Details Table					
Device Details		Network Details	System Details		
Device Name	Function	Protocols	OS, including version	Timing Source	Software & Tools, including version
NSv870	TOE	HTTPS, TLS IPsec	SonicOS 7.0.1-5163	Manually verified and synced	N/A
VMware vSphere ESXi 7.0 on PowerEdge R640	Console Server	HTTPS	ESXi v 7.0	Manually verified and synced	N/A
Test Workstation	Tester's Laptop	SSH	Windows 10	Manually verified and synced	Wireshark Version 3.6.7, Windows Terminal Version: 1.20.11781.0, Putty v 0.77, Chrome (Version 129.0.6668.60), Microsoft Edge (Version 129.0.2792.65), XCA (v2.7.0)
WAN VM	WAN VM	SSH/IPsec	Ubuntu 22.04.4 LTS	Manually verified and synced	strongSwan (5.9.11), rsyslogd (8.2302.0), OpenSSL (3.0.2), tcpdump (4.99.1), netcat (1.218-4ubuntu1), hping3 (3.0.0-alpha-2), Scapy(2.5.0), filezilla (3.58.0), apache2 (2.4.52), acumen-tls-tool (2.1.0), acumen- firewall-vpn-tool(1.0), x509-mod tool(v1.1)
LAN VM/ Audit Server/OC SP Server	LAN VM/ Audit Server/OCS P Server	SSH/IPsec	Ubuntu 22.04.4 LTS	Manually verified and synced	strongSwan (5.9.11), rsyslogd (8.2302.0), OpenSSL (3.0.2), tcpdump (4.99.1), netcat (1.218-4ubuntu1), hping3 (3.0.0-alpha-2),

Device Details Table					
Device Details		Network Details	System Details		
Device Name	Function	Protocols	OS, including version	Timing Source	Software & Tools, including version
					Scapy(2.5.0), filezilla (3.58.0), apache2 (2.4.52), acumen-tls-tool(2.1.0), acumen- firewall-vpn-tool(1.0), x509-mod tool(v1.1), acumen-tlss tool(v1.1)
Bridge	Bridge, Tcpdump	SSH	Raspbian GNU/Linux 9 (stretch) Linux 4.14.71-v7+	Manually verified and synced	tcpdump (4.99.1)
Switch	Switch	N/A	N/A	Manually verified and synced	N/A
Gateway	Gateway	N/A	N/A	Manually verified and synced	N/A

#### 4.4 TEST TIME AND LOCATION

All testing was carried out at Acumen Security office located at 2400 Research Blvd Suite #395, Rockville, MD 20850. Testing occurred from November 2023 to December 2024.

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

## 5 DETAILED TEST CASES (TSS AND AGD ACTIVITIES)

### 5.1 MANDATORY REQUIREMENTS

#### 5.1.1 SECURITY AUDIT (FAU)

##### 5.1.1.1 FAU\_GEN.1 AUDIT DATA GENERATION

###### 5.1.1.1.1 FAU\_GEN.1 TSS

For the administrative task of generating/import of, changing, or deleting of cryptographic keys as defined in FAU\_GEN.1.1c, the TSS should identify what information is logged to identify the relevant key.

###### Evaluator Findings:

The evaluator examined the TSS row **FAU\_GEN.1** and ensured that it identifies what information is logged to identify the relevant cryptographic key during generating/import, changing, or deleting.

The relevant information is found in the following section(s): TOE Summary Specification **FAU\_GEN.1**.

Upon investigation, the evaluator found that the TSS states that: **In the case of key related operations, the name of the certificate the key is associated with is logged and used as the unique reference identifier.**

For distributed TOEs the evaluator shall examine the TSS and ensured that it describes which of the overall required auditable events defined in FAU\_GEN.1.1 are generated and recorded by which TOE components.

###### Evaluator Findings:

The TOE is not a distributed TOE; hence this assurance activity is not applicable.

The evaluator shall ensure that the mapping of audit events to TOE components accounts for, and is consistent with, information provided in Table 1, as well as events in Tables 2, 4, and 5 (as applicable to the overall TOE). The evaluator confirmed that all components defined as generating audit information for a particular SFR contributed to that SFR as defined in the mapping of SFRs to TOE components, and that the audit records generated by each component covered all the SFRs that it implements.

###### Evaluator Findings:

The TOE is not a distributed TOE; hence this assurance activity is not applicable.

###### Verdict:

**PASS.**

###### 5.1.1.1.2 FAU\_GEN.1 AGD

The evaluator shall check the guidance documentation and ensure that it provides an example of each auditable event required by FAU\_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, shall be provided from the actual audit record).

<b>Evaluator Findings:</b>
<p>The evaluator checked the AGD and ensured that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, was provided from the actual audit record).</p> <p>The relevant information is found in the following section(s): <b>Audit Logs</b></p> <p>Upon investigation, the evaluator concluded that the AGD section '<b>Audit Logs</b>' provides an example of each auditable event required by FAU_GEN.1.1.</p>

The evaluator shall also make a determination of the administrative actions related to TSF data related to configuration changes.

<b>Evaluator Findings:</b>
<p>The evaluator made a determination of the administrative actions related to TSF data related to configuration changes.</p> <p>The relevant information is found in the following section(s): <b>Audit Logs</b></p> <p>Upon investigation, the evaluator summarizes that all the administrative actions related to configuration changes and TSF data were adequately provided.</p>

The evaluator shall examine the guidance documentation and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP.

<b>Evaluator Findings:</b>
<p>The evaluator examined the AGD and made a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP.</p> <p>The relevant information is found in the following section(s): <b>Audit logs</b></p> <p>Upon investigation, the evaluator examined the AGD and made a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the</p>



configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP.

<u>Administrative Activity</u>	<u>Method (Command/GUI Configuration)</u>	<u>Section</u>
<b>Start-up and shut-down of the audit functions</b>		
<b>Administrative login and logout</b>	Graphical User Interface/Command Line Interface	Section: 'Initial Setup and Registration Using Local Management' and 'Logging Out'
<b>Generating/import of, changing, or deleting of cryptographic keys</b>	Graphical User Interface	Section: 'Generating a Certificate Singing Request Deleting a Certificate Signing Request' and 'Deleting a Certificate Signing Request'
<b>Resetting passwords</b>	Graphical User Interface	Section: 'Editing Local User'

The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding guidance documentation satisfies the requirements related to it.

**Evaluator Findings:**

The evaluator documented the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding AGD satisfies the requirements related to it.

The relevant information is found in the following section(s): **Audit logs**

**Verdict:**

**PASS.**

## 5.1.1.2 FAU\_GEN.1 AUDIT DATA GENERATION (FFW)

### 5.1.1.2.1 FAU\_GEN.1 TSS

No additional Evaluation Activities are specified.

### 5.1.1.2.2 FAU\_GEN.1 AGD

In addition to the Evaluation Activities specified in the Supporting Document for the Base-PP, the evaluator shall check the guidance documentation to ensure that it describes the audit records specified in Table 2 of the PP-Module in addition to those required by the Base-PP.

#### Evaluator Findings:

The evaluator reviewed the guidance documentation '**All management activities of TSF data (including creation, modification and deletion of firewall rules)**' and '**Application of rules configured with the 'log' operation**' and confirmed that it describes the audit records specified in Table 2 of the PP-Module in addition to those required by the Base-PP.

The relevant information is found in the following section(s): **All management activities of TSF data (including creation, modification and deletion of firewall rules)** and **Application of rules configured with the 'log' operation**

Upon investigation, the evaluator found that the claimed AGD sections describes the audit records specified in Table 2 of the PP-Module in addition to those required by the Base-PP. The AGD states that:

**'All management activities of TSF data (including creation, modification and deletion of firewall rules):**

- **Creation of Firewall rules**

```
2024-01-04T10:41:44.650152+00:00 10.7.1.163 id=firewall sn=2CB8EDA31DC0 time="2024-01-04 10:41:38 UTC" fw=none_1 pri=6 c=16 m=440 msg="Security Policy added" sess="Web" n=12 usr="admin" src=192.168.254.68 note="Allow 'ICMP_T8' from 'LAN X3 Subnet' to 'VPN remote_lan1'" uid="00000000-0000-0010-0700-2cb8eda31dc0" rule="1 (LAN->VPN)" fw_action="NA"
```

- **Modification of Firewall rules**

```
2024-01-04T11:01:26.785923+00:00 10.7.1.163 id=firewall sn=2CB8EDA31DC0 time="2024-01-04 11:01:20 UTC" fw=none_1 pri=6 c=16 m=441 msg="Security Policy modified" sess="Web" n=4 usr="admin" src=192.168.254.68 note="Allow 'ICMP_T3' from 'LAN X3 Subnet' to 'VPN remote_lan1'" uid="00000000-0000-0011-0700-2cb8eda31dc0" rule="2 (LAN->VPN)" fw_action="NA"
```

- **Deletion of Firewall rules**

```
2024-04-02T11:04:49.397893+00:00 10.7.1.163 id=firewall sn=2CB8EDA31DC0 time="2024-04-02 11:06:25 UTC" fw=none_1 pri=7 c=16 m=1382 msg="Configuration succeeded: Deleted 'Policy Action' , test, changed from [test]" oldValue="test" newValue="" usr="admin" src=192.168.228.45:58590 dst=10.1.5.163:443:MGMT auditId=7562 tranId=7924 grpName="Firewall Access Rules" grpIndex="test" uuid="00000000-0000-004f-0700-2cb8eda31dc0" auditTime="UTC 11:06:25 Apr 02 2024" sess="API" userMode="Full"
```

Application of rules configured with the 'log' operation:

```
2024-03-19T14:26:48.437672+00:00 10.7.1.163 id=firewall sn=2CB8EDA31DC0 time="2024-03-19 14:28:12 UTC" fw=none_1 pri=6 c=0 m=1235 msg="Packet allowed: code2 matched policy for non-MGMT traffic" note="policyCheck" n=19042 src=10.1.3.107:50808:X3 dst=10.1.4.116:5655:X4 srcMac=00:50:56:8b:73:b0 dstMac=2c:b8:ed:a3:1d:c3 proto=udp/5655 uuid="00000000-0000-004c-0700-2cb8eda31dc0" rule="1 (LAN->WAN)" fw_action="forward"
```

If the optional SFR FFW\_RUL\_EXT.2 is claimed by the TOE, the evaluator shall also check the guidance documentation to ensure that it describes the relevant audit record specified in Table 3 of the PP-Module.

**Evaluator Findings:**

The optional SFR FFW\_RUL\_EXT.2 is not claimed by the TOE; hence, this activity is not applicable.

**Verdict:**

PASS.

5.1.1.3 FAU\_GEN.1/VPN AUDIT DATA GENERATION (VPN GATEWAY)

5.1.1.3.1 FAU\_GEN.1/VPN TSS

The evaluator shall examine the TSS to verify that it describes the audit mechanisms that the TOE uses to generate audit records for VPN gateway behavior.

**Evaluator Findings:**

The evaluator reviewed the TSS, **FAU\_GEN.1/VPN** row and ensured that it describes the audit mechanisms that the TOE uses to generate audit records for VPN gateway behavior.

The relevant information is found in the following section(s): TOE Summary Specification row **FAU\_GEN.1/VPN**

Upon investigation, the evaluator found that the TSS states that:

**'User activity logs record blocked traffic, blocked websites, VPN activity and other events related to the firewall. Each record contains the date and time, event type, subject identity (when applicable) and outcome of the event.**

**The startup and shutdown of the audit function is tied to the startup and shutdown of the TOE and the TOE generates audit messages for this activity. In addition, when the self-tests are performed, audit logs for successful execution of individual tests are generated in addition to the audit log to indicate that all self-tests have passed. There are overall logs for successful and failure of the self-tests as well. When a self-test fails, the TOE enters into an error state and the local console provides an error message reflecting information about the specific failure to the security administrators.'**

If any audit mechanisms the TSF uses for this are not used to generate audit records for events defined by FAU\_GEN.1 in the Base-PP, the evaluator shall ensure that any VPN gateway-specific audit mechanisms also meet the relevant functional claims from the Base-PP.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that, if any audit mechanisms the TSF uses for this are not used to generate audit records for events defined by FAU\_GEN.1 in the Base-PP, any VPN gateway-specific audit mechanisms also meets the relevant functional claims from the Base-PP.

The relevant information is found in the following section(s): TOE Summary Specification row **FAU\_GEN.1/VPN**

Upon investigation, the evaluator found that the TSS does not mention any implementation about audit mechanisms the TSF uses for this and are not used to generate audit records for events defined by FAU\_GEN.1 in the Base-PP (NDcPP v2.2e). The VPN gateway-specific audit mechanisms also meets the relevant functional claims from the Base-PP (NDcPP v2.2e).

For example, FAU\_STG\_EXT.1 requires all audit records to be transmitted to the OE over a trusted channel. This includes the audit records that are required by FAU\_GEN.1/VPN. Therefore, if the TOE has an audit mechanism that is only used for VPN gateway functionality, the evaluator shall ensure that the VPN gateway related audit records meet this requirement, even if the mechanism used to generate these audit records does not apply to any of the auditable events defined in the Base-PP.

**Evaluator Findings:**

The TOE does not use a separate audit mechanism for VPN gateway functionality; hence, this activity is not applicable.

**Verdict:**

PASS.

**5.1.1.3.2 FAU\_GEN.1/VPN AUDIT DATA GENERATION (VPN GATEWAY) AGD**

The evaluator shall examine the operational guidance to verify that it identifies all security-relevant auditable events claimed in the ST and includes sample records of each event type.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it identifies all security-relevant auditable events claimed in the ST and includes sample records of each event type.

The relevant information is found in the following section(s): **Audit Logs**

Upon investigation, the evaluator summarized that the AGD section '**Audit Logs**' provides an example of each auditable events required by FAU\_GEN.1/VPN.

If the TOE uses multiple audit mechanisms to generate different sets of records, the evaluator shall verify that the operational guidance identifies the audit records that are associated with each of the mechanisms such that the source of each audit record type is clear.

**Evaluator Findings:**

The TOE does not use multiple audit mechanisms; hence, this activity is not applicable.

**Verdict:**

PASS.

**5.1.1.4 SECURITY AUDIT DATA GENERATION FOR IPS REFINEMENT (FAU\_GEN)**

**5.1.1.4.1 FAU\_GEN.1/IPS AUDIT DATA GENERATION (IPS) TSS**

The evaluator shall verify that the TSS describes how the TOE can be configured to log IPS data associated with applicable policies.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it describes how the TOE can be configured to log IPS data associated with applicable policies.

The relevant information is found in the following section(s): TOE Summary Specification  
**FAU\_GEN.1/IPS**

Upon investigation, the evaluator found that the TSS states that:  
**Each IPS event is recorded in the logs as a single event. (i.e. Multiple logs with similar events are never combined to create a more general log entry.) Each log entry is grouped in a log category based on event type..**

The evaluator shall verify that the TSS describes what (similar) IPS event types the TOE will combine into a single audit record along with the conditions (e.g., thresholds and time periods) for so doing. The TSS shall also describe to what extent (if any) that may be configurable.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it describes what (similar) IPS event types the TOE will combine into a single audit record along with the conditions (e.g., thresholds and time periods) for so doing. The TSS also describes to what extent (if any) that may be configurable.

The relevant information is found in the following section(s): TOE Summary Specification  
**FAU\_GEN.1/IPS**

Upon investigation, the evaluator found that the TSS states that:

**Each IPS event is recorded in the logs as a single event. (i.e. Multiple logs with similar events are never combined to create a more general log entry.) Each log entry is grouped in a log category based on event type. Logging can be enabled or disabled per category and event type. Authorized administrators can enable enhanced logging to record configuration changes to IPS functions.**

For IPS\_SBD\_EXT.1, for each field, the evaluator shall verify that the TSS describes how the field is inspected and if logging is not applicable, any other mechanism such as counting that is deployed.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it describes how the field is inspected and if logging is not applicable, any other mechanism such as counting that is deployed.

The relevant information is found in the following section(s): TOE Summary Specification  
**FAU\_GEN.1/IPS**

Upon investigation, the evaluator found that the TSS states that:

**IPS audit records are generated with an ID, category, and priority that are specific to each event type. For example, a single IPS audit record for a TCP flood attack may include the following:**

- **ID = 1366**
- **Category = Attack**
- **Priority = ALERT**

- **Message = TCP-Flooding machine %s blacklisted**

**Verdict:**

**PASS.**

#### 5.1.1.4.2 FAU\_GEN.1/IPS AUDIT DATA GENERATION (IPS) AGD

The evaluator shall verify that the operational guidance describes how to configure the TOE to result in applicable IPS data logging.

##### **Evaluator Findings:**

The evaluator checked the AGD and ensured that it describes how to configure the TOE to result in applicable IPS data logging.

The relevant information is found in the following section(s): **Adding Access Rules** and **App Rules**

Upon investigation, the evaluator found that **point number 21** in the AGD section '**Adding Access Rules**' states how to configure the TOE to result in applicable IPS data logging. The AGD states that: '**21. Go to the Logging tab and enable the Logging option to allow rule logging.**'

Furthermore, the evaluator found that **point number 15** in the AGD section '**App Rules**' states how to configure the TOE to result in applicable IPS data logging. The AGD states that:

'**15. If you want the policy to create a log entry when a match is found, select the  Enable Logging checkbox.**'

The evaluator shall verify that the operational guidance provides instructions for any configuration that may be done in regard to logging similar events (e.g., setting thresholds, defining time windows, etc.).

##### **Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions for any configuration that may be done in regard to logging similar events (e.g., setting thresholds, defining time windows, etc.).

The relevant information is found in the following section(s): **Log Settings and Levels**

Upon investigation, the evaluator found that the AGD section states that:

**The log settings offer different options to reduce the logging intensity and to reduce the logging frequency.**

**To change the log settings go to Device / Log / Settings:**

**Category Column:**

**The Category column of the Log Monitor table has three levels:**

- **Category, first and highest level of the tree structure**
- **Group, the second level**
- **Event, the third level**

**Clicking the small black triangle expands or collapses the category or group contents.**

**Color Column:**

The Color column shows the color with which the event, group, or category is highlighted in the Log Monitor table.

**ID Column:**

The ID column shows the ID number of the event. The ID for a particular message is listed in the SonicOS Combined Log Events Reference Guide.

**Priority Column:**

The Priority column shows the severity or priority of a category, group, or event. For events, a menu is provided that lists the selectable priorities. For categories and groups, the priorities are listed in the dialog when you click the Configure button at the end of the row.

The available priorities are: Alert, Critical, Error, Warning, Notice, Inform, Debug

**Note:** Changing the Event Priority may have serious consequences as the Event Priority for all categories will be changed. Modifying the Event Priority will affect the Syslog output for the tag "pri=" as well as how the event will be treated when performing filtering by priority level. Setting the Event Priority to a level that is lower than the Logging Level will cause those events to be filtered out. Also, as GMS ignores received Syslogs that have a level of Debug, heartbeat messages and reporting messages must have a minimum Event Priority of Inform.

**GUI Column:**

The GUI column shows checkboxes that indicate whether this event is displayed in the Log Monitor. For events, you can show or hide the event by selecting or deselecting the checkbox in the column. For categories and groups, you must use the configure dialog.

**Alert Column:**

The Alert column shows checkboxes that indicate whether an Alert message will be sent for this event, group, or category.

**Syslog Column:**

The Syslog column shows checkboxes that indicate whether the event, group, or category will be sent to a Syslog server.

**Trap Column:** The Trap column shows checkboxes that indicate whether the event or event category for which traps should be sent.

**Email Column:**

The Email column shows checkboxes that indicate whether the log will be emailed to the configured address. For events, these checkboxes are configurable in the column. For categories and groups, Email is configured in the Edit Log Group or Edit Log Category dialogs that appear when you click the Configure button at the end of the row.

**Event Count Column:**

The Event Count column shows the count of events by:

Event level — the value shows the number of times that this event has occurred.

Group level — the value shows the total events that occurred within the group.



Category level — the value shows the total events that occurred within the category.

**Verdict:**

**PASS.**

**5.1.1.5 FAU\_GEN.2 USER IDENTITY ASSOCIATION**

**5.1.1.5.1 TSS & AGD**

The TSS and Guidance Documentation requirements for FAU\_GEN.2 are already covered by the TSS and AGD requirements for FAU\_GEN.1.

**5.1.1.6 FAU\_STG\_EXT.1 PROTECTED AUDIT EVENT STORAGE**

**5.1.1.6.1 FAU\_STG\_EXT.1 TSS**

The evaluator shall examine the TSS to ensure it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.

**Evaluator Findings:**

The evaluator examined the TSS row **FAU\_STG\_EXT.1** and ensured that it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.

The relevant information is found in the following section(s): TOE Summary Specification **FAU\_STG\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE is configured to send audit records to an audit server over an IPsec protected link. The link is established between the TOE and the audit server, and the records are sent over this connection.**

The evaluator shall examine the TSS to ensure it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access.

**Evaluator Findings:**

The evaluator examined the TSS and ensured it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access.

The relevant information is found in the following section(s): TOE Summary Specification **FAU\_STG\_EXT.1**

Upon investigation, the evaluator found that the TSS states that:  
**The maximum number of audit log entries recorded in the database file is limited and this limit for each model is different.**

Model	Maximum Log Entries
TZ270, TZ270W, TZ370, TZ370W, TZ470, TZ470W, TZ570, TZ570W, TZ570P	1000
TZ670	2000
NSa2700, NSa3700	5000
NSa4700	7000
NSa5700, NSa6700	10000
NSsp10700, NSsp11700, NSsp13700	10000
NSv270, NSv470, NSv870	10000

**When the log entries capacity reaches 100%, the oldest 25% of log entries are automatically deleted to free up space for new entries. This setting is non-configurable..**

**Access to view these records is restricted to authorized administrators with the appropriate privilege from the WebGUI. Users who do not have the required privilege are not able to access the audit records. Administrators are not permitted to delete or modify the audit logs.**

The evaluator shall examine the TSS to ensure it describes whether the TOE is a standalone TOE that stores audit data locally or a distributed TOE that stores audit data locally on each TOE component or a distributed TOE that contains TOE components that cannot store audit data locally on themselves but need to transfer audit data to other TOE components that can store audit data locally.

**Evaluator Findings:**

The TOE is not a distributed TOE. The TSS row **FAU\_STG\_EXT.1** states that the local audit data is **stored locally in a database file saved in a specifically reserved area of the system flash.**

The evaluator shall examine the TSS to ensure that for distributed TOEs it contains a list of TOE components that store audit data locally.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

The evaluator shall examine the TSS to ensure that for distributed TOEs that contain components which do not store audit data locally but transmit their generated audit data to other components it contains a mapping between the transmitting and storing TOE components.

**Evaluator Findings:**

The TOE is not a distributed TOE; hence this assurance activity is not applicable.

The evaluator shall examine the TSS to ensure that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE shall also be detailed in the TSS.

**Evaluator Findings:**

The evaluator examined the TSS row **FAU\_STG\_EXT.1** and ensured that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE is detailed in the TSS.

The relevant information is found in the following section(s): TOE Summary Specification  
**FAU\_STG\_EXT.1**

Upon investigation, the evaluator found that the TSS states that: **When the log entries capacity reaches 100%, the oldest 25% of log entries are automatically deleted to free up space for new entries. This setting is non-configurable.**

The 'other actions' is not claimed in the ST.

The evaluator shall examine the TSS to ensure that it details whether the transmission of audit information to an external IT entity can be done in real-time or periodically. In case the TOE does not perform transmission in real-time the evaluator needs to verify that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.

**Evaluator Findings:**

The evaluator examined the TSS row **FAU\_STG\_EXT.1** and ensured that it details whether the transmission of audit information to an external IT entity can be done in real-time, periodically, or both. In the case where the TOE is capable of performing transmission periodically, the evaluator verified that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.

The relevant information is found in the following section(s): TOE Summary Specification  
**FAU\_STG\_EXT.1**

Upon investigation, the evaluator found that the TSS states that: **For the exported audit logs, a separate buffer is maintained. The logs are sent continuously and are removed from the buffer as they are sent. If the connection to the audit server is lost, the logs are stored in a 32-kilobyte rolling log buffer. When the buffer becomes full, the oldest logs are overwritten. The audit logs are sent to the external audit server even when the local audit log database file is full.**

For distributed TOEs the evaluator shall examine the TSS to ensure it describes to which TOE components this SFR applies and how audit data transfer to the external audit server is implemented among the different TOE components (e.g. every TOE components does its own transfer or the data is sent to another TOE component for central transfer of all audit events to the external audit server).

**Evaluator Findings:**

The TOE is not a distributed; TOE hence this assurance activity is not applicable.

For distributed TOEs the evaluator shall examine the TSS to ensure it describes which TOE components are storing audit information locally and which components are buffering audit information and forwarding the information to another TOE component for local storage. For every component the TSS shall describe the behaviour when local storage space or buffer space is exhausted.

**Evaluator Findings:**

The TOE is not a distributed; TOE hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.1.6.2 FAU\_STG\_EXT.1 AGD**

The evaluator shall also examine the guidance documentation to ensure it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.

**Evaluator Findings:**

The evaluator examined the guidance documentation '**Audit Server Configuration, Configuring Syslog Setting and Adding a Syslog Server**' and ensured it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.

The relevant information is found in the following section(s): **Audit Server Configuration, Configuring Syslog Setting and Adding a Syslog Server**

Upon investigation, the evaluator summarizes that the claimed AGD sections describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.

The evaluator shall also examine the guidance documentation to determine that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. For example, when an audit event is generated, is it simultaneously sent to the external server and the local store, or is the local store used as a buffer and "cleared" periodically by sending the data to the audit server.

**Evaluator Findings:**

The evaluator also examined the guidance documentation '**Configuring Syslog Setting**' and determined that it describes the relationship between the local audit data and the audit data that are sent to the audit log server.

The relevant information is found in the following section(s): **Configuring Syslog Setting**

Upon investigation, the evaluator found that the AGD states that:  
**'The appliance can be configured to send audit records to an audit server over an IPsec protected link. The link is established between appliance and the audit server, and the records are sent over this connection. For the exported audit logs, a separate buffer is maintained. The logs are sent continuously and are removed from the buffer as they are sent. If the connection to the audit server is lost, the logs are stored in a 32-kilobyte rolling log buffer. When the buffer becomes full, the oldest logs are overwritten.'**

The evaluator shall also ensure that the guidance documentation describes all possible configuration options for FAU\_STG\_EXT.1.3 and the resulting behaviour of the TOE for each possible configuration. The description of possible configuration options and resulting behaviour shall correspond to those described in the TSS.

#### Evaluator Findings:

The evaluator examined the AGD for all possible configuration options for FAU\_STG\_EXT.1.3 and the resulting behaviour of the TOE for each possible configuration and found that the AGD states that it is non-configurable.

The relevant information is found in the following section(s): **Log rotation and Deletion Policy**

Upon investigation, the evaluator found that the AGD states that:

**When the log capacity reaches 100%, the oldest 25% of log entries are automatically deleted to free up space for new entries. This setting is non-configurable.**

#### Verdict:

PASS.

### 5.1.2 CRYPTOGRAPHIC SUPPORT (FCS)

#### 5.1.2.1 FCS\_CKM.1 CRYPTOGRAPHIC KEY GENERATION

##### 5.1.2.1.1 FCS\_CKM.1 TSS

The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE.

#### Evaluator Findings:

The evaluator ensured that the TSS row **FCS\_CKM.1** identifies the key sizes supported by the TOE.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE supports Rivest-Shamir-Adleman (RSA) using 2048-bits, 3072-bits, and 4096-bits keys, ECDSA using P-256 or P-384 or P-521 keys, and Diffie-Hellman Group 14.**

If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.

#### Evaluator Findings:

The evaluator examined the TSS row **FCS\_CKM.1** and verified that it identifies the usage for each scheme.

The relevant information is found in the following section(s): TOE Summary Specification row **FCS\_CKM.1**.

Upon investigation, the evaluator found that the TSS states that: **EC-Curves and RSA is used in support of TLS and IPsec.**

The TOE performs Elliptic-Curve Diffie-Helman and Diffie-Hellman Group 14 to establish IPsec keys (FCS\_IPSEC\_EXT.1) for secure communications with VPN clients, VPN gateways, and the audit server.

The TOE implements PKCS1\_v1.5 conformant RSA-based key establishment scheme for asymmetric key establishment used in TLS (FCS\_TLSS\_EXT.1) for remote administration.

**Verdict:**

PASS.

5.1.2.1.2 FCS\_CKM.1 AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.

**Evaluator Findings:**

The evaluator verified that the AGD section '**Generating a Certificate Signing Request**' instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.

The relevant information is found in the following section(s): **Generating a Certificate Signing Request**

Upon investigation, the evaluator found that the AGD instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.

**Verdict:**

PASS.

5.1.2.2 FCS\_CKM.1/IKE CRYPTOGRAPHIC KEY GENERATION (FOR IKE PEER AUTHENTICATION)

5.1.2.2.1 FCS\_CKM.1/IKE TSS

The evaluator shall check to ensure that the TSS describes how the key-pairs are generated.

**Evaluator Findings:**

The evaluator reviewed the TSS, section '**FCS\_CKM.1/IKE**' row and ensured that it describes how the key-pairs are generated.

The relevant information is found in the following section(s): TOE Summary Specification row **FCS\_CKM.1/IKE**

Upon investigation, the evaluator found that the TSS states that:

**'The TOE supports Rivest-Shamir-Adleman (RSA) using 2048-bits, 3072-bits and 4096-bit keys, ECDSA using P-256 or P-384 or P-521 keys, and Diffie-Hellman Group 14.**

**RSA and ECDSA keys are generated in accordance with FIPS PUB 186-4 as described in FCS\_COP.1/SigGen.**

**Diffie-Hellman Group 14 keys are generated using the parameters specified in RFC 3526 Section 3.**

In order to show that the TSF implementation complies with FIPS PUB 186-4, the evaluator shall ensure that the TSS contains the following information:

- The TSS shall list all sections of Appendix B to which the TOE complies
- For each applicable section listed in the TSS, for all statements that are not "shall" (that is, "shall not," "should," and "should not"), if the TOE implements such options it shall be described in the TSS. If the included functionality is indicated as "shall not" or "should not" in the standard, the TSS shall provide a rationale for why this will not adversely affect the security policy implemented by the TOE
- For each applicable section of Appendix B, any omission of functionality related to "shall" or "should" statements shall be described

#### **Evaluator Findings:**

The evaluator reviewed the TSS, section **FCS\_CKM.1/IKE**, and ensured that it contains the following information:

**The TOE complies with the requirements in FIPS PUB 186-4, Appendix B.3 for RSA and FIPS PUB 186-4, Appendix B.4 for ECDSA.**

**Diffie-Hellman Group 14 keys are generated using the parameters specified in RFC 3526 Section 3.**

In addition, the TOE complies with all 'shall' and 'should' statements and does not implement any 'should not' or 'shall not' statements. Those implementations of 'should' statements are clarified.

No omission of functionality has been claimed. No alternative implementations are claimed.

Any TOE-specific extensions, processing that is not included in the Appendices, or alternative implementations allowed by the Appendices that may impact the security requirements the TOE is to enforce shall be described.



**Evaluator Findings:**

The evaluator reviewed the TSS, section **FCS\_CKM.1/IKE'** row and verified that any TOE-specific extensions, processing that is not included in the Appendices, or alternative implementations allowed by the Appendices that may impact the security requirements the TOE are described.

The relevant information is found in the following section(s): TOE Summary Specification row **FCS\_CKM.1/IKE**

Upon investigation, the evaluator found that the TSS does not state any TOE-specific extensions, processing that is not included in the Appendices, or alternative implementations allowed by the Appendices that may impact the security requirements the TOE are described. The TSS states that: **'The TOE complies with the requirements in FIPS PUB 186-4, Appendix B as described in FCS\_COP.1/SigGen.'**

**Verdict:**

**PASS.**

**5.1.2.2.2 FCS\_CKM.1/IKE AGD**

The evaluator shall check that the operational guidance describes how the key generation functionality is invoked, and describes the inputs and outputs associated with the process for each signature scheme supported.

**Evaluator Findings:**

The evaluator checked the AGD section **'Generating a Certificate Signing Request'** and ensured that it describes how the key generation functionality is invoked, and the inputs and outputs associated with the process for each supported signature scheme.

The relevant information is found in the following section(s): **Generating a Certificate Signing Request**

Upon investigation, the evaluator found that the claimed AGD section gives description about key generation functionality invocation and the input and outputs for each supported signature scheme. The **point numbers 7** and **8** in the claimed AGD section states that:

- 7. Select a subject key type from the Subject Key Type drop-down menu:**

<b>RSA (default)</b>	<b>A public key cryptographic algorithm used for encrypting data.</b>
<b>ECDSA</b>	<b>Encrypts data using the Elliptic Curve Digital Signature Algorithm, which has a high strength-per-key-bit security.</b>
- 8. Select a subject key size or curve from the Subject Key Size/Curve drop-down menu.**

Not all key sizes or curves are supported by a Certificate Authority, therefore, you should check with your CA for supported key sizes.

If you selected a key type of:

RSA, select a key size	ECDSA, select a curve
2048 bits (default)	prime256v1: X9.62.SECP curve over a 256 bit prime field (default)
3072 bits	secp384r1: NIST/SECP curve over a 384 bit prime field
4096 bits	secp521r1: NIST/SECP curve over a 521 bit prime field

The evaluator shall also check that the operational guidance is provided regarding the format and location of the output of the key generation process.

#### Evaluator Findings:

The evaluator checked the AGD section '**Generating a Certificate Signing Request**' and ensured that it provides the format and location of the output of the key generation process.

The relevant information is found in the following section(s): **Generating a Certificate Signing Request**

Upon investigation, the evaluator found that the claimed AGD section ensured that it provides the format and location of the output of the key generation process. The claimed AGD section states that: '**Private keys and public key certificates are stored encrypted in flash memory in PEM (.pem) or DER (.der or .cer) encoded format.**

Further, the **step 9** states that:

**When the Certificate Signing Request is generated, a message describing the result is displayed and a new entry appears in the Certificates table with the type Pending request.'**

#### Verdict:

PASS.

### 5.1.2.3 FCS\_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT

#### 5.1.2.3.1 FCS\_CKM.2 TSS [TD0580]

The evaluator shall ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS\_CKM.1.1.

**Evaluator Findings:**

The evaluator ensured that the supported key establishment schemes correspond to the key generation schemes identified in FCS\_CKM.1.1.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.2**.

Upon investigation, the evaluator found that: **The TOE supports Rivest-Shamir-Adleman (RSA) using 2048-bits, 3072-bits and 4096-bits keys, ECDSA using P-256 or P-384 or P-521 keys, and Diffie-Hellman Group 14.**

If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme. It is sufficient to provide the scheme, SFR, and service in the TSS.

The intent of this activity is to be able to identify the scheme being used by each service. This would mean, for example, one way to document scheme usage could be as shown in the table.

Scheme	SFR	Service
RSA	FCS_TLSS_EXT.1	Administration
ECDH	FCS_SSHC_EXT.1	Audit Server
Diffie-Hellman (Group 14)	FCS_SSHC_EXT.1	Backup Server
ECDH	FCS_IPSEC_EXT.1	Authentication Server

The information provided in the example above does not necessarily have to be included as a table but can be presented in other ways as long as the necessary data is available.

**Evaluator Findings:**

The evaluator examined the TSS row **FCS\_CKM.2** to verify that it identifies the usage for each scheme. It is sufficient to provide the scheme, SFR, and service in the TSS.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.2**.

Upon investigation, the evaluator found that the TSS states that: **EC-Curves and RSA is used in support of TLS and IPsec.**

**The TOE performs Elliptic-Curve Diffie-Helman and Diffie-Hellman Group 14 to establish IPsec keys (FCS\_IPSEC\_EXT.1) for secure communications with VPN clients, VPN gateways, and the audit server.**

**The TOE implements PKCS1\_v1.5 conformant RSA-based key establishment scheme and NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" conformant EC-based key establishment scheme for asymmetric key establishment used in TLS (FCS\_TLSS\_EXT.1) for remote administration.**

**Verdict:**

**PASS.**

#### 5.1.2.3.2 FCS\_CKM.2 AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s).

##### Evaluator Findings:

The evaluator verified that the AGD section '**Generating a Certificate Signing Request**' instructs the administrator how to configure the TOE to use the selected key establishment scheme(s).

The relevant information is found in the following section(s): **Generating a Certificate Signing Request**

Upon investigation, the evaluator found that the AGD instructs the administrator how to configure the TOE to use the selected key establishment scheme(s) for all cryptographic protocols defined in the Security Target.

##### Verdict:

PASS.

#### 5.1.2.4 FCS\_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION

##### 5.1.2.4.1 FCS\_CKM.4 TSS

The evaluator examines the TSS to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case. For the purpose of this Evaluation Activity the relevant keys are those keys that are relied upon to support any of the SFRs in the Security Target. The evaluator confirms that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT\_APW.EXT.1 and FPT\_SKP\_EXT.1, are accounted for<sup>2</sup>). In particular, if a TOE claims not to store plaintext keys in non-volatile memory then the evaluator checks that this is consistent with the operation of the TOE.

##### Evaluator Findings:

The evaluator examined the TSS row **FCS\_CKM.4** and section **Cryptographic Key Destruction** to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.4** and section **Cryptographic Key Destruction**.

Upon investigation, the evaluator found that the below table contains a column dedicated to the type of the key, the storage of the key, and method of zeroization.

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
RSA private key used for TLS	RSA (2048 bits, 3072 bits, 4096 bits)	Stored in flash memory  Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted  The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
RSA public key used for TLS	RSA (2048 bits, 3072 bits, 4096 bits)	Stored in flash memory  Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted  The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
AES key used for TLS	AES-128 AES-192 AES-256	Keys are not stored  Held in the RAM buffer in plaintext	The key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
Key Agreement Keys used for IPsec	DH (2048 bits)  ECDH (P-256, P-384, P-521)	Keys are not stored  Held in the RAM buffer in plaintext	The key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
Authentication Keys used for IPsec	RSA (2048 bits)  ECDSA (P-256, P-384, P-521)	Stored in flash memory  Held in the RAM buffer in plaintext	The key is overwritten with a block erase when deleted  The plaintext key is overwritten with a pseudo-random pattern upon termination of the

			session or reboot of the appliance
AES Keys used for IPsec	AES-128 AES-256	Keys are not stored  Held in the RAM buffer in plaintext	The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance
SonicWall  Public Key used to verify firmware updates	ECDSA (P-256)	Stored in Flash Memory	The key may be overwritten by a software update
Locally stored passwords	AES-256 in configuration file.	Encryption key is Hardcoded in Flash Memory	Random IV is generated every time the configuration file is updated or imported or after a reboot.

**The keys are generated by the TOE.**

**The SonicWall key used to verify firmware updates supports ECDSA (P-256 NIST curve).**

**The TOE includes two types of memory: RAM and flash. Ephemeral keys are only held in RAM, either in the System RAM or the RAM buffer. The RAM buffer is an area of the System RAM that is allocated for data storage for a period of time. Private keys are only held in plaintext in the RAM buffer. Private keys and public key certificates are stored encrypted in flash memory using OpenSSL 1.0.1. Private and public keys are overwritten in the RAM buffer after use.**

**In the configuration file, only the sensitive data (password) is protected by using AES 256 hash. The encrypt key of this is hardcoded and saved in the flash, and the initialization vector generated randomly. Whenever the configuration file is updated, the initialization vector is also updated. When the configuration file is imported from outside, the TOE generates a new initialization vector. When the TOE is rebooted, the initialization vector is refreshed.**

**Setting the TOE to factory default zeroizes all keys, including the configuration file encrypting key and the keys stored in the flash memory.**

The evaluator shall check to ensure the TSS identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs).

**Evaluator Findings:**

The evaluator confirmed that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT\_APW.EXT.1 and FPT\_SKP\_EXT.1, are accounted for). In particular, the evaluator checked that the claim not to store plaintext keys in non-volatile memory is consistent with the operation of the TOE.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.4**.

Upon investigation, the evaluator found that the TSS states that: **Plaintext key materials held in volatile and non-volatile memory are zeroized after use by direct overwrite consisting of a pseudo-random pattern. The overwrites are read and verified.**

Note that where selections involve ‘destruction of reference’ (for volatile memory) or ‘invocation of an interface’ (for non-volatile memory) then the relevant interface definition is examined by the evaluator to ensure that the interface supports the selection(s) and description in the TSS. In the case of non-volatile memory, the evaluator includes in their examination the relevant interface description for each media type on which plaintext keys are stored. The presence of OS-level and storage device-level swap and cache files is not examined in the current version of the Evaluation Activity.

**Evaluator Findings:**

The evaluator checked to ensure the TSS row **FCS\_CKM.4** identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs).

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.4 & in Table 34 – Cryptographic Key Destruction**

Upon investigation, the evaluator found that the table states that: **The plaintext key is overwritten with a pseudo-random pattern upon termination of the session or reboot of the appliance.**

Where the TSS identifies keys that are stored in a non-plaintext form, the evaluator shall check that the TSS identifies the encryption method and the key-encrypting-key used, and that the key-encrypting-key is either itself stored in an encrypted form or that it is destroyed by a method included under FCS\_CKM.4.

**Evaluator Findings:**

Where the TSS identifies keys that are stored in a non-plaintext form, the evaluator checked that the TSS row **FCS\_CKM.4** identifies the encryption method and the key-encrypting-key used, and that the key-encrypting-key itself is stored in an encrypted form.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.4** and **Table 34 – Cryptographic Key Destruction**.

Upon investigation, the evaluator found that the TSS states that:

**In the configuration file, only the sensitive data (password) is protected by using AES 256 hash. The encrypt key of this is hardcoded and saved in the flash memory. The initialization vector generated randomly to ensure randomness of the first block to ensure the protection of the key. Whenever the configuration file is updated, the initialization vector is also updated. When the configuration file is imported from outside, the TOE generates a new initialization vector. When the TOE is rebooted, the initialization vector is refreshed.**

The evaluator shall check that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below). Note that reference may be made to the Guidance Documentation for description of the detail of such cases where destruction may be prevented or delayed.

**Evaluator Findings:**

The evaluator checked that the TSS row **FCS\_CKM.4** identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below).

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_CKM.4**.

Upon investigation, the evaluator found that the TOE does not have any circumstances that may not conform to key destruction requirements.

Where the ST specifies the use of “a value that does not contain any CSP” to overwrite keys, the evaluator examines the TSS to ensure that it describes how that pattern is obtained and used, and that this justifies the claim that the pattern does not contain any CSPs.

**Evaluator Findings:**

The ST does not specify the use of “a value that does not contain any CSP” to overwrite keys.

**Verdict:**

**PASS.**

**5.1.2.4.2 FCS\_CKM.4 AGD**

A TOE may be subject to situations that could prevent or delay key destruction in some cases. The evaluator shall check that the guidance documentation identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used).



**Evaluator Findings:**

The evaluator checked that the guidance documentation '**Cryptographic Key Destruction**' identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used).

The relevant information is found in the following section(s): **Cryptographic Key Destruction**

Upon investigation, the evaluator found that the AGD states that: **The deletion of keys is a straight-forward process and should not result in any delays. Setting the appliance to factory default zeroizes all keys, including those stored in the flash memory.**

The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer.

For example, when the TOE does not have full access to the physical memory, it is possible that the storage may be implementing wear-levelling and garbage collection. This may result in additional copies of the key that are logically inaccessible but persist physically. Where available, the TOE might then describe use of the TRIM command<sup>3</sup> and garbage collection to destroy these persistent copies upon their deletion (this would be explained in TSS and Operational Guidance).

**Evaluator Findings:**

The evaluator checked that the guidance documentation '**Cryptographic Key Destruction**' provides guidance on situations where key destruction may be delayed at the physical layer.

The relevant information is found in the following section(s): **Cryptographic Key Destruction**

Upon investigation, the evaluator found that the TSF performs all the key destruction mechanisms as specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs). The evaluator reviewed the AGD documentation section '**Cryptographic Key Destruction**' for the TOE and found no situation that would prevent or delay key destruction.

**Verdict:**

**PASS.**

**5.1.2.5 FCS\_COP.1/DATAENCRYPTION CRYPTOGRAPHIC OPERATION (AES DATA ENCRYPTION/DECRYPTION)**

**5.1.2.5.1 FCS\_COP.1/DATAENCRYPTION TSS**

The evaluator shall examine the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption.

**Evaluator Findings:**

The evaluator examined the TSS row **FCS\_COP.1/DataEncryption** to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_COP.1/DataEncryption**.

Upon investigation, the evaluator found that the TSS states that: **The TOE provides AES encryption/decryption in CBC mode with 128-bit, 192-bit, and 256-bit keys and in GCM mode with 128-bit and 256-bit keys.**

**Verdict:**

**PASS.**

**5.1.2.5.2 FCS\_COP.1/DATAENCRYPTION AGD**

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption.

**Evaluator Findings:**

The evaluator verified that the AGD section **‘Enabling NDCPP Compliance’** instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption.

The relevant information is found in the following section(s): **Enabling NDCPP Compliance**

Upon investigation, the evaluator found that the AGD states that: **Once NDPP compliance is enabled, the following settings will be applied by default without any additional configuration changes.**

- **Appliance provides AES encryption/decryption in CBC mode with 128-bit, 192-bit, and 256-bit keys and in GCM mode with 128-bit and 256-bit keys.**

**Verdict:**

**PASS.**

**5.1.2.6 FCS\_COP.1/ DATAENCRYPTION CRYPTOGRAPHIC OPERATION (AES DATA ENCRYPTION/DECRYPTION) (VPNGW)**

**5.1.2.6.1 FCS\_COP.1/DATAENCRYPTION**

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to require the ST author to make certain selections, but these

selections are all part of the original definition of the SFR so no new behavior is defined by the PP-Module.

#### 5.1.2.7 FCS\_COP.1/SIGGEN CRYPTOGRAPHIC OPERATION (SIGNATURE GENERATION AND VERIFICATION)

##### 5.1.2.7.1 FCS\_COP.1/SIGGEN TSS

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

###### Evaluator Findings:

The evaluator examined the TSS row **FCS\_COP.1/SigGen** to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_COP.1/SigGen**.

Upon investigation, the evaluator found that the TSS states that: **The TOE supports signature generation and verification for RSA (4096 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**

###### Verdict:

**PASS.**

##### 5.1.2.7.2 FCS\_COP.1/SIGGEN AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services.

###### Evaluator Findings:

The evaluator verified that the AGD section '**Enabling NDCPP Compliance**' instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services.

The relevant information is found in the following section(s): **Enabling NDCPP Compliance**

Upon investigation, the evaluator found that the AGD states that: **Once NDPP compliance is enabled, the following settings will be applied by default without any additional configuration changes.**

- **Appliance supports signature generation and verification for RSA (2048 bits, 3072 bits, or 4096 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**
- **RSA and ECDSA are used in IKE authentication.**

- RSA is called to verify signatures on firmware uploads for the NSsp 15700 or the NSv Series. For the NSsp 15700 or the NSv Series the NDPP mode does not need to be enabled. It is enabled by default.
- ECDSA is used to verify the signature on firmware updates for the TZ Series, NSa Series firewalls as well as the NSsp 10700, NSsp 11700, and NSsp 13700.

**Verdict:**

**PASS.**

#### 5.1.2.8 FCS\_COP.1/HASH CRYPTOGRAPHIC OPERATION (HASH ALGORITHM)

##### 5.1.2.8.1 FCS\_COP.1/HASH TSS

The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS.

#### **Evaluator Findings:**

The evaluator checked that the association of the hash function with other TSF cryptographic functions is documented in the TSS row **FCS\_COP.1/Hash**.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_COP.1/Hash**.

Upon investigation, the evaluator found that the TSS states that: **The TOE provides cryptographic hashing services for key generation using SHA-256 as specified in NIST SP 800-90 DRBG. SHA-1 and SHA-256 are used in support of TLS. SHA-256, SHA-384, and SHA-512 are used in support of IPsec. SHA-256 is used with ECDSA for the verification of firmware.**

**Verdict:**

**PASS.**

##### 5.1.2.8.2 FCS\_COP.1/HASH AGD

The evaluator checks the AGD documents to determine that any configuration that is required to configure the required hash sizes is present.

#### Evaluator Findings:

The evaluator checked the AGD section 'Enabling NDCPP Compliance' to determine that any configuration that is required to configure the required hash sizes is present.

The relevant information is found in the following section(s): **Enabling NDCPP Compliance**

Upon investigation, the evaluator found that the AGD states that: **Once NDPP compliance is enabled, the following settings will be applied by default without any additional configuration changes.**

- It provides cryptographic hashing services for key generation using SHA-256 as specified in NIST SP 800-90 DRBG.
- SHA-1 and SHA-256 are used in support of TLS.
- SHA-256, SHA-384, and SHA-512 are used in support of IPsec.
- SHA-256 is used with ECDSA for the verification of firmware.
- NSsp 15700 or the NSv Series uses SHA256+RSA2048. These options are enabled by default.

#### Verdict:

PASS.

#### 5.1.2.9 FCS\_COP.1/KEYEDHASH CRYPTOGRAPHIC OPERATION (KEYED HASH ALGORITHM)

##### 5.1.2.9.1 FCS\_COP.1/KEYEDHASH TSS

The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.

#### Evaluator Findings:

The evaluator examined the TSS row **FCS\_COP.1/KeyedHash** to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_COP.1/KeyedHash**.

Upon investigation, the evaluator found that the TSS states that: **The TOE implements HMAC message authentication.**

**HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 are supported with cryptographic key sizes of 160, 256, 384, and 512 bits and message digest sizes of 160, 256, 384, and 512 bits.**

**HMAC-SHA-1 and HMAC-SHA-256 use a block size of 512-bits. HMAC-SHA-384 and HMAC-SHA-512 use a block size of 1024 bits.**

**Verdict:**

PASS.

5.1.2.9.2 FCS\_COP.1/KEYEDHASH AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.

Evaluator Findings:
<p>The evaluator verified that the AGD section ‘<b>Enabling NDCPP Compliance</b>’ instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.</p> <p>The relevant information is found in the following section(s): <b>Enabling NDCPP Compliance</b></p> <p>Upon investigation, the evaluator found that the AGD states that: <b>Once NDPP compliance is enabled, the following settings will be applied by default without any additional configuration changes.</b></p> <ul style="list-style-type: none"><li>• <b>Appliance implements HMAC message authentication. HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 are supported with cryptographic key sizes of 160, 256, 384, and 512 bits and message digest sizes of 160, 256, 384, and 512 bits.</b></li></ul>

**Verdict:**

PASS.

5.1.2.10 FCS\_RBG\_EXT.1 EXTENDED: CRYPTOGRAPHIC OPERATION (RANDOM BIT GENERATION)

5.1.2.10.1 FCS\_RBG\_EXT.1 TSS

The evaluator shall examine the TSS to determine that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min-entropy contained in the combined seed value.

Evaluator Findings:
<p>The evaluator examined the TSS row <b>FCS_RBG_EXT.1</b> and determined that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min- entropy contained in the combined seed value.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_RBG_EXT.1</b>.</p>

Upon investigation, the evaluator found that the TSS states that: **The TOE implements a DRBG in accordance with ISO/IEC 18031:2011 using Hash\_DRBG.**

**The DRBG is seeded using 880-bits from a third-party entropy source provided by the Cavium Octeon hardware on the hardware appliances.**

**The third-party entropy source is assumed to have at least .5 bits of entropy per byte, so the DRBG is seeded with at least 256 bits of entropy.**

**Verdict:**

**PASS.**

### 5.1.2.10.2 FCS\_RBG\_EXT.1 AGD

The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.

**Evaluator Findings:**

The evaluator confirmed that the guidance documentation ‘**Enabling NDCPP Compliance**’ contains appropriate instructions for configuring the RNG functionality.

The relevant information is found in the following section(s): **Enabling NDCPP Compliance**

Upon investigation, the evaluator found that the AGD states that **Once NDPP compliance is enabled, the following settings will be applied by default without any additional configuration changes.**

- **The appliance implements a DRBG in accordance with ISO/IEC 18031:2011 using Hash\_DRBG.**

**Verdict:**

**PASS.**

## 5.1.3 USER DATA PROTECTION (FDP)

### 5.1.3.1 FDP\_RIP.2 FULL RESIDUAL INFORMATION PROTECTION

#### 5.1.3.1.1 FDP\_RIP.2 TSS

“Resources” in the context of this requirement are network packets being sent through (as opposed to “to”, as is the case when a security administrator connects to the TOE) the TOE. The concern is that once a network packet is sent, the buffer or memory area used by the packet still contains data from that packet, and that if that buffer is re-used, those data might remain and make their way into a new

packet. The evaluator shall check to ensure that the TSS describes packet processing to the extent that they can determine that no data will be reused when processing network packets.

The evaluator shall ensure that this description at a minimum describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes packet processing to the extent that they can determine that no data will be reused when processing network packets and that this description describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs.

The relevant information is found in the following section(s): TOE Summary Specification row **FDP\_RIP.2**

Upon investigation, the evaluator found that the TSS states that:

**The TOE ensures that no data is reused with processing network packets.**

**Once packets have been sent from the TOE, the memory buffers are allocated to the buffer pool.**

**When memory is returned to the buffer pool, the memory is overwritten with pseudo random data.**

**The cleared memory can then be reallocated in support of the next request.**

**Verdict:**

**PASS.**

---

**5.1.4 IDENTIFICATION AND AUTHENTICATION (FIA)**

---

**5.1.4.1 FIA\_AFL.1 AUTHENTICATION FAILURE MANAGEMENT**

---

**5.1.4.1.1 FIA\_AFL.1 TSS**

The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.

**Evaluator Findings:**

The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_AFL.1**.



Upon investigation, the evaluator found that the TSS states that: **The SonicWall appliance can be configured to lock out an administrator on the remote administration interface if incorrect login credentials are provided.**

**This is configured using the Enable Administrator/User Lockout features. The number of failed attempts per minute before lockout can be set. The Lockout period, which is the time that must elapse before the user is allowed to attempt to login again, can also be set.**

**If a user enters the configured number of incorrect login credentials, the user is blocked from submitting additional credentials until the lockout period has expired.**

The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking).

**Evaluator Findings:**

The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_AFL.1.**

Upon investigation, the evaluator found that the TSS states that:

**If a user exceeds the configured number of incorrect login credentials, the user is blocked from submitting additional credentials until the lockout period has expired. However, the local console is not subjected to a lockout.**

**Verdict:**

**PASS.**

**5.1.4.1.2 FIA\_AFL.1 AGD**

The evaluator shall examine the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described.

**Evaluator Findings:**

The evaluator examined the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms

are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described.

In addition, the evaluator also found that the AGD states instructions for configuring the number of successive unsuccessful authentication attempts and time period are provided. The claimed section also provides the process of allowing the remote administrator to once again successfully log on after an Administrator defined time period has elapsed.

The relevant information is found in the following section(s): **Configure Administrator Lockout**

The evaluator shall examine the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA\_AFL.1.

**Evaluator Findings:**

The evaluator examined the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA\_AFL.1.

The relevant information is found in the following section(s): **Configure Administrator Lockout**

Upon investigation, the evaluator found that the following point in section '**Configure Administrator Lockout**' of the AGD states that:

**'The appliance ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily, by providing a local logon which is not subject to blocking.'**

**Verdict:**

**PASS.**

**5.1.4.2 FIA\_PMG\_EXT.1 PASSWORD MANAGEMENT**

**5.1.4.2.1 FIA\_PMG\_EXT.1 TSS [TD0792]**

The evaluator shall check that the TSS lists the supported special character(s) for the composition of administrator passwords.

**Evaluator Findings:**

The evaluator examined the TSS and verified that it lists the supported special character(s) for the composition of administrator passwords.

The relevant information is found in the following section(s): TOE Summary Specification  
**FIA\_PMG\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Passwords must meet the rules set by the administrator.**

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

The evaluator shall check the TSS to ensure that the minimum\_password\_length parameter is configurable by a Security Administrator.

The evaluator shall check that the TSS lists the range of values supported for the minimum\_password\_length parameter. The listed range shall include the value of 15.

#### **Evaluator Findings:**

The evaluator examined the TSS and verified that the minimum\_password\_length parameter is configurable by a Security Administrator and that it lists the range of values supported for the minimum\_password\_length parameter. The listed range includes the value of 15.

The relevant information is found in the following section(s): TOE Summary Specification  
**FIA\_PMG\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Minimum password lengths are configurable for 1 to 99 characters. When in NDPP mode, the minimum supported length is 15 characters.**

#### **Verdict:**

**PASS.**

#### **5.1.4.2.2 FIA\_PMG\_EXT.1 AGD**

The evaluator shall examine the guidance documentation to determine that it:

- a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and
- b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported.

### Evaluator Findings:

The evaluator examined the guidance documentation to determine that it:

- a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and
- b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported.

The relevant information is found in the following section(s): **Password Compliance, Enabling NDPP Mode**

Upon investigation, the evaluator found that the following points in section '**Password Compliance**' of the AGD:

- a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords

**'8. Choose how complex a user's password must be to be accepted from the Enforce password complexity drop-down menu:**

- **None (default)**
- **Alphanumeric characters— Requires both alphabetic and numeric characters**
- **Alphanumeric and symbolic characters— Requires alphabetic, numeric, and symbolic characters – for symbolic characters, only !, @, #, \$, %, ^, &, \*, (, and ) are allowed; all others are denied.**

When a password complexity option other than None is selected, the options under Complexity Requirement become active.

**9. Enter the minimum number of alphanumeric and symbolic characters required in a user's password. The default number for each is 0, but the total number of characters for all options cannot exceed 99.**

- **Upper case characters**
- **Lower case characters**
- **Numbers**
- **Symbols**

**The Symbols field becomes active only if alphanumeric and symbolic characters is selected.'**

- b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported.

**'7. Specify the shortest allowed password, enter the minimum number of characters in the Enforce a minimum password length of field. The default number is 8, the minimum is 1, and the maximum is 99.**

**Note: When in NDPP mode, the minimum supported length is 15 characters.'**

### Verdict:

PASS.

### 5.1.4.3 FIA\_UIA\_EXT.1 USER IDENTIFICATION AND AUTHENTICATION

#### 5.1.4.3.1 FIA\_UIA\_EXT.1 TSS

The evaluator shall examine the TSS to determine that it describes the logon process for each logon method (local, remote (HTTPS, SSH, etc.)) supported for the product. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a “successful logon”.

##### **Evaluator Findings:**

The evaluator examined the TSS row **FIA\_UIA\_EXT.1** to determine that it describes the logon process for remote authentication mechanism (e.g. SSH public key, Web GUI password, etc.) and optional local authentication mechanisms supported by the TOE. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a “successful logon”.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_UIA\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The SonicOS Management UI is the application used to manage the TOE devices. It is protected by HTTPS. A directly connected serial console provides a local text-based interface to manage the TOE.**

**The logon process for the SonicOS Management UI and console both require that the user enter the username and password on the logon screen. Passwords are obscured with dots to prevent an unauthorized individual from inadvertently viewing the password. The TOE hashes the user entered password and compares it to the stored hash for the associated username.**

**The authentication is considered successful and access is granted if the hashes match. If unsuccessful, the logon screen will be displayed.**

The evaluator shall examine the TSS to determine that it describes which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration.

##### **Evaluator Findings:**

The evaluator examined the TSS row **FIA\_UIA\_EXT.1** and determined that it describes which actions are allowed before administrator identification and authentication. The description shall cover authentication and identification for local and remote TOE administration.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_UIA\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that:

**A login screen displaying the administrator-configured warning banner is presented to users. Once the warning banner is accepted, in the user authentication page, there are links to Sonicwall's knowledge-base web pages that are available for the public which the users can access before the identification and authentication. The user must be identified and authenticated prior to being granted access to any security functionality.**

**No security functionality is available prior to login other than viewing the previously mentioned warning banner (except for links to Sonicwall's public KB web pages).**

For distributed TOEs the evaluator shall examine that the TSS details how Security Administrators are authenticated and identified by all TOE components. If not, all TOE components support authentication of Security Administrators according to FIA\_UIA\_EXT.1 and FIA\_UAU\_EXT.2, the TSS shall describe how the overall TOE functionality is split between TOE components including how it is ensured that no unauthorized access to any TOE component can occur.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

For distributed TOEs, the evaluator shall examine the TSS to determine that it describes for each TOE component which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration. For each TOE component that does not support authentication of Security Administrators according to FIA\_UIA\_EXT.1 and FIA\_UAU\_EXT.2 the TSS shall describe any unauthenticated services/services that are supported by the component.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.4.3.2 FIA\_UIA\_EXT.1 AGD**

The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator shall ensure the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator shall determine that the guidance documentation provides sufficient instruction on limiting the allowed services.

**Evaluator Findings:**

The evaluator examined the guidance documentation '**Managing through HTTP/HTTPS**', '**Managing through Local Console**' and determined that any necessary preparatory steps (e.g., establishing

credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator shall ensure the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator shall determine that the guidance documentation provides sufficient instruction on limiting the allowed services.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS, Managing through Local Console**

Upon investigation, the evaluator found that the AGD states that:

**'The SonicWall appliance can be managed using HTTP or HTTPS and a Web browser. HTTP web-based management is disabled by default. Use HTTPS to log into the SonicOS Management Interface with factory default settings.**

**To manage through HTTP or HTTPS**

1. **Navigate to Device | Settings > Administration.**
2. **Click Management.**
3. **To enable HTTP management globally, select Allow management via HTTP in the WEB MANAGEMENT SETTINGS section. This option is not selected by default.**
  - **The default port for HTTP is port 80, but you can configure access through another port. Enter the number of the desired port in the HTTP Port field.**

**If you configure another port for HTTP management, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you configure the port to be 76, then you must type LAN IP Address:76 into the Web browser, for example, http://192.18.16.1:76.**

- **The default port for HTTPS management is 443. To add another layer of security for logging into the SonicWall Security Appliance, change the default port, and enter the preferred port number into the HTTPS Port field.**

**If you configure another port for HTTPS management Port, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you use 700 for**

the port, then you must log into SonicWall using the port number as well as the IP address; for example, <https://192.18.16.1:700>.

To access the local console of hardware appliances:

1. Attach the included null modem cable to the appliance port marked CONSOLE. Attach the other end of the null modem cable to a serial port on the configuring computer.
2. Launch the terminal application and select the COM port.
3. Use the following settings to communicate with the serial port connected to the appliance:
  - 115,200 baud
  - 8 data bits
  - No parity
  - 1 stop bit
  - No flow control
4. Press Enter to display the DEVICE NAME> prompt.
5. At the User: prompt enter the administrator's username.

Only the administrator will be able to log in from the CLI. The default administrator's username is admin. The default username can be changed.

6. At the Password: prompt, enter the administrator's password.

If an invalid or mismatched username or password is entered, the CLI prompt returns to User:, and an error message is logged: CLI administrator login denied due to bad credentials.

To access the local console of virtual appliances:

Once the virtual appliance is installed on the ESXi we can use the ESXi IP to take the console access.

1. Take access to the ESXi server from browser.
2. Navigate to Virtual Machines and click on the appliance name.
3. The console of the appliance will be displayed.
4. Press Enter to display the DEVICE NAME> prompt.
5. At the User: prompt enter the administrator's username.
6. Only the administrator will be able to log in from the CLI. The default administrator's username is admin. The default username can be changed.
7. At the Password: prompt, enter the administrator's password.
8. If an invalid or mismatched username or password is entered, the CLI prompt returns to User:, and an error message is logged: CLI administrator login denied due to bad credentials.'

**Verdict:**

**PASS.**

#### 5.1.4.4 FIA\_UAU\_EXT.2 PASSWORD-BASED AUTHENTICATION MECHANISM

Evaluation Activities for this requirement are covered under those for FIA\_UIA\_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA\_UIA\_EXT.1.

#### 5.1.4.5 FIA\_UAU.7 PROTECTED AUTHENTICATION FEEDBACK



#### 5.1.4.5.1 FIA\_UAU.7 TSS

---

None.

#### 5.1.4.5.2 FIA\_UAU.7 AGD

---

The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed.

##### Evaluator Findings:

The evaluator examined the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed.

The relevant information is found in the following section(s): **Password Compliance**

Upon investigation, the evaluator found that the AGD states that:

**'For the Management UI, passwords are obscured with dots to prevent an unauthorized individual from inadvertently viewing the password. For the console, the passwords are obscured with blank spaces.'**

##### Verdict:

PASS.

---

### 5.1.5 SECURITY MANAGEMENT (FMT)

---

#### 5.1.5.1 FMT\_MOF.1/MANUALUPDATE

---

##### 5.1.5.1.1 FMT\_MOF.1/MANUALUPDATE TSS

---

For distributed TOEs see chapter 2.4.1.1. There are no specific requirements for non-distributed TOEs.

##### Evaluator Findings:

The TOE is not a distributed TOE and there are no specific requirements for non-distributed TOES; hence, this assurance activity is not applicable.

##### Verdict:

PASS.

---

##### 5.1.5.1.2 FMT\_MOF.1/MANUALUPDATE AGD

---

The evaluator shall examine the guidance documentation to determine that any necessary steps to perform manual update are described. The guidance documentation shall also provide warnings regarding functions that may cease to operate during the update (if applicable).

## Evaluator Findings:

The evaluator examined the guidance documentation '**Firmware Upgrade**' and determined that any necessary steps to perform manual update are described. The guidance documentation '**Firmware Upgrade**' also provides warnings regarding functions that may cease to operate during the update (if applicable).

The relevant information is found in the following section(s): **Firmware Upgrade**

Upon investigation, the evaluator found that the AGD states necessary steps to perform manual update and also provides warnings regarding functions that may cease to operate during the update:

### **'To upload new firmware**

1. **Download the SonicOS firmware image file from MySonicWall and save it to a location on your local computer.**
2. **Point your browser to the appliance IP address and log in as an administrator.**
3. **In the DEVICE view, on the Settings > Firmware and Settings page, on the Firmware & Local Backups screen, click Upload Firmware.**
4. **In the Backup of current settings popup dialog, click OK to continue the firmware upload.**
5. **In the Upload Firmware dialog, browse to the location where you saved the SonicOS firmware image file, select the file, and click Upload.**

The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware. When a new firmware image is loaded on the physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. When a new image is loaded on a virtual appliance, the cryptographic module verifies the RSA signed SHA-256 hash of the image.

- **If the signature verification succeeds, the firmware is automatically installed.**
  - **If the signature verification fails, the firmware is not loaded and an error appears. Uploading the same firmware is disallowed.**
  - **After the firmware finishes uploading, it is displayed in the table on the Firmware & Local Backups screen.**
  - **Firmware& Local Backup tab shows the current image and recently downloaded inactive image.**
6. **Click the Boot icon in the Uploaded Firmware Version row and select Boot firmware with Current Configuration.**

**Note: Once the new version is installed as the boot image, previously installed image gets replaced**

7. **In the Warning dialog box, click OK. The appliance restarts and displays the login page.**

**Enter your username and password. Your new SonicOS image version information is displayed on the Settings > Status page.'**

Additionally, the evaluator found that the AGD states that: **No functionality will cease during the update process. The device will remain fully operational until the administrator reboots the product.**

For distributed TOEs the guidance documentation shall describe all steps how to update all TOE components. This shall contain description of the order in which components need to be updated if the order is relevant to the update process. The guidance documentation shall also provide warnings regarding functions of TOE components and the overall TOE that may cease to operate during the update (if applicable).

**Evaluator Findings:**

The TOE is not a distributed TOE; hence, this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.5.2 FMT\_MTD.1/COREDATA MANAGEMENT OF TSF DATA**

**5.1.5.2.1 FMT\_MTD.1/COREDATA TSS**

The evaluator shall examine the TSS to determine that, for each administrative function identified in the guidance documentation; those that are accessible through an interface prior to administrator log-in are identified. For each of these functions, the evaluator shall also confirm that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users.

**Evaluator Findings:**

The evaluator confirmed that the TSS row **FMT\_MTD.1/CoreData** details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users.

The relevant information is found in the following section(s): TOE Summary Specification **FMT\_MTD.1/CoreData**.

Upon investigation, the evaluator found that the TSS states that: **The TOE security functions are managed locally and remotely through the web-based management interface and restricted to authorized users assigned the Security Administrator role.**

**Security Administrators must authenticate with the TOE prior to accessing any of the administrative functions.**

**No management of TSF data may be performed through any interface prior to login.**

**Only administrators may login to the administrative interface, ensuring that access to TSF data is disallowed for non-administrative users.**

If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator shall examine the TSS to determine that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted.

**Evaluator Findings:**

If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator examined the TSS row **FMT\_MTD.1/CoreData** and determined that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted.

The relevant information is found in the following section(s): TOE Summary Specification **FMT\_MTD.1/CoreData.**

Upon investigation, the evaluator found that the TSS explains that **only administrators may login to the administrative interface, ensuring that access to TSF data is disallowed for non-administrative users.** Additionally, the TSS also explains that **only the security administrators can Import/delete X.509v3 certificates, and generate/delete cryptographic keys**

**Verdict:**

**PASS.**

**5.1.5.2.2 FMT\_MTD.1/COREDATA AGD**

The evaluator shall review the guidance documentation to determine that each of the TSF-data-manipulating functions implemented in response to the requirements of the c PP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.

**Evaluator Findings:**

The evaluator reviewed the guidance and determined that each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.

The relevant information is found in the following section(s): **Enabling NDPP Mode, Product Administration, User Session Settings, Firmware Upgrade, Setting System Time, Audit Server**

**Configuration, Configure VPN, Configuring Access Rules for a Zone and Configuring IPS Protection on Zones.**

Upon investigation, the evaluator found that the AGD section **Product Administration** states that: **'Only authorized administrators can update and modify product functions.'**

If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator shall also review the guidance documentation to determine that it explains how to designate a CA certificate a trust anchor.

**Evaluator Findings:**

If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator reviewed the guidance documentation **'Managing Certificates'** and determined that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator reviewed the guidance documentation **'Importing a Certificate Authority Certificate'** and determined that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator also reviewed the guidance documentation **'Certificates Table'** and determined that it explains how to designate a CA certificate a trust anchor.

The relevant information is found in the following section(s): **Managing Certificates, Importing a Certificate Authority Certificate and Certificates Table**

Upon investigation, the evaluator found that the AGD section **'Managing Certificates'** states that: **'You import the valid CA certificate into the firewall using the Device | Settings > Certificates page. After you import the valid CA certificate, you can use it to validate your local certificates. SonicOS provides a large number of certificates with the SonicWall network security appliance; these are built-in certificates and cannot be deleted or configured.'**

Furthermore, the evaluator found that the AGD section **'Importing a Certificate Authority Certificate'** provides sufficient information for the administrator to securely load CA certificates into the trust store:

**'To import a certificate from a certificate authority**

- 1. Navigate to Device | Settings > Certificates.**
- 2. Click Import. The IMPORT CERTIFICATE dialog is displayed.**
- 3. Choose Import a CA certificate from a PKCS#7 (\*.p7b) or DER (.der or .cer) encoded file. The Import Certificate dialog settings change.**

4. Click Add File and locate the certificate file.
5. Click Open.
6. Click Import to import the certificate into the firewall. When it is imported, you can view the certificate entry in the Certificates table.
7. Click the certificate displayed on the Certificates page to see the status and other details.'

Lastly the evaluator found that the AGD section 'Certificates Table' that explains how to designate a CA certificate a trust anchor:

'The Certificates page provides all the settings for managing CA and Local Certificates. The table page displays this information about certificates:

Column	Information Displayed
CERTIFICATE	Name of the certificate.
TYPE	Type of certificate: <ul style="list-style-type: none"> <li>• CA certificate</li> <li>• Local certificate</li> <li>• Pending request'</li> </ul>

**Verdict:**

**PASS.**

**5.1.5.3 FMT\_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS**

**5.1.5.3.1 FMT\_SMF.1 TSS (CONTAINING ALSO REQUIREMENTS ON GUIDANCE DOCUMENTATION AND TESTS)**

The evaluator shall examine the TSS, Guidance Documentation and the TOE as observed during all other testing and shall confirm that the management functions specified in FMT\_SMF.1 are provided by the TOE.

**Evaluator Findings:**

The evaluator examined the TSS row **FMT\_SMF.1**, the Guidance Documentation and the TOE as observed during all other testing and confirmed that the management functions specified in FMT\_SMF.1 are provided by the TOE.

The relevant information is found in the following section(s): TOE Summary Specification **FMT\_SMF.1** and below mentioned AGD sections.

Upon investigation, the evaluator found that all the details for the management functions specified in FMT\_SMF.1 and the administrative activities are consistent with the TSS. Further, the TSS lists the following management activities:

**The security administrator is able to perform the following functions:**

- **Administer the TOE locally and remotely;**
- **Configure the access banner;**
- **Configure session inactivity time before session termination or locking;**
- **Manually update the TOE;**
- **Configure the authentication failure parameters;**
- **Configure the cryptographic functionality;**
- **Generate and delete cryptographic keys (generate and delete the cryptographic keys associated with CSRs);**
- **Configure the IPsec functionality;**
- **Import X.509v3 certificates;**
- **Ability to modify (enable/disable) transmission of audit records to an external audit server;**
- **Ability to set the time;**
- **Import and delete X509 Certificates;**
- **Configure firewall rules;**
- **Configure packet filtering rules and associated parameters;**
- **Enable and disable signatures applied to sensor interfaces;**
- **Modify parameters for IPS/IDS;**
- **Import IPS signature databases and custom create IPS signatures;**
- **Configure anomaly detection, time-based detection/prevention, thresholds, known-good & known-bad IP lists, and actions.**

The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface).

#### **Evaluator Findings:**

The evaluator confirmed that the TSS row **FMT\_SMF.1** details which security management functions are available through which interface(s) (local administration interface, remote administration interface).

The relevant information is found in the following section(s): TOE Summary Specification row **FMT\_SMF.1**.

Upon investigation, the evaluator found that the TSS states: **All the management functions can be performed via Web GUI and local console by security administrators.**

The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface.

<b>Evaluator Findings:</b>
<p>The evaluator examined the TSS row <b>FMT_SMF.1</b> and the Guidance Documentation section <b>Managing through Local Console</b> to verify they both describe the local administrative interface.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FMT_SMF.1</b> and AGD section <b>Managing through Local Console</b></p> <p>Upon investigation, the evaluator found that the TSS states that: <b>The TOE security functions are managed locally and remotely through the web-based management interface.</b></p> <p>Upon investigation, the evaluator found that the AGD describes <b>the local interface and the configurations required to communicate on the interface.</b></p>

The evaluator shall ensure the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.

<b>Evaluator Findings:</b>
<p>The evaluator ensured the Guidance Documentation <b>Managing through Local Console</b> includes appropriate warnings for the administrator to ensure the interface is local.</p> <p>The relevant information is found in the following section(s): <b>Managing through Local Console</b></p> <p>Upon investigation, the evaluator found that the AGD states that: <b>Attach the included null modem cable to the appliance port marked CONSOLE.</b> This sufficiently ensures that the interface is a local interface because the only CLI based management interface is the local console and it is expected that a knowledgeable security administrators are managing the device to identify the difference between the CLI and a webGUI.</p>

For distributed TOEs with the option 'ability to configure the interaction between TOE components' the evaluator shall examine that the ways to configure the interaction between TOE components is detailed in the TSS and Guidance Documentation.

<b>Evaluator Findings:</b>
<p>The TOE is not a distributed TOE hence this assurance activity is not applicable.</p>

The evaluator shall check that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS and Guidance Documentation.



**Evaluator Findings:**

The evaluator checked that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS row **FMT\_SMF.1** and the Guidance Documentation. This was verified throughout the testing process during the evaluation.

**Verdict:**

**PASS.**

#### 5.1.5.3.2 FMT\_SMF.1 AGD

---

**Evaluator Findings:**

See section 5.1.4.3.1 of this document for AGD activities.

**Verdict:**

**PASS.**

---

#### 5.1.5.4 FMT\_SMF.1/FFW SPECIFICATION OF MANAGEMENT FUNCTIONS

##### 5.1.5.4.1 FMT\_SMF.1/FFW

---

The evaluation activities specified for FMT\_SMF.1 in the Supporting Document for the Base-PP shall be applied in the same way to the newly added management functions defined in FMT\_SMF.1/FFW in the FW Module.

**Evaluator Findings:**

The evaluator reviewed the AGD sections '**Firewall**' and ST section TSS row **FMT\_SMF.1/FFW** and verified that the evaluation activities specified for FMT\_SMF.1 in the Supporting Document for the Base-PP have been applied in the same way to the newly added management functions defined in FMT\_SMF.1/FFW in the FW Module.

Upon investigation, the evaluator found that all the details for the management functions specified in FMT\_SMF.1 and FMT\_SMF.1/FFW and the administrative activities are consistent with the TSS. The TSS lists the following management activities related to FMT\_SMF.1/FFW:

**The security administrator is able to perform the following functions:**

- ...
- **Configure firewall rules;**
- ...

All other information about the interfaces, local interface descriptions, and warnings is covered in section 5.1.5.3.1 above.

**Verdict:**

**PASS.**

**5.1.5.5 FMT\_SMF.1/VPN SPECIFICATION OF MANAGEMENT FUNCTIONS**

**5.1.5.5.1 FMT\_SMF.1/VPN TSS**

The evaluator shall examine the TSS to confirm that all management functions specified in FMT\_SMF.1/VPN are provided by the TOE.

**Evaluator Findings:**

The evaluator reviewed the TSS, section '**FMT\_SMF.1/VPN**' row and ensured that the TOE provides all management functions specified in FMT\_SMF.1/VPN.

The relevant information is found in the following section(s): TOE Summary Specification row **FMT\_SMF.1/VPN**

Upon investigation, the evaluator found that the TSS states that:

**The security administrator is able to perform the following functions:**

- ...
- **Configure packet filtering rules and associated parameters;**
- ...

**Rules for VPN traffic are configured through the Firewall Access Rules. The Administrator navigates to Firewall > Access Rules and selects the 'Matrix' checkbox. Under 'Zones', the Administrator can select VPN to LAN, WAN or VPN and then configures the rules. This will configure rules specifically for the VPN traffic. Firewall Access Rules for non-VPN traffic are configured using the same method by selecting the appropriate zones.**

As with FMT\_SMF.1 in the Base-PP, the evaluator shall ensure that the TSS identifies what logical interfaces are used to perform these functions and that this includes a description of the local administrative interface.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it identifies what logical interfaces are used to perform these functions including a description of the local administrative interface.

The relevant information is found in the following section(s): TOE Summary Specification row **FMT\_SMF.1/VPN**

Upon investigation, the evaluator found that the TSS states that:

**'The TOE security functions are managed locally and remotely through the web-based management interface and restricted to authorized users assigned the Security Administrator role.**

**All the management functions can be performed via Web GUI and local console by security administrators.'**

**Verdict:**

**PASS.**

**5.1.5.5.2 FMT\_SMF.1/VPN AGD**

The evaluator shall examine the operational guidance to confirm that all management functions specified in FMT\_SMF.1/VPN are provided by the TOE.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that all management functions specified in FMT\_SMF.1/VPN are provided by the TOE. Configuration of these management functions are described in sections **IPSec VPN** and **Firewall** in the AGD.

As with FMT\_SMF.1 in the Base-PP, the evaluator shall ensure that the operational guidance identifies what logical interfaces are used to perform these functions and that this includes a description of the local administrative interface.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it identifies what logical interfaces are used to perform these functions and that this includes a description of the local administrative interface. The logical interfaces are **WebGUI and the local console**. The AGD describes the local administrative interface in section **Managing through the Local Console**.

**Verdict:**

**PASS.**

**5.1.5.6 SPECIFICATION OF MANAGEMENT FUNCTIONS (FMT\_SMF)**

**5.1.5.6.1 FMT\_SMF.1/IPS SPECIFICATION OF MANAGEMENT FUNCTIONS (IPS) TSS**

The evaluator shall verify that the TSS describes how the IPS data analysis and reactions can be configured. This may be performed in conjunction with the evaluation of IPS\_ABD\_EXT.1, IPS\_IPB\_EXT.1, and IPS\_SBD\_EXT.1

<b>Evaluator Findings:</b>
<p>The evaluator reviewed the TSS to ensure that it describes how the IPS data analysis and reactions can be configured.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FMT_SMF.1/IPS</b></p> <p>Upon investigation, the evaluator found that the TSS states that: <b>Administrators can configure the IPS data analysis by selecting signatures from a pre-loaded list or by creating custom signatures. Custom Signatures are created using a combination of Application and Access rules. If a signature calls for matching L3/L4 header content, the Packet Dissection Filter can be used in conjunction with the rules. If the signature calls for application layer header/data matching, the application rules can be created with custom policy and match objects to match the desired offset in the application layer header or payload. The IPS data analysis configuration options provide the ability to deploy selections globally to either all WAN or all LAN interfaces. The access rule policies can be configured to Allow, Deny, and Discard undesired traffic.</b></p>

**Verdict:**  
**PASS.**

#### 5.1.5.6.2 FMT\_SMF.1/IPS SPECIFICATION OF MANAGEMENT FUNCTIONS (IPS) AGD

The evaluator shall verify that the operational guidance describes the instructions for each function defined in the SFR, describes how to configure the IPS data analysis and reactions, including how to set any configurable defaults and how to configure each of the applicable analysis pattern matching methods and reaction modes.

<b>Evaluator Findings:</b>
<p>The evaluator checked the AGD and ensured that it describes the instructions for each function defined in the SFR, describes how to configure the IPS data analysis and reactions, including how to set any configurable defaults and how to configure each of the applicable analysis pattern matching methods and reaction modes.</p> <p>The relevant information is found in the following section(s): <b>Signatures, Match Objects, Packet Dissection Objects, Adding Access Rules, Configuring IPS Protection on Zones and App Rules</b></p> <p>Upon investigation, the evaluator found that the AGD section ‘<b>Signatures</b>’ states that: <b>Administrators can configure the IPS data analysis by selecting signatures from a pre-loaded list or by creating custom signatures. Custom Signatures are created using a combination of Application and Access rules. If a signature calls for matching L3/L4 header content, the Packet Dissection Filter can be used in conjunction with the rules. If the signature calls for application layer header/data matching, the application rules can be created with custom policy and match objects to match the</b></p>

desired offset in the application layer header or payload. The IPS data analysis configuration options provide the ability to deploy selections globally to either all WAN or all LAN interfaces.

Upon investigation, the evaluator found that the AGD describes the Web GUI steps to configure each of the function defined in the SFR as follows:

Configuration of Packet dissection objects	<b>Packet Dissection Objects</b>
Application of packet dissection object to an Access Rule	<b>Point number 26</b> in section <b>Adding Access Rules</b>
Configuration of custom signatures	<b>Match Objects</b>
Application of custom signatures to Application Rule	<b>Configuring IPS Protection on Zones and App Rules</b>

**Verdict:**

**PASS.**

#### 5.1.5.7 FMT\_SMR.2 RESTRICTIONS ON SECURITY ROLES

##### 5.1.5.7.1 FMT\_SMR.2 TSS

The evaluator shall examine the TSS to determine that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE.

#### **Evaluator Findings:**

The evaluator examined the TSS row **FMT\_SMR.2** and determined that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE.

The relevant information is found in the following section(s): TOE Summary Specification **FMT\_SMR.2**.

Upon investigation, the evaluator found that the TSS states that: **The TOE security functions are managed locally and remotely through the web-based management interface and restricted to authorized users assigned the Security Administrator role.**

**Security Administrators must authenticate with the TOE prior to accessing any of the administrative functions.**

**Verdict:**

**PASS.**

##### 5.1.5.7.2 FMT\_SMR.2 AGD

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

## Evaluator Findings:

The evaluator reviewed the AGD '**Managing through HTTP/HTTPS**', '**Managing through Local Console**' and ensured that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS, Managing through Local Console**

Upon investigation, the evaluator found that the claimed AGD sections contains instructions for administering the TOE both locally and remotely:

### **'To manage through HTTP or HTTPS**

1. **Navigate to Device | Settings > Administration.**
2. **Click Management.**
3. **To enable HTTP management globally, select Allow management via HTTP in the WEB MANAGEMENT SETTINGS section. This option is not selected by default.**
  - **The default port for HTTP is port 80, but you can configure access through another port. Enter the number of the desired port in the HTTP Port field.**

**If you configure another port for HTTP management, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you configure the port to be 76, then you must type LAN IP Address:76 into the Web browser, for example, http://192.18.16.1:76.**

- **The default port for HTTPS management is 443. To add another layer of security for logging into the SonicWall Security Appliance, change the default port, and enter the preferred port number into the HTTPS Port field.**

**If you configure another port for HTTPS management Port, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you use 700 for**

the port, then you must log into SonicWall using the port number as well as the IP address; for example, <https://192.18.16.1:700>.

To access the local console of hardware appliances

1. Attach the included null modem cable to the appliance port marked CONSOLE. Attach the other end of the null modem cable to a serial port on the configuring computer.
2. Launch the terminal application and select the COM port.
3. Use the following settings to communicate with the serial port connected to the appliance:
  - 115,200 baud
  - 8 data bits
  - No parity
  - 1 stop bit
  - No flow control
4. Press Enter to display the DEVICE NAME> prompt.
5. At the User: prompt enter the administrator's username.

Only the administrator will be able to log in from the CLI. The default administrator's username is admin. The default username can be changed.

6. At the Password: prompt, enter the administrator's password.

If an invalid or mismatched username or password is entered, the CLI prompt returns to User:, and an error message is logged: CLI administrator login denied due to bad credentials.

To access the local console of virtual appliances

Once the virtual appliance is installed on the ESXi we can use the ESXi IP to take the console access.

1. Take access to the ESXi server from browser.
2. Navigate to Virtual Machines and click on the appliance name.
3. The console of the appliance will be displayed.
4. Press Enter to display the DEVICE NAME> prompt.
5. At the User: prompt enter the administrator's username.
6. Only the administrator will be able to log in from the CLI. The default administrator's username is admin. The default username can be changed.
7. At the Password: prompt, enter the administrator's password.
8. If an invalid or mismatched username or password is entered, the CLI prompt returns to User:, and an error message is logged: CLI administrator login denied due to bad credentials.'

Verdict:

PASS.

---

## 5.1.6 PROTECTION OF THE TSF (FPT)

---

### 5.1.6.1 FPT\_APW\_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS

---

#### 5.1.6.1.1 FPT\_APW\_EXT.1 TSS

The evaluator shall examine the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS

shall also detail passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note.

**Evaluator Findings:**

The evaluator examined the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS also detailed passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note.

The relevant information is found in the following section(s): TOE Summary Specification  
**FPT\_APW\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that:  
**The TSF protects the administrator passwords used to access the device. Passwords and other sensitive data in the configuration file is protected with AES-256 hash. The user interface does not support viewing passwords.**

**Verdict:**

**PASS.**

**5.1.6.2 FPT\_SKP\_EXT.1 PROTECTION OF TSF DATA (FOR READING OF ALL PRE-SHARED, SYMMETRIC AND PRIVATE KEYS)**

**5.1.6.2.1 FPT\_SKP\_EXT.1 TSS**

The evaluator shall examine the TSS to determine that it details how any pre- shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS shall describe how they are protected/obscured.

**Evaluator Findings:**

The evaluator examined the TSS row **FPT\_SKP\_EXT.1** and determined that it details how any pre-shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS describes how they are protected/obscured. This information is present in the section **6.2, table 34– Cryptographic Key Destruction** in the TSS.

The relevant information is found in the following section(s): TOE Summary Specification  
**FPT\_SKP\_EXT.1**

Upon investigation, the evaluator found that the TSS states that: **The TSF does not include any function that allows symmetric keys or private keys to be displayed or exported. The use of shared secrets is not supported in the evaluated configuration. Keys may only be accessed for the purposes of their assigned security functionality.**



**Verdict:**

**PASS.**

**5.1.6.3 FPT\_STM\_EXT.1 RELIABLE TIME STAMPS**

**5.1.6.3.1 FPT\_STM\_EXT.1 TSS [TD0632]**

The evaluator shall examine the TSS to ensure that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions.

**Evaluator Findings:**

The evaluator examined the TSS row **FPT\_STM\_EXT.1** and ensured that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions.

The relevant information is found in the following section(s): TOE Summary Specification **FPT\_STM\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE provides reliable time stamps that are used for audit records.**

**The System > Time page of the web management GUI may be used to configure the time and date settings. In the evaluated configuration, time is set manually. This may be configured by deselecting 'Set time automatically using NTP and populating the appropriate values for daylight savings time adjustments and time format.**

**Only authorized administrators have the required privilege to set the time.**

**Time is maintained by the system clock, which is implemented in the TOE hardware and software. Changes to the time are audited. Therefore, the time services provided are considered to be reliable.**

**Authorized administrators may make changes to the time using the GUI.**

If “obtain time from the underlying virtualization system” is selected, the evaluator shall examine the TSS to ensure that it identifies the VS interface the TOE uses to obtain time. If there is a delay between updates to the time on the VS and updating the time on the TOE, the TSS shall identify the maximum possible delay.

**Evaluator Findings:**

The ST does not select “obtain time from the underlying virtualization system” hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

### 5.1.6.3.2 FPT\_STM\_EXT.1 AGD [TD0632]

The evaluator examines the guidance documentation to ensure it instructs the administrator how to set the time.

#### Evaluator Findings:

The evaluator examined the guidance documentation ‘**Setting System Time**’ and ensured that it instructs the administrator how to set the time.

The relevant information is found in the following section(s): **Setting System Time**

Upon investigation, the evaluator found that the claimed AGD section states that:

**‘The system time can be set in the section Device | Settings > Time**

**To set the system time:**

- 1. Navigate to Device | Settings > Time.**
- 2. Select the time zone you are in from Time Zone.**
- 3. Disable Set time automatically using NTP. The Time and Date options become available.**
- 4. Select the time in the 24-hour format using the Time (hh:mm:ss) drop-down menus**
- 5. Select the date from the Date drop-down menus.’**

If the TOE supports the use of an NTP server, the guidance documentation instructs how a communication path is established between the TOE and the NTP server, and any configuration of the NTP client on the TOE to support this communication.

#### Evaluator Findings:

The TOE does support the use of NTP server; however, it was excluded from the scope of the evaluation, and hence, this activity is not applicable.

If the TOE supports obtaining time from the underlying VS, the evaluator shall verify the guidance documentation specifies any configuration steps necessary. If no configuration is necessary, no statement is necessary in the guidance documentation. If there is a delay between updates to the time on the VS and updating the time on the TOE, the evaluator shall ensure the guidance documentation informs the administrator of the maximum possible delay.

#### Evaluator Findings:

The TOE does not support the use of obtain time from the underlying virtualization system; hence, this activity is not applicable.

#### Verdict:

**PASS.**

### 5.1.6.4 FPT\_TST\_EXT.1 TSF TESTING

#### 5.1.6.4.1 FPT\_TST\_EXT.1 TSS

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

#### **Evaluator Findings:**

The evaluator examined the TSS row **FPT\_TST\_EXT.1** and ensured that it details the self-tests that are run by the TSF; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" is used).

The relevant information is found in the following section(s): TOE Summary Specification **FPT\_TST\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE performs a power on self-test on each device when it is powered on. The following tests are performed:**

**CPU Test - This includes tests and set-up of the following:**

- **MMU**
- **Memory**
- **I/O ports**
- **Interrupts**
- **Timers**

**RAM Test - A memory test is performed.**

**Following these tests, the TSF performs self-tests on the cryptographic module. The following cryptographic algorithm self-tests are performed by the cryptographic module at power-up:**

- **Firmware integrity test**
- **AES-CBC/AES-GCM Encrypt and Decrypt Known Answer Tests**
- **SHA-1, -256, -384, -512 Known Answer Tests**
- **HMAC-SHA-1, -256, -512 Known Answer Tests**
- **DSA Signature Verification Pairwise Consistency Test**
- **RSA Sign and Verify Known Answer Tests**
- **DH Pairwise Consistency Test**
- **DRBG Known Answer Test**
- **ECDSA Known Answer Test**
- **ECDSA Signature and Verification Known Answer Tests**

**For the memory test, 32K bytes of memory are tested in two steps. First, 1 or 0 is written to memory and read to verify. After that, a specific value will be written to the memory and be compared.**

**The physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. For virtual appliances, the RSA signed SHA-512 hash During the startup process on of the image is verified.**

The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.

<b>Evaluator Findings:</b>
<p>The evaluator ensured that the TSS row <b>FPT_TST_EXT.1</b> makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FPT_TST_EXT.1</b>.</p> <p>Upon investigation, the evaluator found that the TSS states that:</p> <p><b>If any of the tests fail, the cryptographic module enters the error state. No security services are provided in the error state. Upon successful completion of the Diagnostic Phase, the cryptographic module enters the Command and Traffic Processing State. Security services are only provided in the Command and Traffic Processing State. No VPN tunnels are started until all tests are successfully completed. This effectively inhibits the data output interface. When all tests are completed successfully, the Test Light Emitting Diode (LED) is turned off.</b></p> <p><b>The SonicWall device is essentially a Finite State Machine that is synonymous with the cryptographic module. Therefore, the cryptographic module self-tests are entirely sufficient to demonstrate the correct operation of the TOE.</b></p>

For distributed TOEs the evaluator shall examine the TSS to ensure that it details which TOE component performs which self-tests and when these self-tests are run.

<b>Evaluator Findings:</b>
<p>The TOE is not a distributed TOE hence this assurance activity is not applicable.</p>

**Verdict:**

**PASS.**

**5.1.6.4.2 FPT\_TST\_EXT.1 AGD**

The evaluator shall also ensure that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.

<b>Evaluator Findings:</b>
<p>The evaluator also ensured that the guidance documentation <b>'Startup and Self-Test'</b> describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors correspond to those described in the TSS.</p> <p>The relevant information is found in the following section(s): <b>Startup and Self-Test</b></p>

Upon investigation, the evaluator found that the claimed AGD section states that:

**'If any of these tests fail, the product enters a hard error state that requires administrative intervention. Rebooting the product generally clears the errors. However, if errors persist, contact [SonicWall Technical Support](#).'**

For distributed TOEs the evaluator shall ensure that the guidance documentation describes how to determine from an error message returned which TOE component has failed the self-test.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.6.4.3 FPT\_TST\_EXT.1 (VPNGW)**

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module requires a particular self-test to be performed, but this self-test is still evaluated using the same methods specified in the Supporting Document.

**5.1.6.5 FPT\_TST\_EXT.3 SELF-TEST WITH DEFINED METHODS**

**5.1.6.5.1 FPT\_TST\_EXT.3 TSS**

The evaluator shall verify that the TSS describes the method used to perform self-testing on the TSF executable code, and that this method is consistent with what is described in the SFR.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FPT\_TST\_EXT.3** row, to ensure that it describes the method used to perform self-testing on the TSF executable code, and that this method is consistent with what is described in the SFR.

The relevant information is found in the following section(s): TOE Summary Specification row **FPT\_TST\_EXT.3**

Upon investigation, the evaluator found that the claimed TSS section states that:

**For the physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. For virtual appliances, the RSA signed SHA-512 hash of the image is verified.**

This method is consistent with what is described in the SFR.

**Verdict:**

**PASS.**

**5.1.6.5.2 FPT\_TST\_EXT.3 AGD**

There are no guidance EAs for this component.

**5.1.6.6 FPT\_TUD\_EXT.1 TRUSTED UPDATE**

**5.1.6.6.1 FPT\_TUD\_EXT.1 TSS**

The evaluator shall verify that the TSS describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS needs to describe how and when the inactive version becomes active. The evaluator shall verify this description.

**Evaluator Findings:**

The evaluator verified that the TSS row **FPT\_TUD\_EXT.1** describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS describes how and when the inactive version becomes active. The evaluator verified this description.

The relevant information is found in the following section(s): TOE Summary Specification **FPT\_TUD\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **TSF software can be updated through the web interface using the System > Settings page. This page displays the current firmware image version.**

**Firmware can be uploaded, but not activated. The new firmware will not be activated until the administrator boots the device with the new firmware by selecting the new firmware and 'Boot'.**

**The version of firmware running may be queried through the TOE UI. The version of the most recently installed firmware may also be queried through the TOE UI.**

The evaluator shall verify that the TSS describes all TSF software update mechanisms for updating the system firmware and software (for simplicity the term 'software' will be used in the following although the requirements apply to firmware and software).

**Evaluator Findings:**

The evaluator verified that the TSS describes all TSF software update mechanisms for updating the system firmware and software.

The relevant information is found in the following section(s): TOE Summary Specification  
**FPT\_TUD\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **TSF software can be updated through the web interface using the System > Settings page.**

The evaluator shall verify that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS shall detail this mechanism instead of the digital signature verification mechanism.

**Evaluator Findings:**

The evaluator verified that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS details this mechanism instead of the digital signature verification mechanism.

The relevant information is found in the following section(s): TOE Summary Specification  
**FPT\_TUD\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware.**

**If the signature verification fails, the firmware is not loaded and an error appears.**

The evaluator shall verify that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.

**Evaluator Findings:**

The evaluator verified that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.

The relevant information is found in the following section(s): TOE Summary Specification  
**FPT\_TUD\_EXT.1..**

Upon investigation, the evaluator found that the TSS states that: **To update the firmware, the administrator must first download the firmware update from SonicWall and save it to an accessible location.**

**The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware.**

**When a new firmware image is loaded on the claimed TZ, NSa, and NSSP physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image.**

**When a new image is loaded on a virtual appliance, the cryptographic module verifies the RSA signed SHA-256 hash of the image. If the signature verification succeeds, the firmware is automatically installed.**

**If the signature verification fails, the firmware is not loaded and an error appears.**

If the options 'support automatic checking for updates' or 'support automatic updates' are chosen from the selection in FPT\_TUD\_EXT.1.2, the evaluator shall verify that the TSS explains what actions are involved in automatic checking or automatic updating by the TOE, respectively.

**Evaluator Findings:**

The options 'support automatic checking for updates' or 'support automatic updates' are not selected in the ST hence this assurance activity is not applicable.

For distributed TOEs, the evaluator shall examine the TSS to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component. Alternatively, this description can be provided in the guidance documentation. In that case the evaluator should examine the guidance documentation instead.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

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



**Evaluator Findings:**

ST does not claim 'Published hash' hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.6.6.2 FPT\_TUD\_EXT.1 AGD**

The evaluator shall verify that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation needs to describe how to query the loaded but inactive version.

**Evaluator Findings:**

The evaluator verified that the guidance documentation section '**Firmware Management**' describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation section '**Firmware Upgrade**' describes how to query the loaded but inactive version.

The relevant information is found in the following section(s): **Firmware Management** and **Firmware Upgrade**

Upon investigation, the evaluator found that the AGD section '**Firmware Management**' states that: '**To verify current firmware version, navigate to Device | Settings | Firmware and Settings.**'

Furthermore, the evaluator examined section '**Firmware Upgrade**' of the AGD and determined that it describes how to query the loaded but inactive version. The bullet points under AGD section '**Firmware Upgrade**' states that:

- '**After the firmware finishes uploading, it is displayed in the table on the Firmware & Local Backups screen.**
- '**Firmware & Local Backup tab now shows the Current Firmware Version and recently Uploaded Firmware Version which is inactive image.**

**Note: Once the new version is installed as the boot image, the previously installed image gets replaced.'**

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

**Evaluator Findings:**

The evaluator verified that the guidance documentation '**Firmware Upgrade**' describes how the verification of the authenticity of the update is performed (digital signature verification). The description includes the procedures for successful and unsuccessful verification. The description corresponds to the description in the TSS.

The relevant information is found in the following section(s): **Firmware Upgrade**

Upon investigation, the evaluator found that the AGD section states that:

**‘The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware. When a new firmware image is loaded on the physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. When a new image is loaded on a virtual appliance, the cryptographic module verifies the RSA signed SHA-256 hash of the image.**

- **If the signature verification succeeds, the firmware is automatically installed.**
- **If the signature verification fails, the firmware is not loaded, and an error appears. Uploading the same firmware is disallowed.’**

If a published hash is used to protect the trusted update mechanism, the evaluator shall verify that the guidance documentation describes how the Security Administrator can obtain authentic published hash values for the updates.

**Evaluator Findings:**

ST does not claim ‘Published hash’ hence this assurance activity is not applicable.

For distributed TOEs the evaluator shall verify that the guidance documentation describes how the versions of individual TOE components are determined for FPT\_TUD\_EXT.1, how all TOE components are updated, and the error conditions that may arise from checking or applying the update (e.g. failure of signature verification, or exceeding available storage space) along with appropriate recovery actions. . The guidance documentation only has to describe the procedures relevant for the Security Administrator; it does not need to give information about the internal communication that takes place when applying updates.

**Evaluator Findings:**

TOE is not a distributed TOE hence this assurance activity is not applicable.

If this information was not provided in the TSS: For distributed TOEs, the evaluator shall examine the Guidance Documentation to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component.

**Evaluator Findings:**

The TOE is not a distributed TOE hence this assurance activity is not applicable.

If this information was not provided in the TSS: If the ST author indicates that a certificate-based mechanism is used for software update digital signature verification, the evaluator shall verify that the

Guidance Documentation contains a description of how the certificates are contained on the device. The evaluator also ensures that the Guidance Documentation describes how the certificates are installed/updated/selected, if necessary.

**Evaluator Findings:**

Certificate-based mechanism is not used for software update digital signature verification hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.1.6.6.3 FPT\_TUD\_EXT.1 (VPNGW)**

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to mandate that a particular selection be chosen, but this selection is part of the original definition of the SFR so no new behavior is defined by the PP-Module.

**5.1.6.7 FPT\_FLS.1/SELFTEST FAILURE WITH PRESERVATION OF SECURE STATE (SELF-TEST FAILURES)**

**5.1.6.7.1 FPT\_FLS.1/SELFTEST TSS**

The evaluator shall ensure the TSS describes how the TOE ensures a shutdown upon a self-test failure, a failed integrity check of the TSF executable image, or a failed health test of the noise source.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FPT\_FLS.1/SelfTest** row, and ensured that it describes how the TOE ensures a shutdown upon a self-test failure, a failed integrity check of the TSF executable image, or a failed health test of the noise source.

The relevant information is found in the following section(s): TOE Summary Specification row **FPT\_FLS.1/SelfTest**

Upon investigation, the evaluator found that the claimed TSS section states that:

**An integrity check of the TSF executable image is run when the image is loaded. A Continuous Random Number Generator Test (CRNGT) is performed on the output of the entropy source prior to seeding the FIPS Approved DRBG to provide health testing of the noise source. Power-on Self-tests are run during boot up. If any of these self-tests fail, the device enters an error state. At this point, a user must power the device down and restart to attempt to resolve the error.**

If there are instances when a shutdown does not occur, (e.g., a failure is deemed non- security relevant), the evaluator shall ensure that those cases are identified and a rationale is provided that supports the classification and justifies why the TOE's ability to enforce its security policies is not affected in any such instance.

**Evaluator Findings:**

There are no instances when a shutdown does not occur; hence, this activity is not applicable.

**Verdict:**

**PASS.**

**5.1.6.7.2 FPT\_FLS.1/SELFTEST AGD**

The evaluator shall verify that the operational guidance provides information on the self-test failures that can cause the TOE to shut down and how to diagnose the specific failure that has occurred, including possible remediation steps if available.

**Evaluator Findings:**

The evaluator checked the AGD '**Startup and Self-Test**' and ensured that it provides information on the self-test failures that can cause the TOE to shut down and how to diagnose the specific failure that has occurred, including possible remediation steps if available.

The relevant information is found in the following section(s): **Startup and Self-Test**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'If any of these tests fail, the product enters a hard error state, and the local console provides an error message reflecting information about the specific failure to the security administrator. Rebooting the product generally clears the errors. However, if errors persist, contact SonicWall Technical Support.'**

**Verdict:**

**PASS.**

**5.1.7 TOE ACCESS (FTA)**

**5.1.7.1 FTA\_SSL\_EXT.1 TSF-INITIATED SESSION LOCKING**

**5.1.7.1.1 FTA\_SSL\_EXT.1 TSS**

The evaluator shall examine the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings.

#### Evaluator Findings:

The evaluator examined the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings.

The relevant information is found in the following section(s): TOE Summary Specification **FTA\_SSL\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **Inactive local and remote sessions to the TOE are automatically terminated after a Security Administrator-configurable time interval between 1 and 9999 minutes.**

**By default, the TOE terminates a session after five minutes of inactivity. In addition, administrators are provided with the capability to terminate their own session.**

#### Verdict:

PASS.

#### 5.1.7.1.2 FTA\_SSL\_EXT.1 AGD

The evaluator shall confirm that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period.

#### Evaluator Findings:

The evaluator confirmed that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period.

The relevant information is found in the following section(s): **Configure Inactivity Time**

Upon investigation, the evaluator found that the claimed AGD section states that:

**Inactive local and remote sessions to the appliance are automatically terminated after a Security Administrator-configurable time interval.**

**To configure inactivity time for CLI:**

1. Login into the local console and type config to start the configuration session.
2. Type the command cli idle-timeout \* (where \* is the timeout value provided in minutes.)
3. To save the configuration, type commit.

#### Verdict:

PASS.

#### 5.1.7.2 FTA\_SSL.3 TSF-INITIATED TERMINATION

#### 5.1.7.2.1 FTA\_SSL.3 TSS

---

The evaluator shall examine the TSS to determine that it details the administrative remote session termination and the related inactivity time period.

##### **Evaluator Findings:**

The evaluator examined the TSS and determined that it details the administrative remote session termination and the related inactivity time period.

The relevant information is found in the following section(s): TOE Summary Specification **FTA\_SSL.3**.

Upon investigation, the evaluator found that the TSS states that: **Inactive local and remote sessions to the TOE are automatically terminated after a Security Administrator-configurable time interval between 1 and 9999 minutes.**

**By default, the TOE terminates a session after five minutes of inactivity. In addition, administrators are provided with the capability to terminate their own session.**

##### **Verdict:**

**PASS.**

#### 5.1.7.2.2 FTA\_SSL.3 AGD

---

The evaluator shall confirm that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination.

##### **Evaluator Findings:**

The evaluator confirmed that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination.

The relevant information is found in the following section(s): **Configure Inactivity Time**

Upon investigation, the evaluator found that the claimed AGD states that:

**Inactive local and remote sessions to the appliance are automatically terminated after a Security Administrator-configurable time interval.**

##### **To configure inactivity time for webUI:**

- 1. Navigate to Device | Settings | Administration.**
- 2. Click Login/Multiple Administrators.**
- 3. To specify the inactive time that can elapse before you are automatically logged out of the Management Interface, enter the time, in minutes, in the Log out the Admin after inactivity of (mins) field. By default, the SonicWall Security Appliance logs out the administrator after 5 minutes of inactivity. The inactivity timeout can range from 1 to 9999 minutes.**

**Verdict:**

**PASS.**

---

**5.1.7.3 FTA\_SSL.4 USER-INITIATED TERMINATION**

**5.1.7.3.1 FTA\_SSL.4 TSS**

The evaluator shall examine the TSS to determine that it details how the local and remote administrative sessions are terminated.

**Evaluator Findings:**

The evaluator examined the TSS and determined that it details how the remote administrative session (and if applicable the local administrative session) are terminated.

The relevant information is found in the following section(s): TOE Summary Specification **FTA\_SSL.4..**

Upon investigation, the evaluator found that the TSS states that: **Administrators are provided with the capability to terminate their own local or remote session using the instructions provided in the administrative guides.**

**Verdict:**

**PASS.**

---

**5.1.7.3.2 FTA\_SSL.4 AGD**

The evaluator shall confirm that the guidance documentation states how to terminate a local or remote interactive session.

**Evaluator Findings:**

The evaluator confirmed that the guidance documentation states how to terminate a remote interactive session (and if applicable the local administrative session).

The relevant information is found in the following section(s): **Logging Out**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'Logout occurs when the user actively ends the session by closing their session window or by using the Logout option provided on the session window.**

**To log out of the CLI, enter `logout`.'**

**Verdict:**

**PASS.**

---

**5.1.7.4 FTA\_TAB.1 DEFAULT TOE ACCESS BANNERS**

#### 5.1.7.4.1 FTA\_TAB.1 TSS

---

The evaluator shall check the TSS to ensure that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS).

**Evaluator Findings:**

The evaluator checked the TSS and ensured that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS).

The relevant information is found in the following section(s): TOE Summary Specification **FTA\_TAB.1**.

Upon investigation, the evaluator found that the TSS states that: **All access to the TOE takes place through the web-based management interface over HTTPS or the local serial console. The web-based management interface can be accessed using the GUI.**

The evaluator shall check the TSS to ensure that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file).

**Evaluator Findings:**

The evaluator checked the TSS and ensured that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file).

The relevant information is found in the following section(s): TOE Summary Specification **FTA\_TAB.1**.

Upon investigation, the evaluator found that the TSS states that: **All access to the TOE takes place through the web-based management interface over HTTPS or the local serial console. The web-based management interface can be accessed using the GUI.**

**All users, both local and remote, are presented with a Security Administrator-configured advisory notice and consent warning prior to TOE login.**

**Verdict:**

**PASS.**

#### 5.1.7.4.2 FTA\_TAB.1 AGD

---

The evaluator shall check the guidance documentation to ensure that it describes how to configure the banner message.



#### Evaluator Findings:

The evaluator examined the guidance documentation and ensured that it describes how to configure the banner message.

The relevant information is found in the following section(s): **Pre-Login Policy Banner**

Upon investigation, the evaluator found that the claimed AGD section states that:

**To create a pre-login policy banner:**

1. **Navigate to Device | Users | Settings.**
2. **Click Customization.**
3. **Scroll to the Pre-Login Policy Banner section.**
4. **In the Pre-Login Policy Banner section, select Start with policy banner before login page. This option is not selected by default.**
5. **In the Policy banner content field, enter your policy text. You can include HTML formatting. The page displayed includes an I Accept button and Cancel button for user confirmation.**
6. **Click Accept.**

**Verdict:**

**PASS.**

---

### 5.1.8 TRUSTED PATH (FTP)

---

#### 5.1.8.1 FTP\_ITC.1 INTER-TSF TRUSTED CHANNEL

---

##### 5.1.8.1.1 FTP\_ITC.1 TSS

The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint.

#### Evaluator Findings:

The evaluator examined the TSS and determined that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint.

The relevant information is found in the following section(s): TOE Summary Specification **FTP\_ITC.1.**

Upon investigation, the evaluator found that the TSS states that: **IPsec VPN tunnels are used to provide a trusted communication channel between the TOE and the external audit server and to support VPN communications.**

**The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel.**

**The TOE supports IKE version 2 in protecting these communications from disclosure and detecting modification.**

The evaluator shall also confirm that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST.

#### **Evaluator Findings:**

The evaluator also confirmed that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST.

The relevant information is found in the following section(s): TOE Summary Specification **FTP\_ITC.1**.

Upon investigation, the evaluator found that the TSS states that: **The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel.**

More information about the cryptographic protocol is described in **FCS\_IPSEC\_EXT.1** in the TSS.

#### **Verdict:**

**PASS.**

#### **5.1.8.1.2 FTP\_ITC.1 AGD**

The evaluator shall confirm that the guidance documentation contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken.

#### **Evaluator Findings:**

The evaluator confirmed that the AGD contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken.

The relevant information is found in the following section(s): **Configure VPN, Configuring the Syslog Settings, Reconnection**

Upon investigation, the evaluator found that the AGD section 'Configure VPN' and 'Configuring the Syslog Settings' states instructions for establishing the allowed protocol with IT entity.

#### To configure a VPN

1. Navigate to the NETWORK | IPsec VPN > Rules and Settings page.
2. Make the appropriate version selection, either IPv4 or IPv6.
3. Click +Add.
4. Complete the General, Network, Proposals, and Advanced tabs on the VPN Policy dialog.

#### To configure Syslog settings on your firewall

1. Navigate to Device > Log > Syslog page.
2. (Optional) If you selected Enhanced Syslog, click the Enhanced Syslog Fields Settings Configure icon. The Enhanced Syslog Field Settings pop-up dialog displays.  
(Optional) Select the Enhanced Syslog options to log. By default, all options are selected; the Host (sn) and Event ID (m) options are dimmed as they cannot be changed.
  - To select all options, click Enable All.
  - To deselect all options, click Disable All.
  - Select only some options, either: Click Disable All and select only those options to log. Or deselect only those options to not log.
3. Click Save.
4. Optionally, if you selected ArcSight, click the ARCSight CEF Fields Settings Configure icon. ArcSight CEF Fields Settings pop-up dialog displays.
5. Optionally, select the ArcSight options to log. By default, all options are selected; the Host and Event ID options are dimmed as they cannot be changed.
  - To select all options, click Enable All.
  - To deselect all options, click Disable All.
  - To select only some options, either Click Disable All and select only those options to log. Or deselect only those options to not log.
6. Click Save.
7. Optionally, select the Enable NDPP Enforcement for Syslog Server.
8. Optionally, select Display Syslog Timestamp in UTC.
9. Click Accept.

Furthermore, the evaluator found that the AGD section 'Reconnection' contains recovery instructions if a connection is unintentionally broken.

'If an IPsec tunnel loses connectivity for a long period of time, the device will attempt to re-connect 5 consecutive times. If the connectivity is restored within this period, no additional administrative actions are required. The tunnel will attempt to restart automatically. Plaintext data will never be sent. If the connectivity is not restored within the 5th reconnection attempt, the IPsec tunnel will be terminated and an security administrator will have to manually enable the tunnel.'

**Verdict:**

PASS.

5.1.8.2 FTP\_ITC.1/VPN INTER-TSF TRUSTED CHANNEL (VPN COMMUNICATIONS)

5.1.8.2.1 FTP\_ITC.1/VPN TSS

The EAs specified for FTP\_ITC.1 in the Supporting Document for the Base-PP shall be applied for IPsec VPN communications.

**Evaluator Findings:**

This is covered in section 5.1.8.1.1 above.

**Verdict:**

PASS.

5.1.8.2.1.1 FTP\_ITC.1/VPN AGD

The EAs specified for FTP\_ITC.1 in the Supporting Document for the Base-PP shall be applied for IPsec VPN communications.

**Evaluator Findings:**

This is covered in section 5.1.8.1.2 above.

**Verdict:**

PASS.

5.1.8.3 FTP\_TRP.1/ADMIN TRUSTED PATH

5.1.8.3.1 FTP\_TRP.1/ADMIN TSS

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

**Evaluator Findings:**

The evaluator examined the TSS and determined that the methods of remote TOE administration are indicated, along with how those communications are protected.

The relevant information is found in the following section(s): TOE Summary Specification **FTP\_TRP.1/Admin.**

Upon investigation, the evaluator found that the TSS states that: **HTTPS is used to provide a trusted path for communications between the TOE and the administrative interface.**

**The TOE supports TLS 1.2 to protect these communications from disclosure and detect modification.**

The evaluator shall also confirm that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

**Evaluator Findings:**

The evaluator also confirmed that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

The relevant information is found in the following section(s): TOE Summary Specification **FTP\_TRP.1/Admin.**

Upon investigation, the evaluator found that the TSS states that: **The TOE supports TLS 1.2 to protect these communications from disclosure and detect modification. All other protocol requests will be denied.**

**RSA with 2048-bits, 3072-bits, and 4096 bits keys is used in the supported TLS ciphersuites.**

**Verdict:**

**PASS.**

**5.1.8.3.2 FTP\_TRP.1/ADMIN AGD**

The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.

**Evaluator Findings:**

The evaluator confirmed that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS, Selecting a Security Certificate and Enforcing TLS Version**

Upon investigation, the evaluator summarized that the claimed AGD sections provide instructions for establishing remote administrative sessions for HTTPS and TLS.

**'The SonicWall appliance can be managed using HTTP or HTTPS and a Web browser. HTTP web-based management is disabled by default. Use HTTPS to log into the SonicOS Management Interface with factory default settings.**

**Security certificates provide data encryption and a secure website.**

**SonicOS supports versions 1.0, 1.1, and 1.2 of the Transport Layer Security (TLS) protocol. You should ensure that the more secure version 1.1 and above are used.'**

**Verdict:**

**PASS.**

---

## 5.1.9 FIREWALL (FFW)

---

### 5.1.9.1 FFW\_RUL\_EXT.1 STATEFUL TRAFFIC FILTERING

---

#### 5.1.9.1.1 FFW\_RUL\_EXT.1 TSS

The evaluator shall verify that the TSS provides a description of the TOE's initialization/startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process.

**Evaluator Findings:**

The evaluator reviewed the TSS section **FFW\_RUL\_EXT.1** row and verified that it provides a description of the TOE's initialization/startup process, which clearly indicates where processing of network packets begins to take place and provides a discussion that supports the assertion that packets cannot flow during this process.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**Packets are received by the SonicWall device on one of three Ethernet links: the LAN, WAN, or optional DMZ link. A flag called gStartupTrulyComplete is set after firewall bootup to identify when the network stack and the policy are ready to process packets. Before this flag is set to TRUE, firewall initializes the interfaces but set the interfaces to DOWN. Only after gStartupTrulyComplete is set to TRUE, TOE enables the interfaces.**

Furthermore, the evaluator examined the TSS section row **FFW\_RUL\_EXT.1** and found that it indicates where processing of network packets begins to take place. The TSS states that:

**Once the interfaces are enabled, the packets are analyzed in the communications stack at a level that is best described as above the Ethernet driver, but below the networking stack. During this analysis, packets are modified, dropped, passed up to the networking stack, or rewritten directly to another Ethernet link, as appropriate.**

The evaluator shall verify that the TSS also include a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describe the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure. This could include the failure of a component, such as a process being terminated, or a failure within a component, such as memory buffers full and cannot process packets.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it includes a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describes the safeguards that would prevent packets from flowing through the TOE without applying the ruleset in the event of a component failure.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator examined the TSS section row **FFW\_RUL\_EXT.1** and found that it identifies the components involved in processing the network packets and that it describe the safeguards preventing packets from flowing through the TOE without applying the ruleset in the event of a component failure (such as a process termination or failure within a component like memory buffers being full). The TSS states that:

**The packets are analyzed in the communications stack at a level that is best described as above the Ethernet driver, but below the networking stack. Transport-and application-layer data is also examined. This higher-level data is used to provide the stateful inspection security. During this analysis, packets are modified, dropped, passed up to the networking stack, or rewritten directly to another Ethernet link, as appropriate.**

**The SonicWall device acts as a single component. If the component fails, processing ceases and all traffic is stopped. If any component fails, packets will not be accepted into the connection cache, and will therefore not be allowed to flow through the device.**

**In the evaluated configuration, the default action is to DENY a packet. The TOE checks the incoming packet against all of the access rules. If the packet does not match any access rule and does not belong to an approved established connection, then the default action is to DENY the packet.**

**A global counter is used by the TOE to track the number of all half-open TCP connections. When this number reaches the value of Maximum Half Open TCP Connections, new incoming TCP connections are dropped.**

The description shall also include a description how the TOE behaves in the situation where the traffic exceeds the amount of traffic the TOE can handle and how it is ensured that also in this condition stateful traffic filtering rules are still applied so that traffic does not pass that shouldn't pass according to the specified rules.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it includes a description of how the TOE behaves in the situation where the traffic exceeds the amount of traffic the TOE can handle and how it is ensured that also in this condition stateful traffic filtering rules are still applied so that traffic does not pass that shouldn't pass according to the specified rules.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator examined the TSS section row **FFW\_RUL\_EXT.1** and found that it describes how the TOE behaves when traffic exceeds the amount the TOE can handle, and how stateful traffic filtering rules are still applied to ensure that traffic does not pass that shouldn't pass according to the specified rules. The TSS states that:

- **There is a TCP Handshake Timeout (seconds)**
  - **Each half-open TCP connection is removed if the handshake is not complete by the time this timeout is reached**
- **There is a maximum number of allowable Half Open TCP Connections**
  - **A global counter is used by the TOE to track the number of all half-open TCP connections. When this number reaches the value of Maximum Half Open TCP Connections, new incoming TCP connections are dropped.**
    - **If the component fails, processing ceases and all traffic is stopped.**

**If any component fails, packets will not be accepted into the connection cache, and will therefore not be allowed to flow through the device.**

**Verdict:**

**PASS.**

**5.1.9.1.2 FFW\_RUL\_EXT.1 AGD**

---

The guidance documentation associated with this requirement is assessed in the subsequent test evaluation activities.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation associated with this requirement and ensured that it is assessed in the subsequent test evaluation activities.



**Verdict:**

PASS.

**5.1.9.1.3 FFW\_RUL\_EXT.1.2/FFW\_RUL\_EXT.1.3/FFW\_RUL\_EXT.1.4 TSS**

---

The evaluator shall verify that the TSS describes a stateful packet filtering policy and the following attributes are identified as being configurable within stateful traffic filtering rules for the associated protocols:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source Address
  - Destination Address
  - Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and ensured that it describes a stateful packet filtering policy and the following attributes are identified as being configurable within stateful traffic filtering rules for the above associated protocols.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

The following RFCs are supported:

- **RFC 792 (ICMPv4)**
  - Type; and
  - Code
- **RFC 4443 (ICMPv6)**
  - Type; and
  - Code
- **RFC 791 (IPv4)**
  - Source address;
  - Destination Address; and
  - Transport Layer Protocol
- **RFC 8200 (IPv6)**
  - Source address;
  - Destination Address;
  - Transport Layer Protocol/Next Header
- **RFC 793 (TCP)**
  - Source Port; and
  - Destination Port
- **RFC 768 (UDP)**
  - Source Port; and
  - Destination Port

Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.

The Stateful packet filtering policy consists of the following rules and attributes.

- **Action: (Allow/Deny/Discard)**
  - Configure to permit or drop the packet
- **From: (Zone/Interface)**
  - Packet ingress point
- **To: (Zone/Interface)**
  - Packet egress point
- **Source Port: (Services Object)**
  - The protocol and the source port of the packet
- **Services: (Services Object)**

- The protocol and the destination port of the packet
  - Source: (Host/Range/Network)
  - Source IP: The source IP of the packet
  - Destination: (Host/Range/Network)
  - Destination IP: the Destination IP of the packet
  - Enable Logging (Checkbox)
  - Log the action when it is taking place
  - TCP Connection Inactivity Timeout (minutes)
  - UDP Connection Inactivity Timeout (seconds)

The attributes are all configurable for ICMPv4, ICMPv6, IPv4, IPv6, TCP and UDP policies. Logging can be configured for each access rule. The source and destination address are configurable for each access rule.

The evaluator shall verify that each rule can identify the following actions: permit or drop with the option to log the operation.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that each rule can identify the following actions: permit or drop with the option to log the operation.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Packet filtering rules are checked. If the packet matches an 'ALLOW' access rule, the connection cache is created. If the packet matches a 'DENY' rule, or there is no matched 'ALLOW' rule, the packet does not proceed.**

**Logging can be configured for each access rule.'**

The evaluator shall verify that the TSS identifies all interface types subject to the stateful packet filtering policy and explains how rules are associated with distinct network interfaces.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it identifies all interface types subject to the stateful packet filtering policy and explains how rules are associated with distinct network interfaces.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Packets are received by the SonicWall device on one of three Ethernet links: the LAN, WAN, or optional DMZ link.**

**The analysis is based on a set of rules entered by the firewall administrator which can be tied to the LAN, WAN, or optional DMZ links.**

**Verdict:**

**PASS.**

#### 5.1.9.1.4 FFW\_RUL\_EXT.1.2/FFW\_RUL\_EXT.1.3/FFW\_RUL\_EXT.1.4 AGD

The evaluators shall verify that the guidance documentation identifies the following attributes as being configurable within stateful traffic filtering rules for the associated protocols:

- ICMPv4
  - Type
  - Code
- ICMPv6
  - Type
  - Code
- IPv4
  - Source address
  - Destination Address
  - Transport Layer Protocol
- IPv6
  - Source Address
  - Destination Address
  - Transport Layer Protocol and where defined by the ST author, Extension Header Type, Extension Header Fields
- TCP
  - Source Port
  - Destination Port
- UDP
  - Source Port
  - Destination Port

#### **Evaluator Findings:**

The evaluator reviewed the guidance documentation '**Access Rules**' and confirmed that it identifies the following attributes as being configurable within stateful traffic filtering rules for the above associated protocols.

The relevant information is found in the following section(s): **Access Rules**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'Rules may be applied to various types of traffic including,**

- **Internet Control Message Protocol version 4 (ICMPv4) (Type, Code): RFC 792**
- **Internet Control Message Protocol version 6 (ICMPv6) (Type, Code): RFC 4443**
- **Internet Protocol (IPv4) (Source address, Destination Address, Transport Layer Protocol): RFC 791**
- **Internet Protocol version 6 (IPv6) (Source Address, Destination Address, Transport Layer Protocol): RFC 8200**
- **Transmission Control Protocol (TCP) (Source Port, Destination Port): RFC 793**
- **User Datagram Protocol (UDP) (Source Port, Destination Port): RFC 768 '**

The evaluator shall verify that the guidance documentation indicates that each rule can identify the following actions: permit, drop, and log.

#### **Evaluator Findings:**

The evaluator reviewed the guidance documentation section **'Adding Access Rules'** and confirmed that it indicates that each rule can identify the following actions: permit, drop, and log.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that **point numbers 5 and 20** in the claimed AGD section states that each rule can identify the following actions: permit, drop, and log.

**5. 'Select an Action, that is, how the rule processes (permits or blocks) the specified IP traffic:**

- **Allow (default): As long as the Enable option is selected, your access rule is active and permits the traffic.**
- **Deny: The firewall denies all connections matching this rule and blocks the page specified and the action profile is served for web traffic. The firewall also resets the connections on both sides.**
- **Discard: Firewall silently drops any packets matching this rule.**

**20. To enable logging for this rule, select Logging.'**

The evaluator shall verify that the guidance documentation explains how rules are associated with distinct network interfaces.

#### **Evaluator Findings:**

The evaluator reviewed the guidance documentation section **'Adding Access Rules'** and confirmed that it explains how rules are associated with distinct network interfaces.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that **point number 09, 10 and 11** in the claimed AGD section explains how rules are associated with distinct network interfaces. The AGD states that:  
**'09. Select the source and destination Zone/Interface from the drop-down menus.**  
**10. Select from the Predefined zones WAN, LAN, DMZ, VPN, MULTICAST, WLAN, and SSLVPN. In addition to predefined zones, custom user-friendly zones can also be configured in Sonicwall, with different security types.**  
**11. Select an interface from the range X0–X33.**  
**The number of physical interfaces varies depending on the firewall model.'**

**Verdict:**

**PASS.**

**5.1.9.1.5 FFW\_RUL\_EXT.1.5 TSS**

The evaluator shall verify that the TSS identifies the protocols that support stateful session handling. The TSS shall identify TCP, UDP, and, if selected by the ST author, also ICMP.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it identifies the protocols that support stateful session handling.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:  
**'Stateful session handling is supported for TCP and UDP.'**

The evaluator shall verify that the TSS describes how stateful sessions are established (including handshake processing) and maintained.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it describes how stateful sessions are established (including handshake processing) and maintained.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:  
**'Source and destination addresses, and source and destination ports are used together to recognize TCP flow in support of stateful session handling. Sequence numbers are used to ensure that the**

received data falls within the window defined for the protocol. Flags are used to track the connection against the defined TCP State Machine states:

- Listen State: Only a TCP packet with just the SYN flag is considered valid.
- Syn-Sent State:
  - ACK number (if present) must be valid.
  - RST packet (with a valid TCP ACK number) is valid.
  - FIN packet (which does not have the SYN bit set) is also considered valid.
- Syn-Received, Established, Fin-Sent, and Fin-Acked States:
  - SEQ number must be within the TCP window for the destination or be that for Keep-Alive packet.
  - RST packet (with a valid TCP SEQ number) is valid.
  - ACK number must also be present and valid in this state.
  - A SYN seen in this state will cause the TCP connection to be closed.
- Close-Wait State:
  - A SYN is valid (to re-open the same TCP connection).
  - Any other packet which is also valid in the previous state is acceptable.

For UDP, source and destination addresses, and source and destination ports are used together to be checked to match with an access rule. Following a UDP request, the TOE will accept return packets for a configurable period of time. This is generally in the order of several seconds and is configurable as the UDP Timeout in the applicable access rule.

The evaluator shall verify that for TCP, the TSS identifies and describes the use of the following attributes in session determination: source and destination addresses, source and destination ports, sequence number, and individual flags.

#### Evaluator Findings:

The evaluator reviewed the TSS and verified that for TCP, it identifies and describes the use of the following attributes in session determination: source and destination addresses, source and destination ports, sequence number, and individual flags.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Source and destination addresses, and source and destination ports are used together to recognize TCP flow in support of stateful session handling. Sequence numbers are used to ensure that the**

received data falls within the window defined for the protocol. Flags are used to track the connection against the defined TCP State Machine states:

- Listen State: Only a TCP packet with just the SYN flag is considered valid.
- Syn-Sent State:
  - ACK number (if present) must be valid.
  - RST packet (with a valid TCP ACK number) is valid.
  - FIN packet (which does not have the SYN bit set) is also considered valid.
- Syn-Received, Established, Fin-Sent, and Fin-Acked States:
  - SEQ number must be within the TCP window for the destination or be that for Keep-Alive packet.
  - RST packet (with a valid TCP SEQ number) is valid.
  - ACK number must also be present and valid in this state.
  - A SYN seen in this state will cause the TCP connection to be closed.
- Close-Wait State:
  - A SYN is valid (to re-open the same TCP connection).
  - Any other packet which is also valid in the previous state is acceptable.

The evaluator shall verify that for UDP, the TSS identifies and describes the following attributes in session determination: source and destination addresses, source and destination ports.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that for UDP, it identifies and describes the following attributes in session determination: source and destination addresses, source and destination ports.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'For UDP, source and destination addresses, and source and destination ports are used together to be checked to match with an access rule. Following a UDP request, the TOE will accept return packets for a configurable period of time. This is generally in the order of several seconds and is configurable as the UDP Timeout in the applicable access rule.'**

The evaluator shall verify that for ICMP (if selected), the TSS identifies and describes the following attributes in session determination: source and destination addresses, other attributes chosen in FFW\_RUL\_EXT.1.5.

**Evaluator Findings:**

ICMP is not supported by the TOE; hence, this activity is not applicable.



The evaluator shall verify that the TSS describes how established stateful sessions are removed.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it describes how established stateful sessions are removed.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Stateful sessions are removed when complete, or when the timeout is triggered.'**

The TSS shall describe how connections are removed for each protocol based on normal completion and/or timeout conditions.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it describes how connections are removed for each protocol based on normal completion and/or timeout conditions.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**Following a UDP request, the TOE will accept return packets for a configurable period of time. This is generally in the order of several seconds and is configurable as the UDP Timeout in the applicable access rule.**

**For TCP connection completion, the connection is closed in one of two ways:**

- **Syn-Sent State**
  - **A validated RST will cause the action of the TCP connection to be closed.**
- **Syn-Received, Established, Fin-Sent, Fin-Acked, and Close-Wait States**
  - **A validated RST will cause the action of the TCP connection to be closed.**
  - **Acknowledged TCP FINs will cause the action of the TCP connection to be closed.'**

The TSS shall also indicate when session removal becomes effective (e.g., before the next packet that might match the session is processed).

#### Evaluator Findings:

The evaluator reviewed the TSS and verified that it indicates when session removal becomes effective (e.g., before the next packet that might match the session is processed).

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Session removal becomes effective immediately after Connection cache is removed.**

**Each packet flow through the TOE triggers a timestamp update to its connection cache. The TOE checks this timestamp, and if the connection cache timeout has been reached, the session is removed.'**

#### Verdict:

PASS.

#### 5.1.9.1.6 FFW\_RUL\_EXT.1.5 AGD

The evaluator shall verify that the guidance documentation describes stateful session behaviours. For example, a TOE might not log packets that are permitted as part of an existing session.

#### Evaluator Findings:

The evaluator reviewed the guidance documentation **'About Stateful Packet Inspection Default Access'** and verified that it describes stateful session behaviours.

The relevant information is found in the following section(s): **About Stateful Packet Inspection Default Access**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'By default, the SonicWall network security appliance's stateful packet inspection allows all communication from the LAN to the Internet and blocks all traffic to the LAN from the Internet. The following behaviors are defined by the Default stateful inspection packet access rule enabled on the SonicWall network security appliance:**

- **Allow all sessions originating from the LAN, WLAN to the WAN, or DMZ (except when the destination WAN IP address is the WAN interface of the firewall itself)**
- **Allow all sessions originating from the DMZ to the WAN.**
- **Deny all sessions originating from the WAN to the DMZ.**
- **Deny all sessions originating from the WAN and DMZ to the LAN or WLAN.**

**SonicWall monitors the initiation sequence, typically a TCP three-way handshake, and records the packet's state: open, established, or closed. Each packet transferred across the network is examined, and its headers and flags are compared against the state table. If the packet is part of an**

**existing, approved connection, it is allowed to pass. If not, the stateful inspection firewall consults its rule set to determine the appropriate action.'**

**Verdict:**

**PASS.**

**5.1.9.1.7 FFW\_RUL\_EXT.1.6 TSS**

The evaluator shall verify that the TSS identifies the following as packets that will be automatically dropped and are counted or logged:

- a) Packets which are invalid fragments, including a description of what constitutes an invalid fragment
- b) Fragments that cannot be completely re-assembled
- c) Packets where the source address is defined as being on a broadcast network
- d) Packets where the source address is defined as being on a multicast network
- e) Packets where the source address is defined as being a loopback address
- f) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4;
- g) The TSF shall reject and be capable of logging network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6;
- h) Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified
- i) Other packets defined in FFW\_RUL\_EXT.1.6 (if any)

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it identifies the following as packets that will be automatically dropped and are counted or logged:

- a) Packets which are invalid fragments, including a description of what constitutes an invalid fragment.
- b) Fragments that cannot be completely re-assembled.
- c) Packets where the source address is defined as being on a broadcast network.
- d) Packets where the source address is defined as being on a multicast network.
- e) Packets where the source address is defined as being a loopback address.
- f) The TSF rejects and is capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e., 0.0.0.0) or an address "reserved for future use" (i.e., 240.0.0.0/4) as specified in RFC 5735 for IPv4.

- g) The TSF rejects and is capable of logging network packets where the source or destination address of the network packet is defined as an “unspecified address” or an address “reserved for future definition and use” (i.e., unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6.
- h) Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified.
- i) Other packets defined in FFW\_RUL\_EXT.1.6 (if any).

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**The TOE will automatically drop and log the event when the following is found:**

- **A packet is found to be an invalid fragment. A fragment is determined to be invalid if it cannot be combined with other fragments to form a packet. The offset may be incorrect, or it may be considered to be too small**
- **A fragmented packet cannot be completely re-assembled**
- **A packet with a source address that is defined as being on a broadcast network**
- **A packet with a source address that is defined as being on a multicast network**
- **A packet with a source address that is defined as being a loopback address**
- **A packet with a source or destination address that is defined as unspecified or reserved for future use as specified in RFC 5735 for IPv4**
- **A packet with a source or destination address that is defined as an unspecified address or an address reserved for future definition and use as specified in RFC 3513 for IPv6**
- **A packet with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified**

**Verdict:**

**PASS.**

#### 5.1.9.1.8 FFW\_RUL\_EXT.1.6 AGD

The evaluator shall verify that the guidance documentation describes packets that are discarded and potentially logged by default. If applicable protocols are identified, their descriptions need to be consistent with the TSS.

#### **Evaluator Findings:**

The evaluator reviewed the guidance documentation **Enabling NDPP Mode** and verified that it describes packets that are discarded and potentially logged by default.

The relevant information is found in the following section(s): **Enabling NDPP Mode**

Upon investigation, the evaluator found that the claimed AGD section states that:  
**'Once NDPP compliance is enabled, the following packets are discarded and logged by default, without any additional configuration changes or access rules.**

- **Packets which are invalid fragments.**
- **Fragmented packets which cannot be re-assembled completely.**
- **Packets where the source address of the network packet is defined as being on a broadcast network.**
- **Packets where the source address of the network packet is defined as being on a multicast network.**
- **Network packets where the source address of the network packet is defined as being a loopback address.**
- **Network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4.**
- **Network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6.**
- **Network packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified and no other rules.'**

If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.

<b>Evaluator Findings:</b>
Logging is not configurable and is enabled by default. ; hence, this activity is not applicable.

**Verdict:**

**PASS.**

#### 5.1.9.1.9 FFW\_RUL\_EXT.1.7 TSS

The evaluator shall verify that the TSS explains how the following traffic can be dropped and counted or logged:

- a) Packets where the source address is equal to the address of the network interface where the network packet was received
- b) Packets where the source or destination address of the network packet is a link-local address
- c) Packets where the source address does not belong to the networks associated with the network interface where the network packet was received, including a description of how the TOE determines whether a source address belongs to a network associated with a given network interface

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it explains how the following traffic can be dropped and counted or logged:

- a) Packets where the source address is equal to the address of the network interface where the network packet was received.
- b) Packets where the source or destination address of the network packet is a link-local address.
- c) Packets where the source address does not belong to the networks associated with the network interface where the network packet was received, including a description of how the TOE determines whether a source address belongs to a network associated with a given network interface.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**The TOE will automatically drop and log the event when the following is found:**

- **Packets where the source address is equal to the address of the network interface where the network packet was received**
- **Packets where the source or destination address of the network packet is a link-local address**
- **Packets where the source address is not identified by the routing table as a network associated with the network interface the packet was received**

**Verdict:**

**PASS.**

**5.1.9.1.10 FFW\_RUL\_EXT.1.7 AGD**

The evaluator shall verify that the guidance documentation describes how the TOE can be configured to implement the required rules.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation **Enabling NDPP Mode** and verified that it describes how the TOE can be configured to implement the required rules.

The relevant information is found in the following section(s): **Enabling NDPP Mode**

Upon investigation, the evaluator found that the claimed AGD section states that:  
**'Once NDPP compliance is enabled, the following packets are discarded and logged by default, without any additional configuration changes or access rules.**

- **Packets where the source address of the network packet is equal to the address of the network interface where the network packet was received.**
- **Packets where the source or destination address of the network packet is a link-local address.**
- **Packets where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received.'**

If logging is configurable, the evaluator shall verify that applicable instructions are provided to configure auditing of automatically rejected packets.

**Evaluator Findings:**

Logging is not configurable and is enabled by default; hence, this activity is not applicable.

**Verdict:**

**PASS.**

**5.1.9.1.11 FFW\_RUL\_EXT.1.8 TSS [TD0545]**

The evaluator shall verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of an administrator-defined and ordered ruleset.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**The algorithm applied to incoming packets performs the following actions:**

- **In the evaluated configuration, the default action is to DENY a packet. The TOE checks the incoming packet against all of the access rules. If the packet does not match any access rule and does not belong to an approved established connection, then the default action is to DENY the packet.**
- **The TOE performs a Connection cache lookup**
  - **each connection cache represents an established session**

- For incoming packets, srcIp, dstIp, srcPort, dstPort, ipType are used together as a hash index to find the matched connection cache
- An access rule check is performed if the connection cache lookup fails
- The TOE performs an access rule check only if the connection cache lookup fails. The following rules are applied in an access rule check:
  - Access rules are ordered by Priority. The rule with higher Priority will be applied
  - For incoming packets, srcZone, dstZone, srcIp, dstIp, srcPort, dstPort, ipType are used together as a hash index to find the matching access rule
  - If an incoming packet matches an access rule with the ALLOW action, a new connection cache is added. Otherwise the packet is dropped

In the evaluated configuration, the default action is to DENY a packet if the packet does not match any of the access rules. However, this does not apply for dynamic protocol traffic.

[TD0545] If the TOE implements a mechanism that ensures that no conflicting rules can be configured, the TSS shall describe the underlying mechanism.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified that it includes a description of the mechanism that ensures no conflicting rules can be configured. The TSS states that:

**The TOE does not allow configuring conflicting rules. To identify conflicting rules, the TOE will validate the action (allow or deny), source and destination IP addresses, protocols, services, and source and destination interfaces/zones attributes of the rules at the time of creation. If any conflict is detected, the rule is not allowed to be created and an error will be displayed indicating that a conflicting rule is present.**

**Verdict:**

**PASS.**

**5.1.9.1.12 FFW\_RUL\_EXT.1.8 AGD**

The evaluator shall verify that the guidance documentation describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation '**Adding Access Rules**' and verified that it describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.



The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that AGD section '**Adding Access Rules**' describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing. The **point number 7** in the claimed section states that:

**'7. Set your access rule's Priority. You can choose to Auto Prioritize, Insert at the End, or a Manual priority for your access rule.**

**Higher numbers indicate lower priority. The lowest priority rule is the final/default rule applied to matching traffic (traffic matching the defined attributes) when no higher priority rules apply. Lower priority rules should be more general than rules with higher priorities. If a higher priority rule does not match all the attributes, then the next rule is evaluated to see if it applies, all the way down the list of rules. Rules with more specific matching attributes need to be set at a higher priority or else a more general rule could match before that specific rule is evaluated.**

**When you added a new Access Rule, the rule module decided where to place it in the Access Rule table. The rule module uses an Auto Prioritize algorithm that places the most specific rules at the top. The only way to change the priority was to manually edit the rule and then provide the index of where to place it. Finding the rule in a large table to edit it can be difficult.**

**The User Priority for Access Rules provides two choices for the priority types of the new rule:**

- **Auto Prioritize, which uses the Auto Prioritize algorithm that places the most specific rules on the top of the Access Rules table. This is the default choice.**
- **Insert at the end, which indicates to the rule module to place the rule at the end of the Access Rules table, and as a result, makes the new rule easy to locate regardless of the size of the table.**

**Verdict:**

**PASS.**

**5.1.9.1.13 FFW\_RUL\_EXT.1.9 TSS**

The evaluator shall verify that the TSS describes the process for applying stateful traffic filtering rules and also that the behavior (either by default, or as configured by the administrator) is to deny packets when there is no rule match unless another required conditions allows the network traffic (i.e., FFW\_RUL\_EXT.1.5 or FFW\_RUL\_EXT.2.1).

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it describes the process for applying stateful traffic filtering rules and that the behavior (either by default or as configured by the administrator) is to deny packets when there is no rule match unless another required condition allows the network traffic (i.e., FFW\_RUL\_EXT.1.5 or FFW\_RUL\_EXT.2.1).

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

- **'The TOE performs an access rule check only if the connection cache lookup fails. The following rules are applied in an access rule check:**
  - **Access rules are ordered by Priority. The rule with higher Priority will be applied**
  - **For incoming packets, srcZone, dstZone, srcIp, dstIp, srcPort, dstPort, ipType are used together as a hash index to find the matching access rule**
  - **If an incoming packet matches an access rule with the ALLOW action, a new connection cache is added. Otherwise the packet is dropped**

**In the evaluated configuration, the default action is to DENY a packet if the packet does not match any of the access rules. However, this does not apply for dynamic protocol traffic.'**

**Verdict:**

**PASS.**

#### 5.1.9.1.14 FFW\_RUL\_EXT.1.9 AGD

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

##### **Evaluator Findings:**

The evaluator reviewed the guidance documentation '**Default Deny Rule**' and verified that it describes the behavior if no rules or special conditions apply to the network traffic.

The relevant information is found in the following section(s): **Default Deny Rule**

Upon investigation, the evaluator found that AGD section '**Default Deny Rule**' states that: **'In the evaluated configuration, a deny rule applied to any interface, any zone, and for any traffic with the lowest priority must be created. This ensures that any traffic that does not match a configured rule will be denied.'**

If the behavior is configurable, the evaluator shall verify that the guidance documentation provides the appropriate instructions to configure the behavior to deny packets with no matching rules.

##### **Evaluator Findings:**

The evaluator reviewed the guidance documentation and verified that it provides the appropriate instructions to configure the behavior to deny packets with no matching rules.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator concluded that the relevant instructions in the specified AGD section allow for configuring Deny rules for any interface and any zone.

**Verdict:**

**PASS.**

**5.1.9.1.15 FFW\_RUL\_EXT.1.10 TSS**

The evaluator shall verify that the TSS describes how the TOE tracks and maintains information relating to the number of half-open TCP connections.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FFW\_RUL\_EXT.1** row, and verified that it describes how the TOE tracks and maintains information relating to the number of half-open TCP connections.

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'The TOE tracks and maintains information relating to the number of half-open TCP connections as follows:**

- **There is an administratively defined limit for half-open TCP connections based on:**
  - **TCP Handshake Timeout (seconds)**
  - **Maximum Half Open TCP Connections'**

The TSS should identify how the TOE behaves when the administratively defined limit is reached and should describe under what circumstances stale half-open connections are removed (e.g. after a timer expires).

**Evaluator Findings:**

The evaluator verified that the TSS identifies how the TOE behaves when the administratively defined limit is reached and describes under what circumstances stale half-open connections are removed (e.g., after a timer expires).

The relevant information is found in the following section(s): TOE Summary Specification row **FFW\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

- **There is a TCP Handshake Timeout (seconds)**

- Each half-open TCP connection is removed if the handshake is not complete by the time this timeout is reached
- There is a maximum number of allowable Half Open TCP Connections
  - A global counter is used by the TOE to track the number of all half-open TCP connections. When this number reaches the value of Maximum Half Open TCP Connections, new incoming TCP connections are dropped.

**Verdict:**

**PASS.**

**5.1.9.1.16 FFW\_RUL\_EXT.1.10 AGD**

The evaluator shall verify that the guidance documentation describes the behaviour of imposing TCP half-open connection limits and its default state if unconfigured.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation ‘**TCP Connection**’ and verified that it describes the behavior of imposing TCP half-open connection limits and its default state if unconfigured.

The relevant information is found in the following section(s): **TCP Connection**

Upon investigation, the evaluator found that AGD section ‘**TCP Connection**’ states that:

**‘The appliance tracks and maintains information relating to the number of half-open TCP connections as follows:**

- There is an administratively defined limit for half-open TCP connections based on:
  - TCP Handshake Timeout (seconds). This is 1500 seconds by default.
  - Maximum Half Open TCP Connections. This is 2000 by default when enabled.
- There is a TCP Handshake Timeout (seconds)
  - Each half-open TCP connection is removed if the handshake is not complete by the time this timeout is reached. This is 15 minutes by default.

**To change TCP settings**

1. Navigate to Network | Firewall | Flood Protection.
2. Adjust settings as needed.
3. Click on Accept.’

The evaluator shall verify that the guidance clearly indicates the conditions under which new connections will be dropped e.g. per-destination or per-client.

<b>Evaluator Findings:</b>
<p>The evaluator reviewed the guidance documentation and verified that it clearly indicates the conditions under which new connections will be dropped (e.g., per-destination or per-client).</p> <p>The relevant information is found in the following section(s): <b>TCP Connection</b></p> <p>Upon investigation, the evaluator found that AGD section ‘<b>TCP Connection</b>’ states that:</p> <ul style="list-style-type: none"><li>• <b>Enable Half Open TCP Connections Threshold–This option denies any new TCP connections if the threshold of overall TCP half-open connections has been reached. By default, the half-open TCP connection is not monitored, so this option is not selected by default.</b></li></ul>

**Verdict:**

**PASS.**

---

#### 5.1.10 PACKET FILTERING (FPF)

---

##### 5.1.10.1 FPF\_RUL\_EXT.1 PACKET FILTERING RULES TSS

---

###### 5.1.10.1.1 FPF\_RUL\_EXT.1.1 TSS

The evaluator shall verify that the TSS provide a description of the TOE’s initialization and startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process.

<b>Evaluator Findings:</b>
<p>The evaluator reviewed the TSS, section ‘<b>FPF_RUL_EXT.1</b>’ row and ensured that it provides a description of the TOE’s initialization and startup process, which clearly indicates where processing of network packets begins to take place, and provides a discussion that supports the assertion that packets cannot flow during this process.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification row <b>FPF_RUL_EXT.1</b></p> <p>Upon investigation, the evaluator found that the TSS states that:</p> <p><b>Packets are received by the SonicWall device on one of three Ethernet links: the LAN, WAN, or optional DMZ link. A flag called gStartupTrulyComplete is set after firewall bootup to identify when the network stack and the policy are ready to process packets. Before this flag is set to TRUE,</b></p>

**firewall initializes the interfaces but set the interfaces to DOWN. Only after gStartupTrulyComplete is set to TRUE, TOE enables the interfaces. Once the interfaces are enabled, the packets are analyzed in the communications stack at a level that is best described as above the Ethernet driver, but below the networking stack. Transport-and application-layer data is also examined. This higher-level data is used to provide the stateful inspection security.**

The evaluator shall verify that the TSS also includes a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describes the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure. This could include the failure of a component, such as a process being terminated, or a failure within a component, such as memory buffers full and cannot process packets.

#### **Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it includes a narrative that identifies the components (e.g., active entity such as a process or task) involved in processing the network packets and describes the safeguards that would prevent packets flowing through the TOE without applying the ruleset in the event of a component failure.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator examined the TSS section row **FPF\_RUL\_EXT.1** and found that it identifies the components involved in processing the network packets and that it describe the safeguards preventing packets from flowing through the TOE without applying the ruleset in the event of a component failure (such as a process termination or failure within a component like memory buffers being full). The TSS states that:

**The packets are analyzed in the communications stack at a level that is best described as above the Ethernet driver, but below the networking stack. Transport-and application-layer data is also examined. This higher-level data is used to provide the stateful inspection security.**

**During this analysis, packets are modified, dropped, passed up to the networking stack, or rewritten directly to another Ethernet link, as appropriate.**

**The SonicWall device acts as a single component. If the component fails, processing ceases and all traffic is stopped.**

**If any component fails, packets will not be accepted into the connection cache, and will therefore not be allowed to flow through the device.**

**In the evaluated configuration, the default action is to DENY a packet. The TOE checks the incoming packet against all of the access rules. If the packet does not match any access rule and does not belong to an approved established connection, then the default action is to DENY the packet.**

**A global counter is used by the TOE to track the number of all half-open TCP connections. When this number reaches the value of Maximum Half Open TCP Connections, new incoming TCP connections are dropped.**

**Verdict:**

**PASS.**

**5.1.10.1.2 FPF\_RUL\_EXT.1.1 AGD**

---

The operational guidance associated with this requirement is assessed in the subsequent test EAs.

**5.1.10.1.3 FPF\_RUL\_EXT.1.2**

---

There are no EAs specified for this element. Definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces is described collectively under FPF\_RUL\_EXT.1.4.

**5.1.10.1.4 FPF\_RUL\_EXT.1.3**

---

There are no EAs specified for this element. Definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces is described collectively under FPF\_RUL\_EXT.1.4.

**5.1.10.1.5 FPF\_RUL\_EXT.1.4 TSS**

---

The evaluator shall verify that the TSS describes a packet filtering policy that can use the following fields for each identified protocol, and that the RFCs identified for each protocol are supported:

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

### **Evaluator Findings:**

The evaluator reviewed the TSS, section **FPF\_RUL\_EXT.1** row, and ensured that it describes a packet filtering policy that can use the following fields for each identified protocol, and that the RFCs identified for each protocol are supported.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**The following RFCs are supported:**

- **RFC 791 (IPv4)**
  - **Source address;**
  - **Destination Address; and**
  - **Transport Layer Protocol**
- **RFC 8200 (IPv6)**
  - **Source address;**
  - **Destination Address;**
  - **Transport Layer Protocol/Next Header**
- **RFC 793 (TCP)**
  - **Source Port; and**
  - **Destination Port**
- **RFC 768 (UDP)**
  - **Source Port; and**
  - **Destination Port**

**The Stateful packet filtering policy consists of the following rules and attributes.**

- **Action: (Allow/Deny/Discard)**
  - **Configure to permit or drop the packet**
- **From: (Zone/Interface)**
  - **Packet ingress point**
- **To: (Zone/Interface)**
  - **Packet egress point**



- **Source Port: (Services Object)**
  - The protocol and the source port of the packet
- **Services: (Services Object)**
  - The protocol and the destination port of the packet
- **Source: (Host/Range/Network)**
- **Source IP: The source IP of the packet**
- **Destination: (Host/Range/Network)**
- **Destination IP: the Destination IP of the packet**
- **Enable Logging (Checkbox)**
- **Log the action when it is taking place**
- **TCP Connection Inactivity Timeout (minutes)**
- **UDP Connection Inactivity Timeout (seconds)**

The evaluator shall verify that the TSS describes how conformance with the identified RFCs has been determined by the TOE developer (e.g., third party interoperability testing, protocol compliance testing).

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes how conformance with the identified RFCs has been determined by the TOE developer. The TOE states that:

**Conformance to these RFCs is demonstrated by protocol compliance testing by the product QA team.**

The evaluator shall verify that each rule can identify the following actions: permit, discard, and log.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that each rule identifies the following actions: permit, discard, and log.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Packet filtering rules are checked. If the packet matches an 'ALLOW' access rule, the connection cache is created. If the packet matches a 'DENY' rule, or there is no matched 'ALLOW' rule, the packet does not proceed.**

**Logging can be configured for each access rule.'**

The evaluator shall verify that the TSS identifies all interface types subject to the packet filtering policy and explains how rules are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used), they can be treated collectively as a distinct network interface.

#### Evaluator Findings:

The evaluator reviewed the TSS and ensured that it identifies all interface types subject to the packet filtering policy and explains how rules are associated with distinct network interfaces.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**'Packets are received by the SonicWall device on one of three Ethernet links: the LAN, WAN, or optional DMZ link.**

**The analysis is based on a set of rules entered by the firewall administrator which can be tied to the LAN, WAN, or optional DMZ links, despite the interfaces being standalone or grouped via link aggregation.'**

#### Verdict:

**PASS.**

#### 5.1.10.1.6 FPF\_RUL\_EXT.1.4 AGD

---

The evaluator shall verify that the operational guidance identifies the following protocols as being supported and the following attributes as being configurable within packet filtering rules for the associated protocols:

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

- o destination port

Evaluator Findings:
<p>The evaluator checked the AGD section <b>Access Rules</b> and ensured that it identifies the following protocols as being supported and the following attributes as being configurable within packet filtering rules for the associated protocols.</p> <p>The AGD states that:</p> <p><b>'Rules may be applied to various types of traffic including,</b></p> <ul style="list-style-type: none"> <li>• <b>Internet Protocol (IPv4) (Source address, Destination Address, Transport Layer Protocol): RFC 791</b></li> <li>• <b>Internet Protocol version 6 (IPv6) (Source Address, Destination Address, Transport Layer Protocol): RFC 8200</b></li> <li>• <b>Transmission Control Protocol (TCP) (Source Port, Destination Port): RFC 793</b></li> <li>• <b>User Datagram Protocol (UDP) (Source Port, Destination Port): RFC 768'</b></li> </ul>

The evaluator shall verify that the operational guidance indicates that each rule can identify the following actions: permit, discard, and log.

Evaluator Findings:
<p>The evaluator reviewed the guidance documentation section <b>'Adding Access Rules'</b> and confirmed that it indicates that each rule can identify the following actions: permit, drop, and log.</p> <p>The relevant information is found in the following section(s): <b>Adding Access Rules</b></p> <p>Upon investigation, the evaluator found that <b>point numbers 5 and 20</b> in the claimed AGD section states that each rule can identify the following actions: permit, drop, and log.</p> <p><b>5. 'Select an Action, that is, how the rule processes (permits or blocks) the specified IP traffic:</b></p> <ul style="list-style-type: none"> <li>• <b>Allow (default): As long as the Enable option is selected, your access rule is active and permits the traffic.</b></li> <li>• <b>Deny: The firewall denies all connections matching this rule and blocks the page specified and the action profile is served for web traffic. The firewall also resets the connections on both sides.</b></li> <li>• <b>Discard: Firewall silently drops any packets matching this rule.</b></li> </ul> <p><b>20. To enable logging for this rule select Logging.'</b></p>

The evaluator shall verify that the operational guidance explains how rules are associated with distinct network interfaces. The guidance may describe the other protocols contained within the ST (e.g., IPsec, IKE, potentially HTTPS, SSH, and TLS) that are processed by the TOE.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation section ‘**Adding Access Rules**’ and confirmed that it explains how rules are associated with distinct network interfaces.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that **point numbers 9, 10 and 11** in the claimed AGD section explains how rules are associated with distinct network interfaces. The AGD states that:

- **9. Select the source and destination Zone/Interface from the drop-down menus.**
- **10. Select from the Predefined zones WAN, LAN, DMZ, VPN, MULTICAST, WLAN, and SSLVPN. In addition to predefined zones, custom user-friendly zones can also be configured in Sonicwall, with different security types.**
- **11. Select an interface from the range X0–X33.**

**The number of physical interfaces varies depending on the firewall model.**

There are many IP Protocols, TCP Protocols, and UDP Protocols supported by the TOE. The AGD describes most of these in various sections. Examples: List of IP Protocols in section “**Adding Access Rules**”, a list of various traffic types in section “**Access Rules**”, etc’

The evaluator shall ensure that it is made clear what protocols were not considered as part of the TOE evaluation.

**Evaluator Findings:**

The evaluator checked the AGD section “**Access Rules**”and ensured that it is clear what protocols were considered as part of the TOE evaluation. The AGD states that:

**Note: There are multiple other protocols that are supported by Sonicwall firewalls. All IPv4 and IPv6 protocols are supported. However, not all protocols were evaluated during the Common Criteria testing. Every protocol that is mentioned above were evaluated.**

**Verdict:**

**PASS.**

**5.1.10.1.7 FPF\_RUL\_EXT.1.5 TSS**

The evaluator shall verify that the TSS describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.

#### Evaluator Findings:

The evaluator reviewed the TSS, **FPF\_RUL\_EXT.1** row, and ensured that it describes the algorithm applied to incoming packets, including the processing of default rules, determination of whether a packet is part of an established session, and application of administrator defined and ordered ruleset.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

**The algorithm applied to incoming packets performs the following actions:**

- **In the evaluated configuration, the default action is to DENY a packet. The TOE checks the incoming packet against all of the access rules. If the packet does not match any access rule and does not belong to an approved established connection, then the default action is to DENY the packet.**
- **The TOE performs a Connection cache lookup**
  - each connection cache represents an established session
  - For incoming packets, srcIp, dstIp, srcPort, dstPort, ipType are used together as a hash index to find the matched connection cache
  - An access rule check is performed if the connection cache lookup fails
- **The TOE performs an access rule check only if the connection cache lookup fails. The following rules are applied in an access rule check:**
  - Access rules are ordered by Priority. The rule with higher Priority will be applied
  - For incoming packets, srcZone, dstZone, srcIP, dstIP, srcPort, dstPort, IPType are used together as a hash index to find the matching access rule
  - If an incoming packet matches an access rule with the ALLOW action, a new connection cache is added. Otherwise the packet is dropped

#### Verdict:

PASS.

#### 5.1.10.1.8 FPF\_RUL\_EXT.1.5 AGD

The evaluator shall verify that the operational guidance describes how the order of packet filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.

#### Evaluator Findings:

The evaluator reviewed the guidance documentation '**Adding Access Rules**' and verified that it describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that AGD section '**Adding Access Rules**' describes how the order of stateful traffic filtering rules is determined and provides the necessary instructions so that an administrator can configure the order of rule processing. The AGD **point number 7** in the claimed section states that:

**'7. Set your access rule's Priority. You can choose to Auto Prioritize, Insert at the End, or a Manual priority for your access rule.**

**Higher numbers indicate lower priority. The lowest priority rule is the final/default rule applied to matching traffic (traffic matching the defined attributes) when no higher priority rules apply. Lower priority rules should be more general than rules with higher priorities. If a higher priority rule does not match all the attributes, then the next rule is evaluated to see if it applies, all the way down the list of rules. Rules with more specific matching attributes need to be set at a higher priority or else a more general rule could match before that specific rule is evaluated.**

**When you added a new Access Rule, the rule module decided where to place it in the Access Rule table. The rule module uses an Auto Prioritize algorithm that places the most specific rules at the top. The only way to change the priority is to manually edit the rule and then provide the index of where to place it. Finding the rule in a large table to edit it can be difficult.**

**The User Priority for Access Rules provides two choices for the priority types of the new rule:**

- **Auto Prioritize, which uses the Auto Prioritize algorithm that places the most specific rules on the top of the Access Rules table. This is the default choice.**
- **Insert at the end, which indicates to the rule module to place the rule at the end of the Access Rules table, and as a result, makes the new rule easy to locate regardless of the size of the table.**

**Verdict:**

**PASS.**

**5.1.10.1.9 FPF\_RUL\_EXT.1.6 TSS**

The evaluator shall verify that the TSS describes the process for applying packet filtering rules and also that the behavior (either by default, or as configured by the administrator) is to discard packets when there is no rule match.

**Evaluator Findings:**

The evaluator reviewed the TSS, section **FWW\_RUL\_EXT.1** row, and ensured that it describes the process for applying packet filtering rules and also that the behavior (either by default, or as configured by the administrator) is to discard packets when there is no rule match.

The relevant information is found in the following section(s): TOE Summary Specification row **FPF\_RUL\_EXT.1**

Upon investigation, the evaluator found that the claimed TSS section states that:

- **'The TOE performs an access rule check only if the connection cache lookup fails. The following rules are applied in an access rule check:**
  - **Access rules are ordered by Priority. The rule with higher Priority will be applied**
  - **For incoming packets, srcZone, dstZone, srcIp, dstIp, srcPort, dstPort, ipType are used together as a hash index to find the matching access rule**
  - **If an incoming packet matches an access rule with the ALLOW action, a new connection cache is added. Otherwise the packet is dropped**

**In the evaluated configuration, the default action is to DENY a packet if the packet does not match any of the access rules. However, this does not apply for dynamic protocol traffic.'**

The evaluator shall verify the TSS describes when the IPv4 and IPv6 protocols supported by the TOE differ from the full list provided in the RFC Values for IPv4 and IPv6 table.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that there is no instance where the full list provided in the RFC values for IPv4 and IPv6 table differs.

**Verdict:**

**PASS.**

**5.1.10.1.10 FPF\_RUL\_EXT.1.6 AGD**

The evaluator shall verify that the operational guidance describes the behavior if no rules or special conditions apply to the network traffic.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation **'Default Deny Rule'** and verified that it describes the behavior if no rules or special conditions apply to the network traffic.

The relevant information is found in the following section(s): **Default Deny Rule**

Upon investigation, the evaluator found that AGD section **'Default Deny Rule'** states that:

**'In the evaluated configuration, a deny rule applied to any interface, any zone, and for any traffic with the lowest priority must be created. This ensures that any traffic that does not match a configured rule will be denied.'**

If the behavior is configurable, the evaluator shall verify that the operational guidance provides the appropriate instructions to configure the behavior to discard packets with no matching rules.

**Evaluator Findings:**

The evaluator reviewed the guidance documentation and verified that it provides the appropriate instructions to configure the behavior to deny packets with no matching rules.

The relevant information is found in the **Default Deny Rule** section of the AGD.

The evaluator shall verify that the operational guidance describes the range of IPv4 and IPv6 protocols supported by the TOE.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it describes the range of IPv4 and IPv6 protocols supported by the TOE.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that the claimed AGD section claims that:

**'Appliance supports all the IPv4 and IPv6 protocols mentioned in**

**['](https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml)**

**Verdict:**

**PASS.**

---

5.1.11 INTRUSION PREVENTION SYSTEM (IPS)

5.1.11.1 ANAMOLY-BASED IPS FUNCTIONALITY (IPS\_ABD\_EXT)

5.1.11.1.1 IPS\_ABD\_EXT.1 ANAMOLY-BASED IPS FUNCTIONALITY TSS

The evaluator shall verify that the TSS describes the composition, construction, and application of baselines or anomaly-based attributes specified in IPS\_ABD\_EXT.1.1.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it describes the composition, construction, and application of baselines or anomaly-based attributes specified in IPS\_ABD\_EXT.1.1.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_ABD\_EXT.1**

Upon investigation, the evaluator found that the TSS states that:

**The TOE supports baseline and anomaly-based traffic based on time of day. If traffic is received outside of the permitted time of day, the TOE may block or drop the flow of traffic. This rule can be**



**applied to any WAN or LAN network interface. Subsequently, if traffic is received within the permitted time of day, the TOE may allow the traffic to flow.**

The evaluator shall verify that the TSS provides a description of how baselines are defined and implemented by the TOE, or a description of how anomaly-based rules are defined and configured by the administrator.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it provides a description of how baselines are defined and implemented by the TOE, or a description of how anomaly-based rules are defined and configured by the administrator.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**When a packet is received by the TOE, the header and payload data elements are analyzed and compared to the list of signatures to identify any policy violations. Reactions to all signature policy violations can be set to either Detection or Prevention. If Detection is enabled, the TOE identifies the policy violation, logs the instance, and allows the traffic to flow through. If Prevention is enabled, the TOE reacts by identifying the violation, logging the instance, and blocking or dropping the traffic. For TCP sequence number errors, the TOE can remap the sequence number and forward the traffic to its destination.**

**The TOE supports string-based detection signatures by inspecting the payload data elements.**

If 'frequency' is selected in IPS\_ABD\_EXT.1.1, the TSS shall include an explanation of how frequencies can be defined on the TOE.

**Evaluator Findings:**

Frequency is not selected by the TOE; hence, this activity is not applicable.

If 'thresholds' is selected in IPS\_ABD\_EXT.1.1, the TSS shall include an explanation of how the thresholds can be defined on the TOE.

**Evaluator Findings:**

Thresholds is not selected by the TOE; hence, this activity is not applicable.

The evaluator shall verify that each baseline or anomaly-based rule can be associated with a reaction specified in IPS\_ABD\_EXT.1.3.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that each baseline or anomaly-based rule can be associated with a reaction specified in IPS\_ABD\_EXT.1.3.

The relevant information is found in the following section(s): TOE Summary Specification

**IPS\_ABD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**If traffic is received outside of the permitted time of day, the TOE may block or drop the flow of traffic. Subsequently, if traffic is received within the permitted time of day, the TOE may allow the traffic to flow.**

This is consistent with the reactions claimed in IPS\_ABD\_EXT.1.3 in the ST.

The evaluator shall verify that the TSS identifies all interface types capable of applying baseline or anomaly-based rules and explains how they are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it identifies all interface types capable of applying baseline or anomaly-based rules and explains how they are associated with distinct network interfaces.

The relevant information is found in the following section(s): TOE Summary Specification

**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**Depending on the model, the TOE supports a number of WAN and LAN interfaces capable of implementing IPS policies while in inline mode and promiscuous mode. All policies including signature-based, baseline, and anomaly-based are deployed globally across all WAN and LAN interfaces. Each instance of the TOE also supports a management interface (MGMT port) used only for the web-based administration of the TOE.**

**Verdict:**

**PASS.**

**5.1.11.1.2 IPS\_ABD\_EXT.1 ANAMOLY-BASED IPS FUNCTIONALITY AGD**

The evaluator shall verify that the operational guidance provides instructions to manually create baselines or anomaly-based rules according to the selections made in IPS\_ABD\_EXT.1.1. Note that dynamic “profiling” of a network to establish a baseline is outside the scope of the PP-Module.

### Evaluator Findings:

The evaluator checked the AGD and ensured that it provides instructions to manually create baselines or anomaly-based rules according to the selections made in IPS\_ABD\_EXT.1.1.

The relevant information is found in the following section(s): **Schedules** and **Adding Access Rules**

Upon investigation, the evaluator found that the AGD section '**Schedules**' provides instructions to manually create baselines or anomaly-based rules according to the selections made in IPS\_ABD\_EXT.1.1.

**'Schedule Groups are groups of schedules to which you can apply firewall rules. For example, you might want to block access to auction sites during business hours but allow employees to access the sites after hours.'**

#### To create a Schedule Group

1. **Navigate to Object | Match Objects | Schedules.**
2. **Click on ADD. The Add object dialog displays.**
3. **In the Schedule Type section, select how often the schedule occurs: Once, Recurring, or Mixed.**
  - **For a schedule that occurs only once, select the year, month, date, hour, and minutes for the Start and End fields.**
  - **For recurring schedules, select the check boxes for each day the schedule applies. Enter the start time for the recurring schedule in the Start Time field and also the End Time field. Make sure to use the 24-hour format for both of them.**
  - **For the mixed schedule type, you can use the recurring and once options in the same configuration.**
4. **After configuring the desired schedule click on Add.**
5. **To delete an existing schedule, click on Delete this schedule icon.**
6. **To edit an existing schedule, click on the Edit this schedule icon .'**

Further, the evaluator found that the **point number 9** in AGD section '**Adding Access Rules**' states that:

**'9. Specify when the rule is applied by selecting a schedule from the Schedule drop-down menu. If the rule is always applied, select Always. If the schedule you want is not listed in the drop-down menu, click the pencil icon to the right of the menu and create a New Schedule Object. The Adding Schedule Object dialog appears.'**

The evaluator shall verify that the operational guidance provides instructions to associate reactions specified in IPS\_ABD\_EXT.1.3 with baselines or anomaly-based rules.

### Evaluator Findings:

The evaluator checked the AGD and ensured that it provides instructions to associate reactions specified in IPS\_ABD\_EXT.1.3 with baselines or anomaly-based rules.

The relevant information is found in the following section(s): **Schedules**

Upon investigation, the evaluator found that the AGD section ‘Schedules’ provides instructions to associate reactions specified in IPS\_ABD\_EXT.1.3 with baselines or anomaly-based rules.  
**‘In SonicWall, schedules in access rules determine when a specific rule is applied. If traffic is received during the scheduled time, the access rule becomes active and allows the traffic to pass. However, if the traffic is received outside the scheduled time, the access rule is not active, and the traffic will be blocked or handled by default rules.**

The evaluator shall verify that the operational guidance provides instructions to associate the different policies with distinct network interfaces.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions to associate the different policies with distinct network interfaces.

The relevant information is found in the following section(s): **Adding Access Rules**

Upon investigation, the evaluator found that **point numbers 10, 11 and 12** in the claimed AGD section provides instructions to associate the different policies with distinct network interfaces. The AGD states that:

- ‘10. Select the source and destination Zone/Interface from the drop-down menus.**
- 11. Select from the Predefined zones WAN, LAN, DMZ, VPN, MULTICAST, WLAN, and SSLVPN. In addition to predefined zones, custom user-friendly zones can also be configured in Sonicwall, with different security types.**
- 12. Select an interface from the range X0–X33.**

**Note: The number of physical interfaces varies depending on the firewall model.’**

**Verdict:**

**PASS.**

**5.1.11.2 IP BLOCKING (IPS\_IPB\_EXT)**

**5.1.11.2.1 IPS\_IPB\_EXT.1 IP BLOCKING TSS**

The evaluator shall verify how good/bad lists affect the way in which traffic is analyzed with respect to processing packets.

**Evaluator Findings:**

The evaluator reviewed the TSS and verified how good/bad lists affect the way in which traffic is analyzed with respect to processing packets.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_IPB\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**Known-good IP addresses are allowed to pass through the TOE to their destination. Known bad IP addresses are blocked from accessing the network.**

The evaluator shall also verify that the TSS provides details for the attributes that create a known good list, a known bad list, and their associated rules, including how to define the source or destination IP address (e.g. a single IP address or a range of IP addresses).

**Evaluator Findings:**

The evaluator reviewed and verified that the TSS provides details for the attributes that create a known good list, a known bad list, and their associated rules, including how to define the source or destination IP address (e.g. a single IP address or a range of IP addresses).

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_IPB\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**IPS policies are configured by defining a known good list ('included') and a known bad list ('excluded') of IP addresses for each IPS Signature. IP addresses can be defined by a single IP or by a range of IP addresses.**

If the TSF uses address types other than a single IP or a range of IP addresses (e.g. MAC addresses), the evaluator shall check that the TSS explains what configurations would cause non-IP lists of known-good and known-bad addresses to take precedence over IP-based address lists.

**Evaluator Findings:**

The TOE does not use address types other than a single IP or a range of IP addresses; hence, this activity is not applicable.

The evaluator shall also verify that the TSS identifies all the roles and level of access for each of those roles that have been specified in the requirement.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it identifies all the roles and level of access for each of those roles that have been specified in the requirement.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_IPB\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**Only authorized users assigned the Security Administrator role can access and configure the IPS policies.**

**Verdict:**

**PASS.**

#### 5.1.11.2.2 IPS\_IPB\_EXT.1 IP BLOCKING AGD

---

The evaluator shall verify that the administrative guidance provides instructions with how each role specified in the requirement can create, modify and delete the attributes of a known good and known bad lists.

##### **Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions with how each role specified in the requirement can create, modify and delete the attributes of a known good and known bad lists.

The relevant information is found in the following section(s): **Configuring Access Rules for a Zone, Adding Access Rules, Editing Access Rules and Deleting a Custom Access Rule**

Upon investigation, the evaluator summarizes that the claimed AGD section describes how to add, modify, reset to defaults, or delete firewall rules for firewall appliances running SonicOS.

AGD sections **'Adding Access Rules'**, **'Editing Access Rules'** and **'Deleting a Custom Access Rule'** provides clear instructions for how each role (Administrator, User, etc.) can perform the create, modify and delete actions on the attributes of known good and known bad lists.

If the TSF uses address types other than a single IP or a range of IP addresses (e.g. MAC addresses), the evaluator shall check that the operational guidance includes instructions for any configurations that would cause non-IP lists of known-good and known-bad addresses to take precedence over IP-based address lists.

##### **Evaluator Findings:**

The evaluator checked the AGD and ensured that, if the TSF uses address types other than a single IP or a range of IP addresses (e.g. MAC addresses), the AGD includes instructions for any configurations that would cause non-IP lists of known-good and known-bad addresses to take precedence over IP-based address lists.

The relevant information is found in the following section(s): **Configuring Access Rules for a Zone**

Upon investigation, the evaluator found that **point number 13** in the claimed AGD states that:

**Note: The appliance supports both single IP addresses and ranges of IP addresses.**

##### **Verdict:**

**PASS.**

---

#### 5.1.11.3 NETWORK TRAFFIC ANALYSIS (IPS\_NTA\_EXT)

##### 5.1.11.3.1 IPS\_NTA\_EXT.1.1 NETWORK TRAFFIC ANALYSIS TSS

---

The evaluator shall verify that the TSS explains the TOE's capability of analyzing IP traffic in terms of the TOE's policy hierarchy (precedence).

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it explains the TOE’s capability of analyzing IP traffic in terms of the TOE’s policy hierarchy (precedence).

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**The TOE analyzes traffic based on IP address, port, and interface. By default, traffic is first analyzed against the anomaly-based rules and then against the signature-based rules.**

The TSS should identify if the TOE’s policy hierarchy order is configurable by the administrator for IPS policy elements (known-good lists, known-bad lists, signature- based rules, and anomaly-based rules).

**Evaluator Findings:**

The evaluator verified the TSS identified that the TOE’s policy hierarchy order is configurable by the administrator for IPS policy elements (known-good lists, known-bad lists, signature- based rules, and anomaly-based rules).

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**This policy hierarchy order is not configurable.**

Regardless of whether the precedence is configurable, the evaluator shall verify that the TSS describes the default precedence as well as the IP analyzing functions supported by the TOE.

**Evaluator Findings:**

The evaluator reviewed the TSS to ensure that it describes the default precedence as well as the IP analyzing functions supported by the TOE.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row.

Upon investigation, the evaluator found that the TSS states that:  
**The TOE analyzes traffic based on IP address, port, and interface.**

**Verdict:**

**PASS.**

**5.1.11.3.2 IPS\_NTA\_EXT.1.1 NETWORK TRAFFIC ANALYSIS AGD**

The evaluator shall verify that the guidance describes the default precedence.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it describes the default precedence.

The relevant information is found in the following section(s): **Intrusion Protection**

Upon investigation, the evaluator found that the AGD states that:

**'The appliance analyzes traffic based on IP address, port, and interface. By default, traffic is first analyzed against the anomaly-based rules and then against the signature-based rules.'**

If the precedence is configurable, the evaluator shall verify that the guidance explains how to configure the precedence.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that the AGD does not describe any possibility of configuring the precedence. This aligns with the claims made in the TSS.

**Verdict:**

**PASS.**

**5.1.11.3.3 IPS\_NTA\_EXT.1.2 NETWORK TRAFFIC ANALYSIS TSS**

The evaluator shall verify that the TSS indicates that the following protocols are supported:

- IPv4
- IPv6
- ICMPv4
- ICMPv6
- TCP
- UDP

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it indicates that the following protocols are supported:

- IPv4
- IPv6
- ICMPv4
- ICMPv6
- TCP
- UDP

The relevant information is found in the following section(s): TOE Summary Specification **IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:



**The TOE supports the following protocols, which have been compliance tested for assurance:**

- IPv4
- IPv6
- ICMPv4
- ICMPv6
- TCP
- UDP

The evaluator shall verify that the TSS describes how conformance with the identified protocols has been determined by the TOE developer (e.g., third party interoperability testing, protocol compliance testing).

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes how conformance with the identified protocols has been determined by the TOE developer (e.g., third party interoperability testing, protocol compliance testing).

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**The TOE supports the following protocols, which have been compliance tested by the product QA team for assurance:**

- IPv4
- IPv6
- ICMPv4
- ICMPv6
- TCP
- UDP

**Verdict:**

**PASS.**

**5.1.11.3.4 IPS\_NTA\_EXT.1.2 NETWORK TRAFFIC ANALYSIS AGD**

---

There are no guidance EAs for this element.

**5.1.11.3.5 IPS\_NTA\_EXT.1.3 NETWORK TRAFFIC ANALYSIS TSS**

---

The evaluator shall verify that the TSS identifies all interface types capable of being deployed in the modes of promiscuous, and or inline mode as well as the interfaces necessary to facilitate each deployment mode (at a minimum, the interfaces need to support inline mode).

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it identifies all interface types capable of being deployed in the modes of inline mode as well as the interfaces necessary to facilitate each deployment mode (at a minimum, the interfaces need to support inline mode).

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**Depending on the model, the TOE supports a number of WAN and LAN interfaces capable of implementing IPS policies while in inline mode. All policies including signature-based, baseline, and anomaly-based are deployed globally across all WAN and LAN interfaces.**

The evaluator shall also check that the TSS provides a description for how the management interface is logically distinct from any sensor interfaces.

**Evaluator Findings:**

The evaluator also checked and ensured that the TSS provides a description for how the management interface is logically distinct from any sensor interfaces.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_NTA\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:  
**Each instance of the TOE also supports a management interface (MGMT port) used only for the web-based administration of the TOE.  
The MGMT port is distinctly labeled on each device.**

**Verdict:**

**PASS.**

**5.1.11.3.6 IPS\_NTA\_EXT.1.3 NETWORK TRAFFIC ANALYSIS AGD**

The evaluator shall verify that the operational guidance provides instructions on how to deploy each of the deployment methods outlined in the TSS.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions on how to deploy each of the deployment methods outlined in the TSS.

The relevant information is found in the following section(s): **Deployment Modes**

The evaluator shall also verify that the operational guidance provides instructions of applying IPS policies to interfaces for each deployment mode.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions of applying IPS policies to interfaces for each deployment mode.

The relevant information is found in the following section(s): **Adding Access Rules** and **Packet Dissection Objects**

Upon investigation, the evaluator summarizes that the AGD section **Adding Access Rules** states instructions of applying IPS policies to interfaces for Ethernet deployment mode.

AGD section **Packet Dissection Objects** states that **'The Packet Dissection Objects lets you specify specific packet characteristics to filter on.'**

Further, the section **Deployment Modes** in the AGD states the following:

**Note: The hierarchy of IPS policies that are applied does not change with the deployment mode. The only change between modes is how the traffic is bypassed/allowed/dropped in each mode. For Management Mode, IPS policies will not be applied because it is configured as an out-of-bound management interface.**

If the management interface is configurable, the evaluator shall verify that the operational guidance explains how to configure the interface as a management interface.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that the AGD explains how to configure the interface as a management interface.

The relevant information is found in the following section(s): **Management Mode, Configure Management Mode in SonicOS** and **Managing through HTTP/HTTPS**

Upon investigation, the evaluator found that the AGD section **'Management Mode'** and **'Configure Management Mode in SonicOS'** states how to configure the interface as a management interface. The AGD states that:

**'Each appliance includes a distinct and dedicated MGMT port. When using this port, you are in management mode.'**

**Configure Management Mode in SonicOS**

To configure an interface for Management Mode, perform the following steps:

- Login to the SonicWall Management GUI
- Navigate to the Network | Interfaces page.
- Click on Configure on MGMT interface.
- Choose zone as MGMT.
- Set Mode / IP Assignment as Static IP Mode
- Assign IP Address, Subnet Mask and Default Gateway (Optional)

**Note: For the NSv devices, an interface has to be configured in the management zone for it to be distinctly identified as the management interface.'**

Further, the evaluator found that the AGD section '**Managing through HTTP/HTTPS**' states how the appliance can be managed using HTTP or HTTPS and a Web browser. The AGD states that:

**'To manage through HTTP or HTTPS**

1. **Navigate to Device | Settings > Administration.**
2. **Click Management.**
3. **To enable HTTP management globally, select Allow management via HTTP in the WEB MANAGEMENT SETTINGS section. This option is not selected by default.**

- **The default port for HTTP is port 80, but you can configure access through another port. Enter the number of the desired port in the HTTP Port field.**

**If you configure another port for HTTP management, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you configure the port to be 76, then you must type LAN IP Address:76 into the Web browser, for example, http://192.18.16.1:76.**

- **The default port for HTTPS management is 443. To add another layer of security for logging into the SonicWall Security Appliance, change the default port, and enter the preferred port number into the HTTPS Port field.**

**If you configure another port for HTTPS management Port, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you use 700 for the port, then you must log into SonicWall using the port number as well as the IP address; for example, https://192.18.16.1:700.**

The evaluator shall verify that the operational guidance explains how the TOE sends commands to remote traffic filtering devices if this functionality is supported.

**Evaluator Findings:**

The TOE does not support the functionality of sending commands to remote traffic filtering devices, hence this requirement is not applicable.

**Verdict:**

**PASS.**

**5.1.11.4 SIGNATURE-BASED IPS FUNCTIONALITY (IPS\_SBD\_EXT)**

**5.1.11.4.1 IPS\_SBD\_EXT.1.1 SIGNATURE-BASED IPS FUNCTIONALITY TSS**

The evaluator shall verify that the TSS describes what is comprised within a signature rule.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes what is comprised within a signature rule.

The relevant information is found in the following section(s): TOE Summary Specification

**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**Signature rules are comprised of the following settings:**

- **Interface (WAN/LAN)**
- **Source Port**
- **Service**
- **Destination**
- **Included/Excluded Users**
- **Schedule**

The evaluator shall verify that each signature can be associated with a reaction specified in IPS\_SBD\_EXT.1.5.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that each signature can be associated with a reaction specified in IPS\_SBD\_EXT.1.5.

The relevant information is found in the following section(s): TOE Summary Specification

**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**The only difference between the inline and sensor (promiscuous mode) interfaces is the intrusion prevention vs intrusion detection. The way the IPS signature rules are applied is the same for both interface modes. Inline mode interfaces can detect and prevent traffic based on IPS rules while the sensor mode can only detect. For management mode interfaces, IPS signature rules cannot be applied. Management mode interface is used as an out-of-bound interface dedicated to being used in a dedicated management network.**

The evaluator shall verify that the TSS identifies all interface types that are capable of applying signatures and explains how rules are associated with distinct network interfaces. Where interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it identifies all interface types that are capable of applying signatures and explains how rules are associated with distinct network interfaces. Where

interfaces can be grouped into a common interface type (e.g., where the same internal logical path is used, perhaps where a common device driver is used) they can be treated collectively as a distinct network interface.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**Signature rules are comprised of the following settings:**

- **Interface (WAN/LAN)**  
**Administrators can download a pre-determined list of signatures from SonicWall and/or manually create custom signatures to be applied to sensor interfaces.**

**The only difference between the inline and sensor (promiscuous mode) interfaces is the intrusion prevention vs intrusion detection. The way the IPS signature rules are applied is the same for both interface modes. Inline mode interfaces can detect and prevent traffic based on IPS rules while the sensor mode can only detect. For management mode interfaces, IPS signature rules cannot be applied. Management mode interface is used as an out-of-bound interface dedicated to being used in a dedicated management network.**

**Verdict:**

**PASS.**

#### **5.1.11.4.2 IPS\_SBD\_EXT.1.1 SIGNATURE-BASED IPS FUNCTIONALITY AGD [TD0722]**

The evaluator shall verify that the operational guidance provides instructions with how to create and/or configure rules using the following protocols and header inspection fields:

- IPv4: version; header length; packet length; ID; IP flags; fragment offset; time to live (TTL); protocol; header checksum; source address; destination address; IP options; and, if selected, type of service (ToS).
- IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and, if selected, traffic class and/or flow label.
- ICMP: type; code; header checksum; and, if selected, other header fields (varies based on the ICMP type and code).
- ICMPv6: type; code; and header checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: source port; destination port; length; and UDP checksum.

#### **Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions with how to create and/or configure rules using the following protocols and header inspection fields:

- IPv4: version; header length; packet length; ID; IP flags; fragment offset; time to live (TTL); protocol; header checksum; source address; destination address; IP options; and, if selected, type of service (ToS).
- IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and, if selected, traffic class and/or flow label.
- ICMP: type; code; header checksum; and, if selected, other header fields (varies based on the ICMP type and code).
- ICMPv6: type; code; and header checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: source port; destination port; length; and UDP checksum.

The relevant information is found in the following section(s): **Packet Dissection Objects**

Upon investigation, the evaluator found that the AGD states that:

**The following fields can be examined:**

- **IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options.**
- **IPv6: Version; traffic class; flow label; payload length; next header; hop limit; source address; destination address; routing header; home address options.**
- **ICMP: Type; Code; Header Checksum; and Rest of Header (varies based on the ICMP type and code).**
- **ICMPv6: Type; Code; and Header Checksum.**
- **TCP: Source port; destination port; sequence number; acknowledgment number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.**
- **UDP: Source port; destination port; length; and UDP checksum.**

Further, the evaluator examined AGD section ‘**Adding Access Rules**’ and found that **point number 26** states how to apply the Packet Dissection Objects with the above listed fields to access rule. The AGD states that:

**‘26. The Packet Dissection Filter allows you to set up rules based on deep packet inspection (DPI). It can analyze not only basic attributes like IP and port but also elements like specific application protocols or encrypted traffic. By integrating Packet Dissection Objects with access rules, this feature provides the ability to inspect traffic before allowing or denying it based on security policies.’**

The evaluator shall verify that the operational guidance provides instructions with how to select and/or configure reactions specified in IPS\_SBD\_EXT.1.5 in the signature rules.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions with how to select and/or configure reactions specified in IPS\_SBD\_EXT.1.5 in the signature rules.

The relevant information is found in the following section(s): **Adding Access Rules** and **About Negative Matching**

Upon investigation, the evaluator examined AGD section **'Adding Access Rules'** and found that **point number 26** provides instructions with how to select and/or configure reactions specified in IPS\_SBD\_EXT.1.5 in the signature rules. The AGD states that:

**'When a PDF object configured with negative matching is applied to a rule, the behavior depends on the Access rule action. If the Access rule action is "allow," packets that match the PDF object are rejected, while all other traffic is allowed. Conversely, if the Access rule action is "deny," all traffic, including packets that match the PDF object, is denied.'**

Further, the evaluator examined AGD section **'About Negative Matching'** and found that the claimed section states the following:

**'Negative matching provides an alternate way to specify which content to block. You can enable negative matching in a match object when you want to block everything except a particular type of content. When you use the object in a policy, the policy executes actions based on the absence of the content specified in the match object. Multiple list entries in a negative matching object are matched using the logical AND, meaning that the policy action is executed only when all specified negative matching entries are matched.'**

**Although all App Rules policies are DENY policies, you can simulate an ALLOW policy by using negative matching. For instance, you can allow email .txt attachments and block attachments of all other file types. Or you can allow a few types and block all others.**

**Not all match object types can utilize negative matching. For those that can, you see the Enable Negative Matching Checkbox on the Match Object Settings dialog.'**

**Verdict:**

**PASS.**

**5.1.11.4.3 IPS\_SBD\_EXT.1.2 SIGNATURE-BASED IPS FUNCTIONALITY TSS**

The evaluator shall verify that the TSS describes what is comprised within a string-based detection signature.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes what is comprised within a string-based detection signature.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_SBD\_EXT.1** row



Upon investigation, the evaluator found that the TSS states that:

**The TOE supports string-based detection signatures by inspecting the payload data elements. String-based pattern matching with the data elements of the following protocols are also supported:**

- **ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.**
- **ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.**
- **TCP data (characters beyond the 20 byte TCP header), with support for detection of:**
  - **FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.**
  - **HTTP (web) commands and content: commands including GET and POST, and administrator defined strings to match URLs/URIs, and web page content.**
  - **SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.**
  - **UDP data: characters beyond the first 8 bytes of the UDP header**

**To properly detect configured strings within streams, the TOE supports stream reassembly to detect malicious payloads even if split across multiple non-fragmented packets.**

The evaluator shall verify that each packet payload string-based detection signature can be associated with a reaction specified in IPS\_SBD\_EXT.1.5.

#### **Evaluator Findings:**

The evaluator reviewed the TSS and ensured that each packet payload string-based detection signature can be associated with a reaction specified in IPS\_SBD\_EXT.1.5.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**When a packet is received by the TOE, the header and payload data elements are analyzed and compared to the list of signatures to identify any policy violations. Reactions to all signature policy violations can be set to either Detection or Prevention. If Detection is enabled, the TOE identifies the policy violation, logs the instance, and allows the traffic to flow through. If Prevention is enabled, the TOE reacts by identifying the violation, logging the instance, and blocking or dropping the traffic. For TCP sequence number errors, the TOE can remap the sequence number and forward the traffic to its destination.**

**The TOE supports string-based detection signatures by inspecting the payload data elements. String-based pattern matching with the data elements of the following protocols are also supported**

**Verdict:**

**PASS.**

#### 5.1.11.4.4 IPS\_SBD\_EXT.1.2 SIGNATURE-BASED IPS FUNCTIONALITY AGD

The evaluator shall verify that the operational guidance provides instructions with how to configure rules using the packet payload string-based detection fields defined in IPS\_SBD\_EXT.1.2.

##### Evaluator Findings:

The evaluator checked the AGD and ensured that it provides instructions with how to configure rules using the packet payload string-based detection fields defined in IPS\_SBD\_EXT.1.2.

The relevant information is found in the following section(s): **Match Objects** and **App Rules**

Upon investigation, the evaluator found that the AGD states that:

**Match objects represent the set of conditions that must be matched for actions to take place. This includes the object type, the match type (exact, partial, regex, prefix, or suffix), the input representation (text or hexadecimal), and the actual content to match. Match objects were referred to as application objects in previous releases.**

**Hexadecimal input representation is used to match binary content such as executable files, while alphanumeric (text) input representation is used to match things like file or email content. You can also use hexadecimal input representation for binary content found in a graphic image. Text input representation could be used to match the same graphic if it contains a certain string in one of its property's fields. Regular expressions (regex) are used to match a pattern rather than a specific string or value and use alphanumeric input representation.**

**The File Content match object type provides a way to match a pattern or keyword within a file. This type of match object can only be used with FTP Data Transfer, HTTP Server, or SMTP Client policies.**

Further, the evaluator examined AGD section '**App Rules**' and found that **point number 9** states how to apply the Match Objects with the above listed fields to App rule. The AGD states that:

**'9. For Match Object Included, select a match object from the drop-down menu containing the defined match objects applicable to the policy type.'**

The evaluator shall verify that the operational guidance provides instructions with how to configure reactions specified in IPS\_SBD\_EXT.1.5 for each string-based detection signature.

##### Evaluator Findings:

The evaluator checked the AGD and ensured that it provides instructions with how to configure reactions specified in IPS\_SBD\_EXT.1.5 for each string-based detection signature.

The relevant information is found in the following section(s): **App Rules**

Upon investigation, the evaluator examined AGD section '**App Rules**' and found that **point number 11** provides instructions with how to configure reactions specified in IPS\_SBD\_EXT.1.5 for each string-based detection signature. The AGD states that:

'11. For Action Object, select an action from the drop-down menu containing actions applicable to the policy type, and can include predefined actions plus any customized actions. The default for all policy types, except CFS is Reset/Drop; the default for CFS is No Action.

NOTE: For a log-only policy, select No Action.'

The evaluator shall verify that the operational guidance provides instructions with how rules are associated with distinct network interfaces that are capable of being associated with signatures.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions with how rules are associated with distinct network interfaces that are capable of being associated with signatures.

The relevant information is found in the following section(s): **Intrusion Protection**

Upon investigation, the evaluator found that the claimed AGD section states that:  
**'The appliance analyzes traffic based on IP address, port, and interface. By default, traffic is first analyzed against the anomaly-based rules and then against the signature-based rules.'**

**Verdict:**

**PASS.**

**5.1.11.4.5 IPS\_SBD\_EXT.1.3 SIGNATURE-BASED IPS FUNCTIONALITY TSS**

The evaluator shall verify that the TSS describes how the attacks defined in IPS\_SBD\_EXT.1.3 are processed by the TOE and what reaction is triggered when these attacks are identified.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes how the attacks defined in IPS\_SBD\_EXT.1.3 are processed by the TOE and what reaction is triggered when these attacks are identified.

The relevant information is found in the following section(s): TOE Summary Specification  
**IPS\_SBD\_EXT.1** row

Upon investigation, the evaluator found that the TSS states that:

**By analyzing the header-based signature traffic, the TOE is able to detect and prevent the following types of attacks:**

- **IP Attacks**
  - **IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)**
  - **IP source address equal to the IP destination (Land attack)**
- **ICMP Attacks**
  - **Fragmented ICMP Traffic (e.g. Nuke attack)**
  - **Large ICMP Traffic (Ping of Death attack)**
- **TCP Attacks**

- TCP NULL flags
- TCP SYN+FIN flags
- TCP FIN only flags
- TCP SYN+RST flags
- UDP Attacks
  - UDP Bomb Attack
  - UDP Chargen DoS Attack

When a packet is received by the TOE, the header and payload data elements are analyzed and compared to the list of signatures to identify any policy violations. Reactions to all signature policy violations can be set to either Detection or Prevention. If Detection is enabled, the TOE identifies the policy violation, logs the instance, and allows the traffic to flow through. If Prevention is enabled, the TOE reacts by identifying the violation, logging the instance, and blocking or dropping the traffic. For TCP sequence number errors, the TOE can remap the sequence number and forward the traffic to its destination.

Verdict:

PASS.

#### 5.1.11.4.6 IPS\_SBD\_EXT.1.3 SIGNATURE-BASED IPS FUNCTIONALITY AGD

The evaluator shall verify that the operational guidance provides instructions with configuring rules to identify the attacks defined in IPS\_SBD\_EXT.1.3 as well as the reactions to these attacks as specified in IPS\_SBD\_EXT.1.5.

##### Evaluator Findings:

The evaluator checked the AGD and ensured that it provides instructions with configuring rules to identify the attacks defined in IPS\_SBD\_EXT.1.3 as well as the reactions to these attacks as specified in IPS\_SBD\_EXT.1.5.

The relevant information is found in the following section(s): **Header-based Signature**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'The following attacks are detected, blocked, and logged by the appliance by default, without any additional configuration:**

- IP Attacks

IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)

IP source address equal to the IP destination (Land attack)

- ICMP Attacks

Fragmented ICMP Traffic (e.g. Nuke attack)

Large ICMP Traffic (Ping of Death attack)

- **TCP Attacks**

**TCP NULL flags**

**TCP SYN+FIN flags**

**TCP FIN only flags**

**TCP SYN+RST flags**

- **UDP Attacks**

**UDP Bomb Attack**

**UDP Chargen DoS Attack: This attack is not detected by default.**

**To detect and block it, follow the steps below:**

- 1. Navigate to OBJECT | Match Objects | Service and click on +Add**
- 2. Create a service object for UDP port 19 to represent a Chargen packet.**
- 3. Navigate to POLICY | Rules and Policies | Access Rules and click on +Add**
- 4. Create an access rule, select the created service object in the access rule, and set the action to 'Drop'.**

**When UDP traffic on port 19 is received, the appliance will detect and drop it according to the configured rule, and a log will be generated.'**

**Verdict:**

**PASS.**

#### **5.1.11.4.7 IPS\_SBD\_EXT.1.4 SIGNATURE-BASED IPS FUNCTIONALITY TSS**

The evaluator shall verify that the TSS describes how the attacks defined in IPS\_SBD\_EXT.1.4 are processed by the TOE and what reaction is triggered when these attacks are identified.

**Evaluator Findings:**

The evaluator reviewed the TSS and ensured that it describes how the attacks defined in IPS\_SBD\_EXT.1.4 are processed by the TOE and what reaction is triggered when these attacks are identified.

The relevant information is found in the following section(s): TOE Summary Specification **IPS\_SBD\_EXT.1** row.

Upon investigation, the evaluator found that the TSS states that:

By analyzing the header-based signature traffic, the TOE is able to detect and prevent the following types of attacks:

- **Flooding a host (DoS attack)**
  - ICMP Flooding (Smurf attack, and ping flood)
  - TCP flooding (e.g. SYN flood)
- **Flooding a network (DoD attack)**
- **TCP Attacks**
  - IP protocol scanning
  - TCP port scanning
  - UDP port scanning
  - ICMP scanning

When a packet is received by the TOE, the header and payload data elements are analyzed and compared to the list of signatures to identify any policy violations. Reactions to all signature policy violations can be set to either Detection or Prevention. If Detection is enabled, the TOE identifies the policy violation, logs the instance, and allows the traffic to flow through. If Prevention is enabled, the TOE reacts by identifying the violation, logging the instance, and blocking or dropping the traffic. For TCP sequence number errors, the TOE can remap the sequence number and forward the traffic to its destination.

**Verdict:**

**PASS.**

#### 5.1.11.4.8 IPS\_SBD\_EXT.1.4 SIGNATURE-BASED IPS FUNCTIONALITY AGD

The evaluator shall verify that the operational guidance provides instructions with configuring rules to identify the attacks defined in IPS\_SBD\_EXT.1.4 as well as the reactions to these attacks as specified in IPS\_SBD\_EXT.1.5.

##### **Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides instructions with configuring rules to identify the attacks defined in IPS\_SBD\_EXT.1.4 as well as the reactions to these attacks as specified in IPS\_SBD\_EXT.1.5.

The relevant information is found in the following section(s): **SYN Flood Protection, ICMP Flood Protection and Port Scan Detection**

Upon investigation, the evaluator summarizes that the AGD section **SYN Flood Protection** provides configuration for TCP flooding (e.g. SYN flood). The AGD states that:

**'A SYN Flood Protection mode is the level of protection that you can select to protect your network against half-opened TCP sessions and high frequency SYN packet transmissions.**

**To enable SYN flood Protection**

1. **Navigate to Network | Firewall.**
2. **Go to TCP > Layer 3 SYN Flood Protection- SYN Proxy tab.**
3. **In the SYN Flood Protection Mode drop-down menu, select a protection mode.**

**Watch and Report Possible SYN Floods** – The device monitors SYN traffic on all interfaces and logs suspected SYN flood activity that exceeds a packet-count threshold. This option does not actually turn on the SYN Proxy on the device, so the device forwards the TCP three-way handshake without modification.

This is the least invasive level of SYN Flood protection. Select this option if your network is not in a high-risk environment.'

Further, the summarizes that the AGD section **ICMP Flood Protection, flooding a network (DoS attack)** provides configuration for ICMP flooding (Smurf attack, and ping flood). The AGD states that: **'ICMP Flood attacks are a type of denial-of-service (DoS) attack.**

**To enable ICMP flood protection**

1. **Navigate to Network | Firewall | Flood Protection.**
2. **Under ICMP Flood Protection, enable the check box for Enable ICMP Flood Protection.**

**If the rate of ICMP packets per second exceeds the allowed threshold for a specified duration of time, the appliance drops subsequent ICMP packets to protect against a flood attack.'**

The evaluator examined AGD section **Port Scan Detection** and found that it summarizes configuration for Protocol and port scanning for IP protocol scanning, TCP port scanning, UDP port scanning and ICMP scanning. The AGD states that:

**'The Port Scan Detection feature detects if someone is scanning your ports and notifies you.**

**To enable Port Scan Detection**

1. **Open the Internal Diag page/Internal Settings.**
2. **Click on Internal settings to access the Internal Settings page or Diag page.**
3. **In NDPP Port Scan Settings, select Enable NDPP Port Scan Detection.**

**The Port Scan Detection feature detects if someone is scanning your ports and notifies you'**

**Verdict:**

**PASS.**

#### 5.1.11.4.9 IPS\_SBD\_EXT.1.5 SIGNATURE-BASED IPS FUNCTIONALITY TSS

---

There are no TSS EAs for this element.

#### 5.1.11.4.10 IPS\_SBD\_EXT.1.5 SIGNATURE-BASED IPS FUNCTIONALITY AGD

---

The guidance EAs for this element are performed in conjunction with IPS\_SBD\_EXT.1.1, IPS\_SBD\_EXT.1.3, and IPS\_SBD\_EXT.1.4.

#### 5.1.11.4.11 IPS\_SBD\_EXT.1.6 SIGNATURE-BASED IPS FUNCTIONALITY TSS

---

There are no TSS EAs for this element.

#### 5.1.11.4.12 IPS\_SBD\_EXT.1.6 SIGNATURE-BASED IPS FUNCTIONALITY AGD

---

The evaluator shall verify that the operational guidance provides configuration instructions, if needed, to detect payload across multiple packets.

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it provides configuration instructions, if needed, to detect payload across multiple packets.

The relevant information is found in the following sections: **Match Objects** and **App Rules**

Upon investigation, the evaluator found that the AGD section '**Match Objects**' and '**App Rules**' provides configuration instructions to detect payload across multiple packets.

The AGD section '**Match Object**' states that:

**'Match objects represent the set of conditions that must be matched for actions to take place. This includes the object type, the match type (exact, partial, regex, prefix, or suffix), the input representation (text or hexadecimal), and the actual content to match.'**

**Hexadecimal input representation is used to match binary content such as executable files, while alphanumeric (text) input representation is used to match things like file or email content.**

**The File Content match object type provides a way to match a pattern or keyword within a file. This type of match object can only be used with FTP Data Transfer, HTTP Server, or SMTP Client policies. To configure a match object**

- 1. Navigate to Object | Match Objects | Match Objects.**
- 2. Click on ADD. The Add object dialog displays.**
- 3. In the Object Name field, type a descriptive name for the object.**
- 4. Select a Match Object Type from the drop-down menu. Your selection affects the available options on this screen.**
- 5. Select a Match Type from the drop-down menu. The available selections depend on the match object type.**
- 6. For the Input Representation field, click Alphanumeric to match a text pattern, or click Hexadecimal if you want to match binary content.**
- 7. In the Content text box, type the pattern to match.**
- 8. Click Add icon. The content appears in the List field. Repeat to add another element to match.'**

Upon further investigation, the evaluator found that the AGD section '**App Rules**' states that:

**'App Rules provide a solution for setting policy rules for application signatures. As a set of application-specific policies, App Rules provide you with granular control over network traffic on the level of users, email addresses, schedules, and IP-subnets. The primary functionality of this application-layer access control feature is to regulate Web browsing, file transfer, email, and email attachments.**

**To configure App Rules**

- 1. Navigate to Policy | Rules and Policies | App Rules.**
- 2. Click on Add. Add App Rule dialog displays.**
- 3. Enter a descriptive name in the Policy Name field.**



4. Select a Policy Type from the drop-down menu. Your selection here affects the options available in the dialog.
5. Select the Address Source and Address Destination from the drop-down menu.
6. Select the Service Source and Service Destination from the drop-down menu.
7. For Exclusion Address, optionally select an Address Group or Address Object from the drop-down menu. This address is not affected by the policy.
8. For Exclusion Service, optionally select a Service Group or Service Object from the drop-down menu. This address is not affected by the policy.
9. For Match Object Included, select a match object from the drop-down menu containing the defined match objects applicable to the policy type.
10. For Match Object Excluded, select the match object from the drop-down. The excluded match object provides the ability to differentiate subdomains in the policy.  
The Excluded Match Object does not take effect when the match object type is set to Custom Object. Custom Objects cannot be selected as the Exclusion Match Object.
11. For Action Object, select an action from the drop-down menu containing actions applicable to the policy type, and can include predefined actions plus any customized actions. The default for all policy types, except CFS, is Reset/Drop; the default for CFS is No Action.
12. For Users/Groups, select from the drop-down menus for both Included and Excluded. The selected users or group under Excluded are not affected by the policy.
13. For Schedule, select from the drop-down menu, which contains a variety of schedules for the policy to be in effect. Specifying a schedule other than the default, Always On, turns on the rule only during the scheduled time.
14. If you want the policy to create a flow reporting when a match is found, select the Enable Flow Reporting checkbox.
15. If you want the policy to create a log entry when a match is found, select the Enable Logging checkbox.
16. To record more details in the log, select the Log individual object content checkbox.
17. If the policy type is IPS Content, select the Log using IPS message format checkbox to display the category in the log entry as Intrusion Prevention rather than Application Control, and to use a prefix such as IPS Detection Alert in the log message rather than Application Control Alert. This is useful if you want to use log filters to search for IPS alerts.
18. If the policy type is App Control Content, select the Log using App Control message format checkbox to display the category in the log entry as Application Control, and to use a prefix such as Application Control Detection Alert in the log message. This is useful if you want to use log filters to search for Application Control alerts.
19. If the policy type is CFS, select the Log using CFS message format checkbox to display the category in the log entry as Network Access, and to use a log message such as website access denied in the log message rather than no prefix. This is useful if you want to use log filters to search for content filtering alerts.
20. For Log Redundancy Filter, you can select Global Settings to use the global value, or you can enter a number of seconds to delay between each log entry for this policy. The local setting overrides the global setting only for this policy; other policies are not affected.
21. For Direction, click either Basic or Advanced and select a direction from the drop-down menu. Basic allows you to select incoming, outgoing, or both. Advanced allows you to select between zones, such as LAN to WAN. IPS Content, App Control Content, or CFS policy types do not provide this configuration option.

22. If the policy type is IPS Content, App Control Content, or CFS, select a zone from the Zone drop-down menu. The policy is applied to this zone.
23. Click OK.
24. Once the APP rule is configured, it is automatically enabled.'

**Verdict:**

**PASS.**

## 5.2 SELECTION-BASED REQUIREMENTS

### 5.2.1 CRYPTOGRAPHIC SUPPORT (FCS)

#### 5.2.1.1 FCS\_HTTPS\_EXT.1 HTTPS PROTOCOL

##### 5.2.1.1.1 FCS\_HTTPS\_EXT.1 TSS

The evaluator shall examine the TSS and determine that enough detail is provided to explain how the implementation complies with RFC 2818.

#### Evaluator Findings:

The evaluator examined the TSS and determine that enough detail is provided to explain how the implementation complies with RFC 2818.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_HTTPS\_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TLS Server protocol is implemented in support of the HTTPS connection to the administrative interface. The TLS implantation is described by FCS\_TLSS\_EXT.1.**

**The TOE is always the receiver of HTTPS connections. The TOE's HTTPS protocol complies with RFC 2818 by implementing all the "MUST", "REQUIRED", and "SHOULD" statements. The TOE does not implement any "MUST NOT" or "SHOULD NOT" statements. The TOE initiates an exchange of closure alerts before closing a connection. The TOE does not implement an "incomplete close".**

#### Verdict:

**PASS.**

##### 5.2.1.1.2 FCS\_HTTPS\_EXT.1 AGD

The evaluator shall examine the guidance documentation to verify it instructs the Administrator how to configure TOE for use as an HTTPS client or HTTPS server.

#### Evaluator Findings:

The evaluator examined the guidance documentation to verify it instructs the Administrator how to configure TOE for use as an HTTPS client or HTTPS server.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS and Selecting a Security Certificate**

Upon investigation, the evaluator found that the AGD section 'Managing through HTTP/HTTPS' states that:

'The SonicWall appliance can be managed using HTTP or HTTPS and a Web browser.'

#### To manage through HTTP or HTTPS

4. Navigate to Device | Settings > Administration.
5. Click Management.
6. To enable HTTP management globally, select Allow management via HTTP in the WEB MANAGEMENT SETTINGS section. This option is not selected by default.
  - The default port for HTTP is port 80, but you can configure access through another port. Enter the number of the desired port in the HTTP Port field.

If you configure another port for HTTP management, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you configure the port to be 76, then you must type LAN IP Address:76 into the Web browser, for example, <http://192.18.16.1:76>.
  - The default port for HTTPS management is 443. To add another layer of security for logging into the SonicWall Security Appliance, change the default port, and enter the preferred port number into the HTTPS Port field.

If you configure another port for HTTPS management Port, you must include the port number when you use the IP address to log into the SonicWall Security Appliance. For example, if you use 700 for the port, then you must log into SonicWall using the port number as well as the IP address; for example, <https://192.18.16.1:700>.

Furthermore, the evaluator investigated the AGD section 'Selecting a Security Certificate' which states that:

#### To specify the type of security certificate

1. Navigate to Device | Settings > Administration.
2. Click Management.
3. From Certificate Selection drop-down list, select the type of certificate for your website:
  - Using Self-signed Certificate allows you to continue using a certificate without downloading a new one each time you log into the SonicWall Security Appliance. This option is selected by default.
  - Use Import Certificate to select an imported certificate from the Device | Settings > Certificates page to use for authentication to the management interface. A confirmation message is displayed.
4. Click OK. The Device | Settings > Certificates page displays.

5. In the Certificate Common Name field, enter the IP address or common name for the firewall. If you choose Use Selfsigned Certificate, SonicOS populates the field with the firewall's IP address.
6. Click Accept.

**Verdict:**

**PASS.**

### 5.2.1.2 FCS\_IPSEC\_EXT.1 IPSEC PROTOCOL

#### 5.2.1.2.1 FCS\_IPSEC\_EXT.1.1 TSS

The evaluator shall examine the TSS and determine that it describes what takes place when a packet is processed by the TOE, e.g., the algorithm used to process the packet. The TSS describes how the SPD is implemented and the rules for processing both inbound and outbound packets in terms of the IPsec policy. The TSS describes the rules that are available and the resulting actions available after matching a rule. The TSS describes how those rules and actions form the SPD in terms of the BYPASS (e.g., no encryption), DISCARD (e.g., drop the packet), and PROTECT (e.g., encrypt the packet) actions defined in RFC 4301.

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

#### **Evaluator Findings:**

The evaluator examined the TSS and determine that it describes what takes place when a packet is processed by the TOE, e.g., the algorithm used to process the packet. The TSS describes how the SPD is implemented and the rules for processing both inbound and outbound packets in terms of the IPsec policy. The TSS describes the rules that are available and the resulting actions available after matching a rule. The TSS describes how those rules and actions form the SPD in terms of the BYPASS (e.g., no encryption), DISCARD (e.g., drop the packet), and PROTECT (e.g., encrypt the packet) actions defined in RFC 4301.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TOE implements IPsec in accordance with RFC 4301.**

**The TOE Administrator implements an IPsec policy to encrypt data between the TOE and the audit server.**

**In general, an IPsec policy can be established to encrypt data (PROTECT). If traffic not belonging to the protected interface or subnet is found on this interface, the traffic will bypass encryption and be routed to the destination in plaintext (BYPASS). If plaintext traffic is received on a protected interface or subnet, the traffic is discarded and deleted (DISCARD).**

**IPsec VPN traffic is secured in two stages:**

- **Authentication: The first phase establishes the authenticity of the sender and receiver of the traffic using an exchange of the public key portion of a public-private key pair. This phase must be successful before the VPN tunnel can be established.**
- **Encryption: The traffic in the VPN tunnel is encrypted using AES.**

**The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel. The TOE supports IKE version 2.**

**Verdict:**

**PASS.**

#### 5.2.1.2.2 FCS\_IPSEC\_EXT.1 TSS (VPNGW)

All existing activities regarding "Pre-shared keys" apply to all selections including pre-shared keys. If any selection with "Pre-shared keys" is included, the evaluator shall check to ensure that the TSS describes how the selection works in conjunction with the authentication of IPsec connections.

##### **Evaluator Findings:**

The evaluator reviewed the ST and the TSS and verified that pre-shared keys are not selected, hence, this is not applicable.

**Verdict:**

**PASS.**

#### 5.2.1.2.3 FCS\_IPSEC\_EXT.1.3 TSS

The evaluator checks the TSS to ensure it states that the VPN can be established to operate in transport mode and/or tunnel mode (as identified in FCS\_IPSEC\_EXT.1.3).

##### **Evaluator Findings:**

The evaluator checks the TSS to ensure it states that the VPN can be established to operate in transport mode and/or tunnel mode (as identified in FCS\_IPSEC\_EXT.1.3).

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TOE can be only operated in Tunnel mode in the evaluated configuration. This is a default setting and cannot be changed when using IKEv2.**

**Verdict:**

**PASS.**

**5.2.1.2.4 FCS\_IPSEC\_EXT.1.4 TSS**

The evaluator shall examine the TSS to verify that the selected algorithms are implemented. In addition, the evaluator ensures that the SHA-based HMAC algorithm conforms to the algorithms specified in FCS\_COP.1/KeyedHash Cryptographic Operations (for keyed-hash message authentication) and if the SHA-based HMAC function truncated output is utilized it must also be described.

**Evaluator Findings:**

The evaluator examined the TSS to verify that the selected algorithms are implemented. In addition, the evaluator ensures that the SHA-based HMAC algorithm conforms to the algorithms specified in FCS\_COP.1/KeyedHash Cryptographic Operations (for keyed-hash message authentication) and if the SHA-based HMAC function truncated output is utilized it must also be described.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **AES-CBC-128, AES-CBC-192, and AES-CBC-256 are supported for ESP.**

**The HMAC implementation conforms to HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512.**

**The IKE payload is encrypted using AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, or AES-GCM-256.**

**Verdict:**

**PASS.**

**5.2.1.2.5 FCS\_IPSEC\_EXT.1.5 TSS**

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

**Evaluator Findings:**

The evaluator examined the TSS to verify that IKEv1 and/or IKEv2 are implemented.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The exchange of information to authenticate the members of the VPN and encrypt/decrypt the data uses the Internet Key Exchange (IKE) protocol for exchanging authentication information (keys) and establishing the VPN tunnel.**

**The TOE supports IKE version 2.**

**IKEv2 is the default proposal type for new VPN policies.**

For IKEv1 implementations, the evaluator shall examine the TSS to ensure that, in the description of the IPsec protocol, it states that aggressive mode is not used for IKEv1 Phase 1 exchanges, and that only main mode is used. It may be that this is a configurable option.

**Evaluator Findings:**

The TOE does not support IKE version 1; hence this assurance activity is not applicable.

**Verdict:**

**PASS.**

**5.2.1.2.6 FCS\_IPSEC\_EXT.1.6 TSS**

The evaluator shall ensure the TSS identifies the algorithms used for encrypting the IKEv1 and/or IKEv2 payload, and that the algorithms chosen in the selection of the requirement are included in the TSS discussion.

**Evaluator Findings:**

The evaluator ensured the TSS identifies the algorithms used for encrypting the IKEv1 and/or IKEv2 payload, and that the algorithms chosen in the selection of the requirement are included in the TSS discussion.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The IKE payload is encrypted using AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, or AES-GCM-256.**

**Verdict:**

**PASS.**

**5.2.1.2.7 FCS\_IPSEC\_EXT.1.7 TSS**

The evaluator shall ensure the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 1 SA lifetime and/or the IKEv2 SA lifetime.



**Evaluator Findings:**

The evaluator ensured the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 1 SA lifetime and/or the IKEv2 SA lifetime.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The IKEv2 SA lifetime is selected in the SPD and can be set to be between 120 and 86400 seconds (24 hours).**

The evaluator shall verify that the selection made here corresponds to the selection in FCS\_IPSEC\_EXT.1.5.

**Evaluator Findings:**

The evaluator verified that the selection made here corresponds to the selection in FCS\_IPSEC\_EXT.1.5.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the selection made here corresponds to the selection in FCS\_IPSEC\_EXT.1.5.

**Verdict:**

**PASS.**

**5.2.1.2.8 FCS\_IPSEC\_EXT.1.8 TSS**

The evaluator shall ensure the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 2 SA lifetime and/or the IKEv2 Child SA lifetime.

**Evaluator Findings:**

The evaluator ensured the TSS identifies the lifetime configuration method used for limiting the IKEv1 Phase 2 SA lifetime and/or the IKEv2 Child SA lifetime.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The IKEv2 Child SA lifetime is selected in the SPD and can also be set to be between 120 and 28800 seconds (8 hours).**

The evaluator shall verify that the selection made here corresponds to the selection in FCS\_IPSEC\_EXT.1.5.

<b>Evaluator Findings:</b>
The evaluator verified that the selection made here corresponds to the selection in FCS_IPSEC_EXT.1.5.
The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_IPSEC_EXT.1.</b>
Upon investigation, the evaluator found that the selection made here corresponds to the selection in FCS_IPSEC_EXT.1.5.

**Verdict:**

**PASS.**

**5.2.1.2.9 FCS\_IPSEC\_EXT.1.9 TSS**

The evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating "x".

<b>Evaluator Findings:</b>
The evaluator checked and ensured that for each DH group supported, the TSS describes the process for generating "x".
The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_IPSEC_EXT.1.</b>
Upon investigation, the evaluator found that the TSS states that: <b>The DRBG described in FCS_RBG_EXT.1 is used to generate each nonce for DH groups 14, 19, 20, and 21 for IKEv2</b>

The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of "x" meets the stipulations in the requirement.

<b>Evaluator Findings:</b>
The evaluator verified that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of "x" meets the stipulations in the requirement.
The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_IPSEC_EXT.1.</b>

Upon investigation, the evaluator found that the TSS states that:  
**The DRBG described in FCS\_RBG\_EXT.1 is used to generate each nonce for DH groups 14, 19, and 20 for IKEv2, having possible lengths of 224, 256, 384, and 512 bit, corresponding to each of the supported DH group.**

**Verdict:**

**PASS.**

**5.2.1.2.10 FCS\_IPSEC\_EXT.1.10 TSS**

If the first selection is chosen, the evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating each nonce.

**Evaluator Findings:**

The first selection is not chosen, hence, this is not applicable.

The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.

**Evaluator Findings:**

The TOE uses PRF hash to generate nonces. This is described in the two requirements below.

If the second selection is chosen, the evaluator shall check to ensure that, for each PRF hash supported, the TSS describes the process for generating each nonce.

**Evaluator Findings:**

If the second selection is chosen, the evaluator checked and ensured that for each PRF hash supported, the TSS describes the process for generating each nonce.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that:  
**The DRBG described in FCS\_RBG\_EXT.1 is used to generate each nonce for DH groups 14, 19, 20, and 21 for IKEv2, having possible lengths of 224, 256, 384, and 512 bit, corresponding to each of the supported DH group. The TOE supports SHA-256, SHA-384, and SHA-512 as the hash in the PRF.**

The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.

#### Evaluator Findings:

The evaluator verified that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The size of the nonce is 128-256 bits (half of the pseudorandom function with a minimum of 128 bits).**

#### Verdict:

**PASS.**

#### 5.2.1.2.11 FCS\_IPSEC\_EXT.1.11 TSS

The evaluator shall check to ensure that the DH groups specified in the requirement are listed as being supported in the TSS. If there is more than one DH group supported, the evaluator checks to ensure the TSS describes how a particular DH group is specified/negotiated with a peer.

#### Evaluator Findings:

The evaluator checked to ensure that the DH groups specified in the requirement are listed as being supported in the TSS. If there is more than one DH group supported, the evaluator checks to ensure the TSS describes how a particular DH group is specified/negotiated with a peer.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that:  
**The TOE supports Group 14, 256-bit Random ECP Group (Group 19), 384-bit Random ECP Group (Group 20), and 521-bit Random ECP Group (Group 21).**

**The DH Group selection can be made in the VPN Policy page.**

#### Verdict:

**PASS.**

#### 5.2.1.2.12 FCS\_IPSEC\_EXT.1.12 TSS

The evaluator shall check that the TSS describes the potential strengths (in terms of the number of bits in the symmetric key) of the algorithms that are allowed for the IKE and ESP exchanges. The TSS shall also describe the checks that are done when negotiating IKEv1 Phase 2 and/or IKEv2 CHILD\_SA suites and ensured that the strength (in terms of the number of bits of key in the symmetric algorithm) of the negotiated algorithm is less than or equal to that of the IKE SA this is protecting the negotiation.

**Evaluator Findings:**

The evaluator checked that the TSS describes the potential strengths (in terms of the number of bits in the symmetric key) of the algorithms that are allowed for the IKE and ESP exchanges. The TSS also describes the checks that are done when negotiating IKEv1 Phase 2 and/or IKEv2 CHILD\_SA suites and ensured that the strength (in terms of the number of bits of key in the symmetric algorithm) of the negotiated algorithm is less than or equal to that of the IKE SA this is protecting the negotiation.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The symmetric algorithms supported for IKEv2 IKE\_SA uses the same or greater key length as the symmetric algorithms used to protect IKEv2 CHILD\_SA.**

**The available options ensure that the IKEv2 IKE\_SA symmetric algorithm key length is equal to or greater than the IKEv2 CHILD\_SA symmetric algorithm key length.**

**Verdict:**

**PASS.**

**5.2.1.2.13 FCS\_IPSEC\_EXT.1.13 TSS**

The evaluator ensures that the TSS identifies RSA and/or ECDSA as being used to perform peer authentication. The description must be consistent with the algorithms as specified in FCS\_COP.1/SigGen Cryptographic Operations (for cryptographic signature).

**Evaluator Findings:**

The evaluator ensures that the TSS identifies RSA and/or ECDSA as being used to perform peer authentication. The description must be consistent with the algorithms as specified in FCS\_COP.1/SigGen Cryptographic Operations (for cryptographic signature).

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Peer authentication is performed using third-party RSA or ECDSA certificates that conform to RFC 4945.**

If pre-shared keys are chosen in the selection, the evaluator shall check to ensure that the TSS describes how pre-shared keys are established and used in authentication of IPsec connections. The description in the TSS shall also indicate how pre-shared key establishment is accomplished for TOEs that can generate a pre-shared key as well as TOEs that simply use a pre-shared key.

**Evaluator Findings:**

The ST does not claim pre-shared keys, hence, not applicable.

**Verdict:**

PASS.

5.2.1.2.14 FCS\_IPSEC\_EXT.1.14 TSS

The evaluator shall ensure that the TSS describes how the TOE compares the peer's presented identifier to the reference identifier. This description shall include which field(s) of the certificate are used as the presented identifier (DN, Common Name, or SAN). If the TOE simultaneously supports the same identifier type in the CN and SAN, the TSS shall describe how the TOE prioritizes the comparisons (e.g. the result of comparison if CN matches but SAN does not). If the location (e.g. CN or SAN) of non-DN identifier types must explicitly be configured as part of the reference identifier, the TSS shall state this. If the ST author assigned an additional identifier type, the TSS description shall also include a description of that type and the method by which that type is compared to the peer's presented certificate, including what field(s) are compared and which fields take precedence in the comparison.

**Evaluator Findings:**

The evaluator ensured that the TSS describes how the TOE compares the peer's presented identifier to the reference identifier. This description includes which field(s) of the certificate are used as the presented identifier (DN, Common Name, or SAN). If the TOE simultaneously supports the same identifier type in the CN and SAN, the TSS describes how the TOE prioritizes the comparisons (e.g. the result of comparison if CN matches but SAN does not). If the location (e.g. CN or SAN) of non-DN identifier types must explicitly be configured as part of the reference identifier, the TSS states this. If the ST author assigned an additional identifier type, the TSS description also includes a description of that type and the method by which that type is compared to the peer's presented certificate, including what field(s) are compared and which fields take precedence in the comparison.

The relevant information is found in the following section(s): TOE Summary Specification **FCS\_IPSEC\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Reference identifiers are supported for SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN, and Distinguished Name (DN). SAN takes precedence over CN.**

**The format of any Subject Distinguished Name is determined by the issuing Certification Authority. Common fields are Country (C=), Organization (O=), Organizational Unit (OU=), Common Name (CN=), Locality (L=), and vary with the issuing Certification Authority. The actual Subject Distinguished Name field in an X.509 Certificate is a binary object which is converted to a string and compared with the expected string.**

**Verdict:**

PASS.

5.2.1.2.15 FCS\_IPSEC\_EXT.1.1 AGD

The evaluator shall examine the guidance documentation to verify it instructs the Administrator how to construct entries into the SPD that specify a rule for processing a packet. The description includes all

three cases – a rule that ensures packets are encrypted/decrypted, dropped, and flow through the TOE without being encrypted.

**Evaluator Findings:**

The evaluator examined the guidance documentation to verify it instructs the Administrator how to construct entries into the SPD that specify a rule for processing a packet. The description includes all three cases – a rule that ensures packets are encrypted/decrypted, dropped, and flow through the TOE without being encrypted.

The relevant information is found in the following section(s): **Configuring Access Rules for a Zone and Adding Access Rules**

Upon investigation, the evaluator found that the AGD section **‘Configuring Access Rules for a Zone’** states that:

**‘To Add Access Rules**

- 1. Navigate to Policy | Rules and Policies | Access Rules.**
- 2. Click on Add. The Add Rule dialog displays.**

**To configure rules, the service or service group that the rule applies to must first be defined. If it is not, you can define the service or service group and then create one or more rules for it.’**

Furthermore, the evaluator examined the AGD section **‘Adding Access Rules’** and found that the following bullet in the claimed section includes description for creating rules such that the packets are encrypted/decrypted (Allow rule) and dropped (Deny rule):

**‘5. Select an Action, that is, how the rule processes (permits or blocks) the specified IP traffic:**

- Allow (default): As long as the Enable option is selected, your access rule is active and permits the traffic.**
- Deny: The firewall denies all connections matching this rule and blocks the page specified and the action profile is served for web traffic. The firewall also resets the connections on both sides.**
- Discard: Firewall silently drops any packets matching this rule.’**

Lastly the evaluator examined the AGD section **‘Configuring Routing Rules’** and found that the flow of traffic through TOE without being encrypted (bypass) can be achieved by disabling **‘Allow VPN path to take precedence’** in Routing Rules option under POLICY | Rules and Policies tab.

The AGD states that:

**17. Select Allow VPN path to take precedence to allow a matching VPN network to take precedence over the static route when the VPN tunnel is up. This option is not selected by default.**

The evaluator shall determine that the description in the guidance documentation is consistent with the description in the TSS, and that the level of detail in the guidance documentation is sufficient to allow the administrator to set up the SPD in an unambiguous fashion. This includes a discussion of how ordering of rules impacts the processing of an IP packet.

**Evaluator Findings:**

The evaluator determined that the description in the guidance documentation is consistent with the description in the TSS, and that the level of detail in the guidance documentation is sufficient to allow the administrator to set up the SPD in an unambiguous fashion. This includes a discussion of how ordering of rules impacts the processing of an IP packet.

The relevant information is found in the following section(s): **Configuring Access Rules for a Zone and Adding Access Rules**

Upon investigation, the evaluator summarizes that the claimed AGD sections are sufficient to allow the administrator to set up the SPD in an unambiguous fashion.

Furthermore, the AGD section **'Adding Access Rules'** also includes information of how ordering of rules impacts the processing of an IP packet. A NOTE under a bullet state that:

**'The access rules are sorted from the most specific at the top, to less specific at the bottom of the table. At the bottom of the table is the Any rule. The default access rule is all IP services except those listed in the Access Rules page.**

**NOTE: Higher numbers indicate lower priority. The lowest priority rule is the final/default rule applied to matching traffic (traffic matching the defined attributes) when no higher priority rules apply. Lower priority rules should be more general than rules with higher priorities. If a higher priority rule does not match all the attributes, then the next rule is evaluated to see if it applies, all the way down the list of rules. Rules with more specific matching attributes need to be set at a higher priority or else a more general rule could match before that specific rule is evaluated.'**

**Verdict:**

PASS.

5.2.1.2.16 FCS\_IPSEC\_EXT.1 AGD (VPNGW)

If any selection with "Pre-shared Keys" is selected, the evaluator shall check that the operational guidance describes any configuration necessary to enable any selected authentication mechanisms.

**Evaluator Findings:**

The ST does not claim pre-shared keys, hence, not applicable.

**Verdict:**

PASS.

5.2.1.2.17 FCS\_IPSEC\_EXT.1.3 AGD

The evaluator shall confirm that the guidance documentation contains instructions on how to configure the connection in each mode selected.



#### Evaluator Findings:

The evaluator confirmed that the guidance documentation contains instructions on how to configure the connection in each mode selected.

The relevant information is found in the following section(s): **IPsec VPN** and **Configure VPN**

Upon investigation, the evaluator found that the AGD section '**IPsec VPN**' states that:  
**'The appliance only operates in Tunnel mode in the evaluated configuration. This is a default setting and cannot be changed when using IKEv2.'**

Furthermore, the evaluator also examined the AGD section '**Configure VPN**' which states that:

**'To configure a VPN**

1. **Navigate to the NETWORK | IPsec VPN > Rules and Settings page.**
2. **Make the appropriate version selection, either IPv4 or IPv6.**
3. **Click +Add.**
4. **Complete the General, Network, Proposals, and Advanced tabs on the VPN Policy dialog. The following sections provide additional information for each of those tabs.'**

#### Verdict:

**PASS.**

#### 5.2.1.2.18 FCS\_IPSEC\_EXT.1.4 AGD

The evaluator checks the guidance documentation to ensure it provides instructions on how to configure the TOE to use the algorithms selected.

#### Evaluator Findings:

The evaluator checked the guidance documentation to ensure it provides instructions on how to configure the TOE to use the algorithms selected.

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator found that **point number 17** in the claimed AGD sections states that: '

17. **Set the options in the IPsec (Phase 2) Proposal section. The default values for Protocol, Encryption, Authentication, Enable Perfect Forward Secrecy, and Life Time (seconds) are acceptable for most VPN SA configurations.:**

**Be sure the Phase 2 values on the opposite side of the tunnel are configured to match.**

If you selected ESP in the Protocol field, in the Encryption field you can select from six encryption algorithms that are included in Suite B cryptography:

Suite B Cryptography Options	Other Options
AESGCM16-128	DES
AESGCM16-192	3DES
AESGCM16-256	AES-128
AESGMAC-128	AES-192
AESGMAC-192	AES-256
AESGMAC-256	None

If NDCPP compliance is enabled, then in the Encryption field you can select from AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, or AES-GCM-256.

If you selected AH in the Protocol field, the Encryption field is dimmed, and you cannot select any options.

For the Authentication field if IKEv2 mode was selected, choose SHA-256, SHA-384, and SHA-512 from the drop-down menu.

For all Exchange modes, enter a value for Life Time (seconds). The default setting of 28800 forces the tunnel to renegotiate and exchange keys every 8 hours.'

Verdict:

PASS.

#### 5.2.1.2.19 FCS\_IPSEC\_EXT.1.5 AGD

The evaluator shall check the guidance documentation to ensure it instructs the administrator how to configure the TOE to use IKEv1 and/or IKEv2 (as selected), and how to configure the TOE to perform NAT traversal (if selected).

##### Evaluator Findings:

The evaluator checked the AGD to ensure it instructs the administrator how to configure the TOE to use IKEv1 and/or IKEv2 (as selected), and how to configure the TOE to perform NAT traversal (if selected).

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key, Configuring IKE Using Third Party Certificates and NAT Traversal**

Upon investigation, the evaluator found that the AGD section '**Configuring IKE Using a Preshared Secret Key and Configuring IKE Using Third Party Certificates**' instructs the administrator how to configure the TOE to use IKEv2.

15. Under IKE (Phase 1) Proposal, choose one of the following options from the Exchange drop-down menu:

<b>Main Mode</b>	Uses IKEv1 Phase 1 proposals with IPsec Phase 2 proposals. Suite B cryptography options are available for the DH Group in IKE Phase 1 settings, and for Encryption in the IPsec Phase 2 settings.
<b>Aggressive Mode</b>	Generally used when WAN addressing is dynamically assigned. Uses IKEv1 Phase 1 proposals with IPsec Phase 2 proposals. Suite B cryptography options are available for the DH Group in IKE Phase 1 settings, and for Encryption in the IPsec Phase 2 settings.
<b>IKEv2 Mode</b>	Causes all negotiation to happen through IKEv2 protocols, rather than using IKEv1 phase 1.

Furthermore, the evaluator found that the AGD section ‘NAT Traversal’ instructs the administrator how to configure the TOE to perform NAT traversal.

**To find NAT Traversal setting**

1. Login to SonicWall appliance.
2. Click Network in the top navigation menu.
3. Click IPsec VPN | Advanced.
4. Toggle the ‘Enable NAT Traversal’ switch.

If the IKEv1 Phase 1 mode requires configuration of the TOE prior to its operation, the evaluator shall check the guidance documentation to ensure that instructions for this configuration are contained within that guidance.

<b>Evaluator Findings:</b>
The ST does not select IKEv1, hence this part of assurance activity is not applicable.

**Verdict:**  
**PASS.**

**5.2.1.2.20 FCS\_IPSEC\_EXT.1.6 AGD**

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

<b>Evaluator Findings:</b>
The evaluator ensured that the guidance documentation describes the configuration of all selected algorithms in the requirement.
The relevant information is found in the following section(s): <b>Configuring IKE Using a Preshared Secret Key</b> and <b>Configuring IKE Using Third Party Certificates</b>

Upon investigation, the evaluator found that the **point number 16** in the AGD section ‘**Configuring IKE Using a Preshared Secret Key**’ and **point number 15** AGD section ‘**Configuring IKE Using Third Party Certificates**’ states that: ‘

**16. Under IKE (Phase 1) Proposal, set the values for the remaining options. The default values for DH Group, Encryption, Authentication, and Life Time are acceptable for most VPN configurations.**

If IKEv2 Mode is selected for the Exchange field, the DH Group, Encryption, and Authentication fields are dimmed, and no selection can be made for those options.

Be sure the Phase 1 values on the opposite side of the tunnel are configured to match.

For the DH Group, when in Main Mode or Aggressive Mode, you can select from several Diffie-Hellman exchanges:

**Diffie-Hellman Groups Included in Suite B Cryptography**

**Other Diffie-Hellman Options**

---

**256-bit Random ECP Group**

**Group 1**

---

**384-bit Random ECP Group**

**Group 2**

---

**521-bit Random ECP Group**

**Group 5**

---

**192-bit Random ECP Group**

**Group 14**

---

**224-bit Random ECP Group**

---

For DH Group, when in IKEv2 mode, you can select from supports Group 14, 256-bit Random ECP Group (Group 19), 384-bit Random ECP Group (Group 20), and 521-bit Random ECP Group (Group 21).

For the Encryption field, if IKEv2 mode was selected, choose AES-CBC-128, AES-CBC-192, and AES-CBC-256 from the drop-down menu.

For the Encryption field, if Main Mode or Aggressive Mode was selected, choose 3DES, DES, AES-128 (default), AES-192, or AES-256 from the drop-down menu.

For the Authentication field, if Main Mode or Aggressive Mode was selected, choose SHA-1 (default), MD5, SHA256, SHA384, or SHA512 for enhanced authentication security.

For the Authentication field if IKEv2 mode was selected, choose SHA-256, SHA-384, and SHA-512 from the drop-down menu.’

**Verdict:**

**PASS.**

#### **5.2.1.2.21 FCS\_IPSEC\_EXT.1.7 AGD [TD0800]**

The evaluator shall verify that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before

the maximum SA lifetime of 24 hours is exceeded (e.g. configure a time value of 23h 45min to ensure the actual rekey is performed no later than 24h).

**Evaluator Findings:**

The evaluator verified that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before the maximum SA lifetime of 24 hours is exceeded (e.g. configure a time value of 23h 45min to ensure the actual rekey is performed no later than 24h).

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator found that the **point number 16** in the AGD sections '**Configuring IKE Using a Preshared Secret Key**' and '**Configuring IKE Using Third Party Certificates**' states that: '**For all Exchange modes, enter a value for Life Time (seconds). The default setting of 28800 forces the tunnel to renegotiate and exchange keys every 8 hours.**'

The evaluator shall verify that the guidance documentation allows the Administrator to configure the Phase 1 SA value of 24 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 24 hours. It is not permitted to configure a value of 24 hours if that leads to an actual rekey after more than 24hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.

**Evaluator Findings:**

The evaluator verified that the guidance documentation allows the Administrator to configure the Phase 1 SA value of 24 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 24 hours. It is not permitted to configure a value of 24 hours if that leads to an actual rekey after more than 24hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator found that the **point number 16** in the AGD sections '**Configuring IKE Using a Preshared Secret Key**' and '**Configuring IKE Using Third Party Certificates**' states that: '**For all Exchange modes, enter a value for Life Time (seconds). The default setting of 28800 forces the tunnel to renegotiate and exchange keys every 8 hours.**'

**Verdict:**

PASS.

### 5.2.1.2.22 FCS\_IPSEC\_EXT.1.8 AGD [TD0800]

The evaluator shall verify that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before the maximum SA lifetime of 8 hours is exceeded (e.g. configure a time value of 7h 45min to ensure the actual rekey is performed no later than 8h).

#### Evaluator Findings:

The evaluator verified that the values for SA lifetimes can be configured and that the instructions for doing so are located in the guidance documentation. If time-based limits are supported, configuring the limit may lead to a rekey no later than the specified limit. For some implementations, it may be necessary, though, to configure the TOE with a lower time value to ensure a rekey is performed before the maximum SA lifetime of 8 hours is exceeded (e.g. configure a time value of 7h 45min to ensure the actual rekey is performed no later than 8h).

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator found that the **point number 17** in the AGD sections '**Configuring IKE Using a Preshared Secret Key**' and '**Configuring IKE Using Third Party Certificates**' states that: '**For all Exchange modes, enter a value for Life Time (seconds). The default setting of 28800 forces the tunnel to renegotiate and exchange keys every 8 hours.**'

The evaluator shall verify that the guidance documentation allows the Administrator to configure the Phase 2 SA value of 8 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 8 hours. It is not permitted to configure a value of 8 hours if that leads to an actual rekey after more than 8 hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.

#### Evaluator Findings:

The evaluator verified that the guidance documentation allows the Administrator to configure the Phase 2 SA value of 8 hours or provides sufficient instruction about the time value to configure to ensure the rekey is performed no later than the maximum SA lifetime of 8 hours. It is not permitted to configure a value of 8 hours if that leads to an actual rekey after more than 8 hours. Currently there are no values mandated for the number of bytes, the evaluator just ensures that this can be configured if selected in the requirement.

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator found that the **point number 16** in the AGD sections '**Configuring IKE Using a Preshared Secret Key**' and '**Configuring IKE Using Third Party Certificates**' states that: '**For all Exchange modes, enter a value for Life Time (seconds). The default setting of 28800 forces the tunnel to renegotiate and exchange keys every 8 hours.**'

**Verdict:**

**PASS.**

**5.2.1.2.23 FCS\_IPSEC\_EXT.1.11 AGD**

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

**Evaluator Findings:**

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

The relevant information is found in the following section(s): **Configuring IKE Using a Preshared Secret Key** and **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator examined the claimed AGD sections and summarizes that **point numbers 16 and 17** of AGD section '**Configuring IKE Using a Preshared Secret Key**' and **point numbers 15 and 17** of AGD section '**Configuring IKE Using Third Party Certificates**' states that:

**'Under IKE (Phase 1) Proposal, set the values for the remaining options. The default values for DH Group, Encryption, Authentication, and Life Time are acceptable for most VPN configurations. For the Encryption field, if IKEv2 mode was selected, choose AES-CBC-128, AES-CBC-192, and AES-CBC-256 from the drop-down menu.**

**For the Authentication field if IKEv2 mode was selected, choose SHA-256, SHA-384, and SHA-512 from the drop-down menu.**

**Set the options in the IPsec (Phase 2) Proposal section. The default values for Protocol, Encryption, Authentication, Enable Perfect Forward Secrecy, and Life Time (seconds) are acceptable for most VPN SA configurations.**

**If NDCPP compliance is enabled, then in the Encryption field you can select from AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, or AES-GCM-256.**

**If you selected AH in the Protocol field, the Encryption field is dimmed, and you cannot select any options.**

**For the Authentication field if IKEv2 mode was selected, choose SHA-256, SHA-384, and SHA-512 from the drop-down menu.'**

**Verdict:**

**PASS.**

**5.2.1.2.24 FCS\_IPSEC\_EXT.1.13 AGD**

The evaluator ensures the guidance documentation describes how to set up the TOE to use certificates with RSA and/or ECDSA signatures and public keys.

#### **Evaluator Findings:**

The evaluator ensures the guidance documentation describes how to set up the TOE to use certificates with RSA and/or ECDSA signatures and public keys.

The relevant information is found in the following section(s): **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator examined the claimed AGD section and found that the section describes how to set up the TOE to use certificates with RSA and/or ECDSA signatures and public keys. The AGD section states that:

**'You must have a valid certificate from a third-party certificate authority installed on your SonicWall firewall before you can configure your VPN policy using a third-party IKE certificate. Reference identifiers are supported for SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN, and Distinguished Name (DN). SAN takes precedence over CN. The format of any Subject Distinguished Name is determined by the issuing Certification Authority. Common fields are Country (C=), Organization (O=), Organizational Unit (OU=), Common Name (CN=), Locality (L=), and vary with the issuing Certification Authority.**

To create a VPN SA using IKE and third-party certificates:

1. Navigate to NETWORK | IPsec VPN > Rules and Settings.
2. Click +Add to create a new policy or click the Edit icon if you are updating an existing policy.
3. In the Authentication Method field, select IKE using 3rd Party Certificates. The VPN Policy window displays the third-party certificate options in the IKE Authentication section.
4. Type a name for the Security Association in the Name field.
5. Type the IP address or Fully Qualified Domain Name (FQDN) of the primary remote SonicWall in the IPsec Primary Gateway Name or Address field.
6. If you have a secondary remote SonicWall, enter the IP address or Fully Qualified Domain Name (FQDN) in the IPsec Secondary Gateway Name or Address field.
7. Under IKE Authentication, select a third-party certificate from the Local Certificate list. You must have imported local certificates before selecting this option.
8. For Local IKE ID Type, the default is Default ID from Certificate. Or, choose one of the following:
  - Distinguished Name (DN)
  - Email ID (UserFQDN)
  - Domain Name (FQDN)
  - IP Address (IPV4)



These alternate selections are the same as those for Peer IKE ID Type, described in the next step.

**NOTE:** In NDPP mode, only certificates with a valid CA are available for selection.

8. From the Peer IKE ID Type drop-down menu, select one of the following Peer ID types:

**Peer IKE ID Type**

Option	Definition
Default ID from Certificate	Authentication is taken from the default ID on the certificate.
Distinguished Name (DN)	Authentication is based on the certificate's Subject Distinguished Name field, which is contained in all certificates by default. The entire Distinguished Name field must be entered for site-to-site VPNs. Wild card characters are not supported. The format of any Subject Distinguished Name is determined by the issuing Certificate Authority. Common fields are Country (C=), Organization (O=), Organizational Unit (OU=), Common Name (CN=), Locality (L=), and vary with the issuing Certificate Authority. The actual Subject Distinguished Name field in an X.509 Certificate is a binary object which must be converted to a string for matching purposes. The fields are separated by the forward slash character, for example: /C=US/O=SonicWall, Inc./OU=TechPubs/CN=Joe Pub.
Email ID (UserFQDN)	Authentication based on the Email ID (UserFQDN) types are based on the certificate's Subject Alternative Name field, which is not contained in all certificates by default. If the certificate contains a Subject Alternative Name, that value must be used. For site-to-site VPNs, wild card characters cannot be used. The full value of the Email ID must be entered. This is because site-to-site VPNs are expected to connect to a single peer, whereas Group VPNs expect to connect to multiple peers.
Domain Name (FQDN)	Authentication based on the Domain Name (FQDN) types are based on the certificate's Subject Alternative Name field, which is not contained in all certificates by default. If the certificate contains a Subject Alternative Name, that value must be used. For site-to-site VPNs, wild card characters cannot be used. The full value of the Domain Name must be entered because site to site VPNs are expected to connect to a single peer, whereas Group VPNs expect to connect to multiple peers.
IP Address (IPV4)	Based on the IPv4 IP address.

To find the certificate details (Subject Alternative Name, Distinguished Name, and so on), navigate to the **DEVICE | Settings > Certificates** page.

9. Type an ID string in the Peer IKE ID field.'

The evaluator shall check that the guidance documentation describes how pre- shared keys are to be generated and established. The description in the guidance documentation shall also indicate how pre- shared key establishment is accomplished for TOEs that can generate a pre-shared key as well as TOEs that simply use a pre-shared key.

**Evaluator Findings:**

The ST does not claim pre-shared keys, hence, not applicable.

The evaluator will ensure that the guidance documentation describes how to configure the TOE to connect to a trusted CA and ensure a valid certificate for that CA is loaded into the TOE and marked “trusted”.

**Evaluator Findings:**

The evaluator will ensure that the guidance documentation describes how to configure the TOE to connect to a trusted CA and ensure a valid certificate for that CA is loaded into the TOE and marked “trusted”.

The relevant information is found in the following section(s): **Certificates Table, Certificate Details and Importing a Certificate Authority Certificate**

Upon investigation, the evaluator found that the AGD section ‘**Importing a Certificate Authority Certificate**’ describes how to configure the TOE to connect to a trusted CA:

**To import a certificate from a certificate authority**

1. **Navigate to Device | Settings > Certificates.**
2. **Click Import. The IMPORT CERTIFICATE dialog is displayed.**
3. **Choose Import a CA certificate from a PKCS#7 (\*.p7b) or DER (.der or .cer) encoded file. The Import Certificate dialog settings change.**
4. **Click Add File and locate the certificate file.**
5. **Click Open.**
6. **Click Import to import the certificate into the firewall. When it is imported, you can view the certificate entry in the Certificates table.**
7. **Click the certificate displayed on the Certificates page to see the status and other details.**

Furthermore, the evaluator found that the AGD section **‘Certificates Table’** describes how to ensure a valid certificate for that CA is loaded into the TOE and marked “trusted”.

**‘The Certificates page provides all the settings for managing CA and Local Certificates. The table page displays this information about certificates:**

Column	Information Displayed
<b>CERTIFICATE</b>	<b>Name of the certificate.</b>
<b>TYPE</b>	<b>Type of certificate:</b> <ul style="list-style-type: none"> <li>• CA certificate</li> <li>• Local certificate</li> <li>• Pending request</li> </ul>
<b>VALIDATED</b>	<b>Validation information:</b> <ul style="list-style-type: none"> <li>• Blank</li> <li>• Yes- When a local certificate with a valid CA is loaded onto the TOE, the "VALIDATED" column will display 'Yes'.</li> <li>• Invalid</li> <li>• Expire in n days</li> <li>• Expired</li> </ul>

Moreover, the **‘Certificate Details’** section of the AGD states that:

**Click the certificate's row in the table to display information about the certificate. This might include the following, depending on the type of certificate:**

- **Status-** Shows 'Verified' when a local certificate with a valid CA is successfully loaded onto the TOE.

**Verdict:**

**PASS.**

**5.2.1.2.25 FCS\_IPSEC\_EXT.1.14 AGD**

The evaluator shall ensure that the operational guidance describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by the TOE does not guarantee unique identifiers, the evaluator shall ensure that the operational guidance provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.

**Evaluator Findings:**

The evaluator ensured that the operational guidance describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by the TOE does not guarantee unique identifiers, the evaluator ensured that the

operational guidance provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.

The relevant information is found in the following section(s): **Configuring IKE Using Third Party Certificates**

Upon investigation, the evaluator examined the claimed AGD section and found that it describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). The **point number 8** from AGD section '**Configuring IKE Using Third Party Certificates**' states that:

**'8. For Local IKE ID Type, the default is Default ID from Certificate. Or, choose one of the following:**

- **Distinguished Name (DN)**
- **Email ID (UserFQDN)**
- **Domain Name (FQDN)**
- **IP Address (IPV4)**

**These alternate selections are the same as those for Peer IKE ID Type, described in the next step.**

**SAN takes precedence over CN.**

**NOTE: In NDPP mode, only certificates with a valid CA are available for selection. Appliance does guarantee unique identifiers.**

**NOTE: In NDPP mode, only certificates with a valid CA are available for selection'**

**Verdict:**

**PASS.**

### 5.2.1.3 FCS\_TLSS\_EXT.1 EXTENDED: TLS SERVER PROTOCOL WITHOUT MUTUAL AUTHENTICATION

#### 5.2.1.3.1 FCS\_TLSS\_EXT.1.1 TSS

The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified.

#### **Evaluator Findings:**

The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the ciphersuites supported are specified.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TOE operates as a TLS server for the web GUI trusted path.**

**The server only allows TLS protocol version 1.2 and rejects all other protocol version, including SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 and any other unknown TLS version string supplied.**

**The TLS server is restricted to the following ciphersuites:**

- TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA
- TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA384
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256
- TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384
- TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256
- TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384
- TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256
- TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384

**The ciphersuites are not configurable.**

The evaluator shall check the TSS to ensure that the ciphersuites specified are identical to those listed for this component.

#### **Evaluator Findings:**

The evaluator checked the TSS and ensured that the ciphersuites specified are identical to those listed for this component.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1**

Upon investigation, the evaluator found that the ciphersuites specified in the TSS of the ST document are identical to those listed for this component.

#### **Verdict:**

**PASS.**

#### 5.2.1.3.2 FCS\_TLSS\_EXT.1.2 TSS

The evaluator shall verify that the TSS contains a description of how the TOE technically prevents the use of old SSL and TLS versions.

**Evaluator Findings:**

The evaluator verified that the TSS contains a description of how the TOE technically prevents the use of old SSL and TLS versions.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1**

Upon investigation, the evaluator found that the TSS states that: **The server only allows TLS protocol version 1.2 and rejects all other protocol version, including SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 and any other unknown TLS version string supplied.**

**Verdict:**

**PASS.**

**5.2.1.3.3 FCS\_TLSS\_EXT.1.3 TSS [TD0635]**

If using ECDHE and/or DHE ciphers, the evaluator shall verify that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server. For example, if the TOE supports TLS\_DHE\_RSA\_WITH\_AES\_128\_CBC\_SHA cipher and Diffie-Hellman parameters with size 2048 bits, then list Diffie-Hellman Group 14.

**Evaluator Findings:**

The evaluator verified that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1**

Upon investigation, the evaluator found that the TSS states that:

**For RSA key agreement schemes, key agreement parameters are restricted to 2048-bits, 3072-bits, and 4096-bits.**

**For ECDSA key agreement schemes, the key agreement parameters are restricted to secp256r1, secp384r1, and secp521r1 curves.**

DHE ciphersuites are not supported by TOE.

**Verdict:**

**PASS.**

**5.2.1.3.4 FCS\_TLSS\_EXT.1.4 TSS [TD0569]**

The evaluator shall verify that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246) and/or if session resumption based on session tickets is supported (RFC 5077).

<b>Evaluator Findings:</b>
<p>The evaluator verified that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246) and/or if session resumption based on session tickets is supported (RFC 5077).</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_TLSS_EXT.1</b></p> <p>Upon investigation, the evaluator found that the TSS states that: <b>The TLS server supports session resumption based on session tickets and session IDs. Session tickets adhere to the structural format provided in section 4 of RFC 5077. Session IDs adhere to the structural format provided in RFC 5246.</b></p>

If session tickets are supported, the evaluator shall verify that the TSS describes that the session tickets are encrypted using symmetric algorithms consistent with FCS\_COP.1/DataEncryption.

<b>Evaluator Findings:</b>
<p>The evaluator verified that the TSS describes that the session tickets are encrypted using symmetric algorithms consistent with FCS_COP.1/DataEncryption.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_TLSS_EXT.1</b></p> <p>Upon investigation, the evaluator found that the TSS states that: <b>Session tickets and session IDs are encrypted according to the TLS negotiated symmetric encryption algorithm derived from the TLS handshake.</b></p>

The evaluator shall verify that the TSS identifies the key lengths and algorithms used to protect session tickets.

<b>Evaluator Findings:</b>
<p>The evaluator verified that the TSS identifies the key lengths and algorithms used to protect session tickets.</p> <p>The relevant information is found in the following section(s): TOE Summary Specification <b>FCS_TLSS_EXT.1</b></p>

Upon investigation, the evaluator found that the TSS states that:  
**Session tickets and session IDs are encrypted according to the TLS negotiated symmetric encryption algorithm derived from the TLS handshake.**

If session tickets are supported, the evaluator shall verify that the TSS describes that session tickets adhere to the structural format provided in Section 4 of RFC 5077 and if not, a justification shall be given of the actual session ticket format.

**Evaluator Findings:**

The evaluator verified that the TSS describes that session tickets adhere to the structural format provided in Section 4 of RFC 5077 and if not, a justification was given of the actual session ticket format.

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1**

Upon investigation, the evaluator found that the TSS states that:  
**The TLS server supports session tickets. Session tickets adhere to the structural format provided in section 4 of RFC 5077.**

If the TOE claims a (D)TLS server capable of session resumption (as a single context, or across multiple contexts), the evaluator shall verify that the TSS describes how session resumption operates (i.e. what would trigger a full handshake, e.g. checking session status, checking Session ID, etc.). If multiple contexts are used the TSS describes how session resumption is coordinated across those contexts. In case session establishment and session resumption are always using a separate context, the TSS shall describe how the contexts interact with respect to session resumption (in particular regarding the session ID). It is acceptable for sessions established in one context to be resumable in another context.

**Evaluator Findings:**

The relevant information is found in the following section(s): TOE Summary Specification  
**FCS\_TLSS\_EXT.1**

Upon investigation, the evaluator found that the TSS states that:  
**The TLS server supports session resumption based on session tickets and session IDs.**

**Session tickets and session IDs are encrypted according to the TLS negotiated symmetric encryption algorithm derived from the TLS handshake.**

**Session resumption and establishment require session tickets and session IDs. The TOE-generated session IDs are used for session resumption and establishment in the Server Hello message in the TLS handshake. When session tickets are used, the TOE generates session tickets after the initial handshake.**



**Verdict:**

PASS.

5.2.1.3.5 FCS\_TLSS\_EXT.1.1 AGD

The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).

**Evaluator Findings:**

The evaluator checked the AGD and ensured that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS**

Upon investigation, the evaluator found that the claimed AGD section states that:

**'The appliance operates as a TLS server for the web GUI trusted path.**

- **The TLS server is restricted to the following cipher suites:**
  - TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_CBC\_SHA256
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_CBC\_SHA384
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_128\_GCM\_SHA256
  - TLS\_ECDHE\_ECDSA\_WITH\_AES\_256\_GCM\_SHA384
  - TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256
  - TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384
  - TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256
  - TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384
- **To configure supported cipher suites no other configuration is required other than enabling NDCPP mode refer Enabling NDCPP Compliance.**

**Verdict:**

PASS.

#### 5.2.1.3.6 FCS\_TLSS\_EXT.1.2 AGD

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

##### Evaluator Findings:

The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD.

The relevant information is found in the following section(s): **Enforcing TLS Version and Managing through HTTP/HTTPS**

Upon investigation, the evaluator found that the AGD states that in **NDPP mode, the server only allows TLS protocol version 1.2 and rejects all other protocol versions, including SSL 2.0, SSL 3.0, TLS 1.0, TLS 1.1, and any other unknown TLS version strings supplied.**

##### Verdict:

**PASS.**

#### 5.2.1.3.7 FCS\_TLSS\_EXT.1.3 AGD

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

##### Evaluator Findings:

The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS and Selecting a Security Certificate**

Upon investigation, the evaluator found that the claimed section **Managing through HTTP/HTTPS** states that:

**“The appliance does not support DHE cipher suite. For RSA key agreement schemes, the key agreement parameters are restricted to 2048-bits, 3072-bits, and 4096-bits keys. For ECDSA key agreement schemes, the key agreement parameters are restricted by default to secp256r1, secp384r1, and secp521r1 curves; no other configuration is required once NDPP mode is enabled.”**

**The section “Selecting a Security Certificate” provides configuration steps to assign a certificate as the TLS Server certificate.**

##### Verdict:

**PASS.**

#### 5.2.1.3.8 FCS\_TLSS\_EXT.1.4 AGD [TD0569]

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

##### Evaluator Findings:

The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD.

The relevant information is found in the following section(s): **Managing through HTTP/HTTPS**

Upon investigation, the evaluator found that the claimed AGD section states that:

**By default, the TLS server supports session resumption based on session tickets and session IDs and no additional configuration is required once NDPP mode is enabled.**

##### Verdict:

**PASS.**

## 5.2.2 IDENTIFICATION AND AUTHENTICATION (FIA)

### 5.2.2.1 FIA\_X509\_EXT.1/REV X.509 CERTIFICATE VALIDATION

#### 5.2.2.1.1 FIA\_X509\_EXT.1/REV TSS

The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA\_X509\_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). It is expected that revocation checking is performed when a certificate is used in an authentication step and when performing trusted updates (if selected). It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected).

##### Evaluator Findings:

The evaluator ensured the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA\_X509\_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied).

The relevant information is found in the following section(s): TOE Summary Specification  
**FIA\_X509\_EXT.1/Rev.**

Upon investigation, the evaluator found that the TSS states that: **The validity of certificates is checked on certificate import and prior to usage of the public key within the certificate.**

**Certificate validation includes checks of:**

- the certificate validity dates
- the validation path, ensuring that the certificate path terminates with a trusted CA certificate
- basicConstraints, ensuring the presence of the basicConstraints extension
- revocation status, using OCSP
- extendedKeyUsage properties, when the certificate is used for OCSP

**The certificate path validation algorithm is implemented as described in RFC 5280.**

The TSS shall describe when revocation checking is performed and on what certificates. If the revocation checking during authentication is handled differently depending on whether a full certificate chain or only a leaf certificate is being presented, any differences must be summarized in the TSS section and explained in the Guidance.

#### **Evaluator Findings:**

The TSS describes when revocation checking is performed and on what certificates.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_X509\_EXT.1/Rev.**

Upon investigation, the evaluator found that the TSS states that:

**When a certificate is used for secure channels, an OCSP server is contacted to verify that the certificate is still valid. If the validity of a certificate that is used for IPSec tunnel cannot be verified, the system rejects the certificate and drops the connection for IPSec tunnels.**

#### **Verdict:**

**PASS.**

#### **5.2.2.1.2 FIA\_X509\_EXT.1/REV AGD**

The evaluator shall also ensure that the guidance documentation describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA\_X509\_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate.

#### **Evaluator Findings:**

The evaluator also ensured that the AGD describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA\_X509\_EXT.1.1) that are not

supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate.

The relevant information is found in the following section(s): **Certificate Validation and Revocation Checking using OCSP**

Upon investigation, the evaluator examined the AGD sections '**Certificate Validation**' and '**Revocation Checking using OCSP**' and found that the AGD describes where the check of validity of the certificates takes place, any of the rules for extendedKeyUsage fields, how certificate revocation checking is performed and on which certificate.

The AGD states that:

**The validity of certificates is checked on certificate import and prior to usage of the public key within the certificate.**

**Certificate validation includes checks of:**

- **The certificate validity dates**
- **The validation path, ensuring that the certificate path terminates with a trusted CA certificate**
- **basicConstraints, ensuring the presence of the basicConstraints extension**
- **Revocation status, using OCSP**
- **extendedKeyUsage properties, when the certificate is used for OCSP**

**The certificate path is also validated when a certificate is imported. This validation includes a check of the certificate chain, and the keys of each of the certificates in the chain. The validity period of the certificate is also checked at this time.**

**The TOE validates the extendedKeyUsage field according to the following rules:**

- *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
- *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
- *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
- *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

**When a certificate is used for secure channels, an OCSP server is contacted to verify that the certificate is still valid. If the validity of a certificate that is used for IPSec tunnel cannot be verified, the system rejects the certificate and drops the connection for IPSec tunnels.**

The TOE communicates with an OCSP responder. The OCSP server must also be configured with an OCSP response signing certificate issued by the CA server. The TOE issues a status request to an OCSP responder and suspends the acceptance of the certificate until the responder provides a response.

The OCSP responder receives the request from the client and checks that the message is properly formed and if the responder is able to respond to the service request. Then it checks if the request contains the correct information needed for the service desired. If all conditions are satisfied, the responder returns a definitive response to the OCSP client. The OCSP responder is required to provide a basic response of GOOD, REVOKED, or UNKNOWN.

The GOOD state is the desired response as it indicates the certificate has not been revoked. The REVOKED state indicates that the certificate has been revoked. The UNKNOWN state indicates the responder does not have information about the certificate in question.

The certificate revocation checking is performed on the local and Intermediate certificates.

**Verdict:**

**PASS.**

#### 5.2.2.1.3 FIA\_X509\_EXT.1/REV (VPNGW)

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.

#### 5.2.2.2 FIA\_X509\_EXT.2 X.509 CERTIFICATE AUTHENTICATION

##### 5.2.2.2.1 FIA\_X509\_EXT.2 TSS

The evaluator shall check the TSS to ensure that it describes how the TOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.

**Evaluator Findings:**

The evaluator checked the TSS and ensured that it describes how the TOE chooses which certificates to use, and any necessary instructions in the AGD for configuring the operating environment so that the TOE can use the certificates.

The relevant information is found in the following section(s): TOE Summary Specification  
**FIA\_X509\_EXT.2.**

Upon investigation, the evaluator found that the TSS states that: **Certificates are used for IPsec, TLS (HTTPS).**

**Certificates used for IPsec are assigned a name when imported and are selected by name when the parameters are selected for an IPsec Security Policy.**

**The certificate used for TLS/HTTPS is called the 'HTTPS Management Certificate' and is created for that purpose on the TOE device.**

The evaluator shall examine the TSS and confirmed that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

**Evaluator Findings:**

The evaluator examined the TSS and confirmed that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

The relevant information is found in the following section(s): TOE Summary Specification **FIA\_X509\_EXT.2.**

Upon investigation, the evaluator found that the TSS states that: **If the validity of a certificate cannot be verified, the system rejects the certificate and drops the connection.**

The evaluator shall verify that any distinctions between trusted channels are described. If the requirement that the administrator is able to specify the default action, then the evaluator shall ensure that the AGD contains instructions on how this configuration action is performed.

**Evaluator Findings:**

IPSec is claimed as the only trusted channel, hence, there are no distinctions to report in the ST.

The evaluator examined the AGD and found no instructions to configure the default action of the trusted channel establishment when validity check cannot be completed. The evaluator also looked for any such configuration during testing and did not find any possible configuration.

**Verdict:**

**PASS.**

**5.2.2.2.2 FIA\_X509\_EXT.2 AGD**

The evaluator shall also ensure that the guidance documentation describes the configuration required in the operating environment so the TOE can use the certificates. The guidance documentation shall also include any required configuration on the TOE to use the certificates. The guidance document shall also describe the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

## Evaluator Findings:

The evaluator also ensured that the AGD describes the configuration required in the operating environment so the TOE can use the certificates. The AGD also includes any required configuration on the TOE to use the certificates. The AGD also describes the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

The relevant information is found in the following section(s): **Revocation Checking using OCSP and Managing Certificates**

Upon investigation, the evaluator examined the AGD section '**Revocation Checking using OCSP**' and found that the AGD describes the configuration required in the operating environment so the TOE can use the certificates. The AGD states that:

**'The device communicates with an OCSP responder. The OCSP server must also be configured with an OCSP response signing certificate issued by the CA server. The OCSP responder can be a CA server or another server that communicates with the CA server to determine the certificate status. The device issues a status request to an OCSP responder and suspends the acceptance of the certificate until the responder provides a response.'**

**Furthermore**, the evaluator examined the AGD section '**Managing Certificates**' and found that the AGD describes any required configuration on the TOE to use the certificates. The AGD states that: **'The device automatically uses all valid certificates without needing any additional configuration. It does not permit the use of invalid, expired, or unverified certificates.'**

Finally, the evaluator examined the AGD section '**Revocation Checking using OCSP**' and found that the AGD describes some steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The AGD states that:

**'If the validity of a certificate cannot be verified, the system rejects the certificate and drops the connection. Below are some steps a Security Administrator can follow if a connection cannot be established during the validity check of a certificate-**

- 1. Check OCSP Responder URL: Verify that the OCSP responder URL specified in the certificate is correct and accessible.**
- 2. Test OCSP Responder Connectivity: Try accessing the OCSP responder URL directly using to confirm that the responder is reachable from your network.**
- 3. Check System Time and Date: Ensure that the system time and date are correct, as incorrect time settings can affect certificate validity checks.**
- 4. Check OCSP Responder Status: Ensure that the OCSP responder service is up and running.**
- 5. Review Logs: Examine system and application logs for any errors or warnings related to the certificate validation process. These logs can provide clues about what might be going wrong.'**

**Verdict:**

**PASS.**



### 5.2.2.2.3 FIA\_X509\_EXT.2 (VPNGW)

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to support its use for IPsec at a minimum. The evaluator shall ensure that all evaluation of this SFR is performed against its use in IPsec communications as well as any other supported usage.

### 5.2.2.3 FIA\_X509\_EXT.3 EXTENDED: X509 CERTIFICATE REQUESTS

#### 5.2.2.3.1 FIA\_X509\_EXT.3 TSS

If the ST author selects "device-specific information", the evaluator shall verify that the TSS contains a description of the device-specific fields used in certificate requests.

Evaluator Findings:
The 'device-specific information' is not selected by the ST author.

**Verdict:**

PASS.

#### 5.2.2.3.2 FIA\_X509\_EXT.3 AGD

The evaluator shall check to ensure that the guidance documentation contains instructions on requesting certificates from a CA, including generation of a Certificate Request. If the ST author selects "Common Name", "Organization", "Organizational Unit", or "Country", the evaluator shall ensure that this guidance includes instructions for establishing these fields before creating the Certification Request.

Evaluator Findings:
The evaluator checked and ensured that the AGD contains instructions on requesting certificates from a CA, including generation of a Certificate Request. The evaluator ensured that the AGD includes instructions for establishing the "Common Name", "Organization", and "Country" fields before creating the Certification Request.
The relevant information is found in the following section(s): <b>Generating a Certificate Signing Request</b>
Upon investigation, the evaluator found that the claimed AGD section contains instructions for establishing the "Common Name", "Organization", and "Country" fields before creating the Certification Request and on requesting certificates from a CA. The AGD states that:
<b>To generate a certificate signing request</b>
<ol style="list-style-type: none"><li>1. <b>Navigate to Device   Settings &gt; Certificates.</b></li><li>2. <b>Click New Signing Request. The Certificate dialog displays.</b></li></ol>

3. Enter an alias name for the certificate in the Certificate Alias field.
4. Create a Distinguished Name (DN) using the drop-down menus shown in table below; then enter information for the certificate in the associated fields.

For each DN, you can select your country from the associated drop-down menu. For all other components, enter the information in the associated field.

Drop-down menu	Select appropriate information
<b>Country</b>	Country (default)
	State
	Locality or County
	Company or Organization
<b>State</b>	Country
	State (default)
	Locality, City, or County
	Company or Organization
	Department
<b>Locality, City, or County</b>	Locality, City, or County (default)
	Company or Organization
	Department
	Group
	Team
<b>Company or Organization</b>	Company or Organization (default)
	Department
	Group
	Team
	Common Name
	Serial Number
	E-Mail Address
<b>Department</b>	Department (default)
	Group
	Team
	Common Name
	Serial Number
	E-Mail Address

<b>Group</b>	<b>Group (default)</b>
	Team
	Common Name
	Serial Number
	E-Mail Address

---

<b>Team</b>	<b>Team (default)</b>
	Common Name
	Serial Number
	E-Mail Address

---

<b>Common Name</b>	<b>Common Name (default)</b>
	Serial Number
	E-Mail Address

---

As you enter information for the components, the Distinguished Name (DN) is created in the Subject Distinguished Name field.

5. Optionally, you can also attach a SUBJECT ALTERNATIVE NAME to the certificate after selecting the type from the drop-down menu:

- Domain Name
- Email Address
- IPv4 Address

6. Select a signature algorithm from the Signature Algorithm drop-down menu:

- SHA1 (default)
- MD5
- SHA256
- SHA384
- SHA512

9. Select a subject key type from the Subject Key Type drop-down menu:

**RSA (default)** A public key cryptographic algorithm used for encrypting data.

**ECDSA** Encrypts data using the Elliptic Curve Digital Signature Algorithm, which has a high strength-per-key-bit security.

---

10. Select a subject key size or curve from the Subject Key Size/Curve drop-down menu.

Not all key sizes or curves are supported by a Certificate Authority, therefore, you should check with your CA for supported key sizes.

If you selected a key type of:

RSA, select a key size	ECDSA, select a curve
2048 bits	prime256v1: X9.62.SECP curve over a 256 bit prime field (default)
3072 bits	secp384r1: NIST/SECP curve over a 384 bit prime field
4096 bits	secp521r1: NIST/SECP curve over a 521 bit prime field

11. Click Generate to create a certificate signing request file.  
When the Certificate Signing Request is generated, a message describing the result is displayed and a new entry appears in the Certificates table with the type Pending request.
12. Click the Export icon. The Export Certificate Request dialog displays.
13. Click the Export icon to download the file to your computer. An Opening <certificate> dialog displays.
14. Click OK to save the file to a directory on your computer.  
You have generated the Certificate Request that you can send to your Certificate Authority for validation.
15. Click the Upload icon to upload the signed certificate for a signing request. The Upload Certificate dialog is displayed.
16. Click Choose File to select a file.
17. Select the file and click Open.
18. Click UPLOAD.

**Verdict:**

PASS.

5.2.2.3.3 FIA\_X509\_EXT.3 (VPNGW)

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.

5.2.3 SECURITY MANAGEMENT (FMT)

5.2.3.1 FMT\_MOF.1/SERVICES MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

5.2.3.1.1 FMT\_MOF.1/SERVICES TSS

**Evaluator Findings:**

The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall ensure the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed.

**Evaluator Findings:**

The evaluator examined the TSS and ensured that, for non-distributed TOEs, it lists the services the Security Administrator is able to start and stop and how that how that operation is performed.

The relevant information is found in the following section(s): **FMT\_MOF.1/Services**  
Upon investigation, the evaluator found **that the security administrator has ability to modify (enable/disable) transmission of audit records to an external audit server.**

**Verdict:**

**PASS.**

**5.2.3.1.2 FMT\_MOF.1/SERVICES AGD**

For distributed TOEs see Section 2.4.1.2.

**Evaluator Findings:**

The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed.

**Evaluator Findings:**

The evaluator examined the AGD and ensured that, for non-distributed TOEs, it describes how the TSS lists the services the Security Administrator is able to start and stop and how that how that operation is performed.

The relevant information is found in the following section(s): **Enabling Syslog Servers and Disabling Syslog Servers**

Upon investigation, the evaluator found that the claimed AGD sections states that:

**'To enable a single Syslog server:**

**Navigate to Device | log | Syslog and select the toggle button in the Enable column.**

**To enable all Syslog servers, select the Syslog servers and click Enable All.**

**To disable a single Syslog server:**

Navigate to Device | log | Syslog and deselect the toggle button in the Enable column.

To disable all Syslog servers, select the Syslog servers and click Disable All.'

**Verdict:**

PASS.

5.2.3.2 FMT\_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA

5.2.3.2.1 FMT\_MTD.1/CRYPTOKEYS TSS

**Evaluator Findings:**

The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall ensure the TSS lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

**Evaluator Findings:**

The evaluator examined the TSS and ensured that, for non-distributed TOEs, it lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

The relevant information is found in the following section(s): **FMT\_MTD.1/CryptoKeys**

Upon investigation, the evaluator found that the TSS states that:

**The security administrator is able to Generate and delete cryptographic keys (generate and delete the cryptographic keys associated with CSRs).**

**Verdict:**

PASS.

5.2.3.2.2 FMT\_MTD.1/CRYPTOKEYS AGD

For distributed TOEs see Section 2.4.1.2.

**Evaluator Findings:**

The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

#### Evaluator Findings:

The evaluator examined the AGD and ensured that, for non-distributed TOEs, it lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

The relevant information is found in the following section(s): **Generating a Certificate Signing Request** and **Deleting a Certificate Signing Request**

Upon investigation, the evaluator found that the claimed AGD sections states that:

**'The Security Administrator can manage generating and importing keys. The Security Administrator can manage deleting keys.'**

The evaluator summarizes that the AGD sections **'Generating a Certificate Signing Request'** and **'Deleting a Certificate Signing Request'** describe steps on how generation of keys and deletion of keys is carried on the device.

#### Verdict:

PASS.

#### 5.2.3.2.3 FMT\_MTD.1/CRYPTOKEYS (VPNGW)

---

There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.

## 6 SECURITY ASSURANCE REQUIREMENTS

### 6.1 ADV: DEVELOPMENT

#### 6.1.1 BASIC FUNCTIONAL SPECIFICATION (ADV\_FSP.1)

##### 6.1.1.1 (5.2.1.1) EVALUATION ACTIVITY

The evaluator shall examine the interface documentation to ensure it describes the purpose and method of use for each TSFI that is identified as being security relevant.

In this context, TSFI are deemed security relevant if they are used by the administrator to configure the TOE, or to perform other administrative functions (e.g. audit review or performing updates). Additionally, those interfaces that are identified in the ST, or guidance documentation, as adhering to the security policies (as presented in the SFRs), are also considered security relevant. The intent is that these interfaces will be adequately tested and having an understanding of how these interfaces are used in the TOE is necessary to ensure proper test coverage is applied.

The set of TSFI that are provided as evaluation evidence are contained in the Administrative Guidance and User Guidance.

#### **Evaluator Findings:**

TOE Design information that can be made public is available in the guidance documentation and in the ST. Any sensitive or proprietary information required by this protection profile is not to be made public.

It is not necessary to provide a complete specification of the TSFIs. For NDcPP, additional “functional specification” documentation is not necessary because this requirement is satisfied by multiple other documents (AGD, TSS, and Testing). All associated activities are covered in the Test Report, ST, and AGD documents.

NDcPP2.2e, section 7.2.1 states that:

“For this cPP, the Evaluation Activities for this family focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation.”

All of the above information is applicable to the ADV Evaluation Activities (5.2.1.1, 5.2.1.2, and 5.2.1.3) in NDcPP2.2e-SD.

The evaluator examined the ST (Security Target) and the AGD (interface documentation) to verify that it describes the purpose and method of use for each TSFI that is identified as being security relevant. The evaluator examined the entire AGD. The evaluator verified the AGD describes the purpose and method of use for each security relevant TSFI by verifying the AGD satisfies all the AGD Evaluation Activities.



During testing, the evaluator used the product and its interfaces extensively and did not find any areas that were deficient.

**Verdict:**

**PASS.**

**6.1.1.2 (5.2.1.2) EVALUATION ACTIVITY**

The evaluator shall check the interface documentation to ensure it identifies and describes the parameters for each TSFI that is identified as being security relevant.

**Evaluator Findings:**

The evaluator checked the interface documentation (AGD) and ensured it identifies and describes the parameters for each TSFI that is identified as being security relevant. This is covered in the previous evaluation activity above.

**Verdict:**

**PASS.**

**6.1.1.3 (5.2.1.3) EVALUATION ACTIVITY**

The evaluator shall examine the interface documentation to develop a mapping of the interfaces to SFRs.

The evaluator uses the provided documentation and first identifies, and then examines a representative set of interfaces to perform the EAs presented in Section 2, including the EAs associated with testing of the interfaces.

It should be noted that there may be some SFRs that do not have an interface that is explicitly “mapped” to invoke the desired functionality. For example, generating a random bit string, destroying a cryptographic key that is no longer needed, or the TSF failing to a secure state, are capabilities that may be specified in SFRs, but are not invoked by an interface.

However, if the evaluator is unable to perform some other required EA because there is insufficient design and interface information, then the evaluator is entitled to conclude that an adequate functional specification has not been provided, and hence that the verdict for the ADV\_FSP.1 assurance component is a ‘fail’.

**Evaluator Findings:**

The evaluator examined the interface documentation to develop a mapping of the interfaces to SFRs.

The evaluator used the provided documentation to first identify, and then examine a representative set of interfaces to perform the EAs presented in Section 2, including the EAs associated with testing of the interfaces.

This is covered in the previous evaluation activity above.

**Verdict:**

**PASS.**

## 6.2 AGD: GUIDANCE DOCUMENTS

### 6.2.1 OPERATIONAL USER GUIDANCE (AGD\_OPE.1)

#### 6.2.1.1 (5.3.1.1) EVALUATION ACTIVITY

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

**Evaluator Findings:**

The evaluator checked the requirements above are met by the AGD. The AGD is distributed to administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration. Upon investigation, the evaluator found that the CC guidance will be published with the CC certificate on [www.niap-ccevs.org](http://www.niap-ccevs.org).

**Verdict:**

**PASS.**

#### 6.2.1.2 (5.3.1.2) EVALUATION ACTIVITY

The evaluator shall ensure that the Operational guidance is provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.

**Evaluator Findings:**

The evaluator ensured that the AGD is provided for every Operational Environment that the product supports as claimed in the Security Target. The section titled **Supported Platforms** of the AGD was used to determine the verdict of this assurance activity. The AGD specifies that the platforms supported are:

Appliance Series	Appliance Model	Operational Environment	Microarchitecture
TZ	TZ 670	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570W	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570P	Marvell CN9130	Quad Core Armv8 Cortex-A72

	TZ 470	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 470W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270W	Marvell 88F7040	Quad core Armv8 Cortex-A72
NSa	NSa 2700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 3700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 4700	Intel Xeon D-2123IT	Skylake
	NSa 5700	Intel Xeon D-2123IT	Skylake
	NSa 6700	Intel Xeon D-2123IT	Skylake
NSsp	NSsp 10700	Intel Xeon D-2166NT	Skylake
	NSsp 11700	Intel Xeon D-2166NT	Skylake
	NSsp 13700	Intel Xeon D-2187NT	Skylake
<hr/>			
<b>Appliance Series</b>	<b>Appliance Model</b>	<b>Operational Environment</b>	
NSv	NSv 270	ESXi 7.0 and 8.0 on Dell PowerEdge R640 (Running on Intel Xeon Silver 4208 (Cascade Lake))	
	NSv 470		
	NSv 870		

**Verdict:**

**PASS.**

**6.2.1.3 (5.3.1.3) EVALUATION ACTIVITY**

The evaluator shall ensure that the Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.

**Evaluator Findings:**

The AGD section “Enabling NDPP Mode” states that:  
“The appliance only uses cryptographic engines which are hard-coded and non-configurable. In the NDPP mode, the cryptographic algorithms are limited to the only ones that are supported by the Common Criteria (CC). These algorithms were tested as a part of the evaluation of the product.”

**Verdict:**

PASS.

6.2.1.4 (5.3.1.4) EVALUATION ACTIVITY

The evaluator shall ensure the Operational guidance makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs.

**Evaluator Findings:**

The entire AGD was used to determine the verdict of this work unit. Each section in the AGD indicates tested options. Additionally, the section titled ‘**Product Functionality Not Included in the Scope of the Evaluation**’ specifies features that are not assessed and tested by the EAs. The evaluator ensured the AGD makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs.

**Verdict:**

PASS.

6.2.1.5 (5.3.1.5) EVALUATION ACTIVITY [TD0536]

In addition, the evaluator shall ensure that the following requirements are also met:

- The AGD shall contain instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.
- [TD0536] The documentation must describe the process for verifying updates to the TOE for each method selected for FPT\_TUD\_EXT.1.3 in the Security Target. The evaluator shall verify that this process includes the following steps:
  - Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).
  - Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes instructions that describe at least one method of validating the hash/digital signature.
- The TOE will likely contain security functionality that does not fall in the scope of evaluation under this cPP. The AGD shall make it clear to an administrator which security functionality is covered by the Evaluation Activities.

## Evaluator Findings:

The evaluator verified the AGD contains instructions for configuring any cryptographic implementations in **Enabling NDPP Mode**.

The relevant information is found in the following section(s): **Enabling NDPP Mode, Generating a Certificate Signing Request** and **Firmware Upgrade**

Upon investigation, the evaluator found that the section '**Enabling NDPP Mode**' of the AGD states that:

**'A SonicWall network security appliance can be enabled to be compliant with the Network Device Protection Profile (NDPP), but certain firewall configurations are either not allowed or are required. The appliance only uses cryptographic engines which are hard-coded and non-configurable. In the NDPP mode, the cryptographic algorithms are limited to the only ones that are supported by the Common Criteria (CC). These algorithms were tested as a part of the evaluation of the product.'**

The evaluator also verified sections '**Firmware Upgrade**' of the AGD and found that the AGD describes the process for verifying updates to the TOE for the method selected for FPT\_TUD\_EXT.1.3 in the Security Target. The AGD also states the instructions for obtaining the update itself and for initiating the update process as well as discerning whether the process was successful or unsuccessful. The AGD states that:

**'To upload new firmware**

- 1. Download the SonicOS firmware image file from MySonicWall and save it to a location on your local computer.**
- 2. Point your browser to the appliance IP address and log in as an administrator.**
- 3. In the DEVICE view, on the Settings > Firmware and Settings page, on the Firmware & Local Backups screen, click Upload Firmware.**
- 4. In the Backup of current settings popup dialog, click OK to continue the firmware upload.**
- 5. In the Upload Firmware dialog, browse to the location where you saved the SonicOS firmware image file, select the file, and click Upload.**

The digital signature on the firmware is automatically verified using the SonicWall public key. This key is appended to each firmware image made available to customers and is used to verify the new firmware. When a new firmware image is loaded on the physical appliances, the cryptographic module verifies the ECDSA signed SHA-256 hash of the image. When a new image is loaded on a virtual appliance, the cryptographic module verifies the RSA signed SHA-256 hash of the image.

- If the signature verification succeeds, the firmware is automatically installed.
  - If the signature verification fails, the firmware is not loaded, and an error appears.
  - Uploading the same firmware is disallowed.
  - After the firmware finishes uploading, it is displayed in the table on the Firmware & Local Backups screen.
  - Firmware & Local Backup tab now shows the Current Firmware Version and recently Uploaded Firmware
  - Version which is inactive image.
- 6. Click the Boot icon in the Uploaded Firmware Version row and select Boot firmware with Current Configuration.**

**Note: Once the new version is installed as the boot image, the previously installed image gets replaced.**

**7. In the Warning dialog box, click OK. The appliance restarts and displays the login page.**

**Note: No functionality will cease during the update process. The device will remain fully operational until the administrator reboots the product.**

**8. Enter your username and password. Your new SonicOS image version information is displayed on the Settings > Status page.**

Additionally, the section titled **'Product Functionality Not Included in the Scope of the Evaluation'** specifies features that are not assessed and tested by the EAs.

**Verdict:**

**PASS.**

---

## 6.2.2 PREPARATIVE PROCEDURES (AGD\_PRE.1)

---

### 6.2.2.1 (5.3.2.1) EVALUATION ACTIVITY

The evaluator shall examine the Preparative procedures to ensure they include a description of how the Security Administrator verifies that the operational environment can fulfil its role to support the security functionality (including the requirements of the Security Objectives for the Operational Environment specified in the Security Target).

The documentation should be in an informal style and should be written with sufficient detail and explanation that they can be understood and used by the target audience (which will typically include IT staff who have general IT experience but not necessarily experience with the TOE product itself).

<b>Evaluator Findings:</b>
The evaluator examined the Preparative procedures to ensure they include a description of how the administrator verifies that the operational environment can fulfil its role to support the security functionality. The evaluator reviewed the sections titled <b>'Operational Environment'</b> of the AGD. The evaluator found that this section describe how the Operational Environment must meet:

'The following environmental components are required to operate the TOE (Target of Evaluation) in the evaluated configuration:

- **TOE: Sonicwall SonicOS 7.0.1 running on a claimed physical appliance or a virtual appliance, typically deployed as a gateway between two networks, such as LAN and the internet.**
- **Management workstation: Any IT environment management workstation.**
- **Remote Logging: Audit Server supporting syslog protocol with an IPsec peer supporting IKEv2 and ESP.**
- **Management Console: Any computer that provides a supported browser to access administrative web GUI via HTTPS and direct serial connection providing administrative CLI access.**
- **VPN Gateway: VPN connections via IPsec.**
- **WAN/Internet: External IP interface.**
- **LAN/Internal: Internal IP interface.'**

**Verdict:**

**PASS.**

**6.2.2.2 (5.3.2.2) EVALUATION ACTIVITY**

The evaluator shall examine the Preparative procedures to ensure they are provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.

**Evaluator Findings:**

The evaluator checked the requirements above are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that the AGD describes each of the devices in the operating environment, including,

- **TOE: Sonicwall SonicOS 7.0.1 running on a claimed physical appliance or a virtual appliance, typically deployed as a gateway between two networks, such as LAN and the internet.**
- **Management workstation: Any IT environment management workstation.**
- **Remote Logging: Audit Server supporting syslog protocol with an IPsec peer supporting IKEv2 and ESP.**
- **Management Console: Any computer that provides a supported browser to access administrative web GUI via HTTPS and direct serial connection providing administrative CLI access.**
- **VPN Gateway: VPN connections via IPsec.**
- **WAN/Internet: External IP interface.**
- **LAN/Internal: Internal IP interface.**

The section titled **Supported Platforms** of AGD identifies the following supported platform:

<b>Appliance Series</b>	<b>Appliance Model</b>	<b>Operational Environment</b>	<b>Microarchitecture</b>
TZ	TZ 670	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570W	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570P	Marvell CN9130	Quad Core Armv8 Cortex-A72

	TZ 470	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 470W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270W	Marvell 88F7040	Quad core Armv8 Cortex-A72
NSa	NSa 2700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 3700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 4700	Intel Xeon D-2123IT	Skylake
	NSa 5700	Intel Xeon D-2123IT	Skylake
	NSa 6700	Intel Xeon D-2123IT	Skylake
NSsp	NSsp 10700	Intel Xeon D-2166NT	Skylake
	NSsp 11700	Intel Xeon D-2166NT	Skylake
	NSsp 13700	Intel Xeon D-2187NT	Skylake
<b>Appliance Series      Appliance Model      Operational Environment</b>			
NSv	NSv 270	ESXi 7.0 and 8.0 on Dell PowerEdge R640 (Running on Intel Xeon Silver 4208 (Cascade Lake))	
	NSv 470		
	NSv 870		

**Verdict:**

**PASS.**

**6.2.2.3 (5.3.2.3) EVALUATION ACTIVITY**

The evaluator shall examine the preparative procedures to ensure they include instructions to successfully install the TSF in each Operational Environment.

Evaluator Findings:
<p>The evaluator checked the requirements are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that AGD describes all of the instructions necessary to install and configure the TOE to work in the target operating environment, including:</p> <ul style="list-style-type: none"> <li>• <b>Initial Setup</b></li> <li>• <b>Configuring Administrative Accounts and Passwords</b></li> <li>• <b>Configuring HTTPS and TLS Connections</b></li> <li>• <b>Configuring the Remote Syslog Server</b></li> <li>• <b>Configuring Audit Log Options</b></li> <li>• <b>Configuring Event Logging</b></li> <li>• <b>Configuring IPsec</b></li> <li>• <b>Configuring Access Rules and App Rules</b></li> <li>• <b>Configuring Certificate Validation</b></li> </ul>

**Verdict:**

**PASS.**



#### 6.2.2.4 (5.3.2.4) EVALUATION ACTIVITY

The evaluator shall examine the preparative procedures to ensure they include instructions to manage the security of the TSF as a product and as a component of the larger operational environment.

##### Evaluator Findings:

The evaluator ensured the preparative procedures include instructions to manage the security of the TSF as a product and as a component of the larger operational environment. The entire AGD was used to determine the verdict of this work unit. The same commands, configurations, and interfaces used to install the TOE are also used for ongoing management, so this is satisfied by AGD\_PRE.1 Test #3

##### Verdict:

PASS.

#### 6.2.2.5 (5.3.2.5) EVALUATION ACTIVITY

In addition, the evaluator shall ensure that the following requirements are also met. The preparative procedures must:

- a) include instructions to provide a protected administrative capability; and
- b) identify TOE passwords that have default values associated with them and instructions shall be provided for how these can be changed.

##### Evaluator Findings:

The evaluator ensured the preparative procedures include instructions to provide a protected administrative capability and changing default passwords. The section titled **Product Administration** and the associated sub-sections were used to determine the verdict of this work unit.

##### Verdict:

PASS.

### 6.3 AVA: VULNERABILITY ASSESSMENT

#### 6.3.1 VULNERABILITY SURVEY (AVA\_VAN.1)

##### 6.3.1.1 (5.6.1.1) EVALUATION ACTIVITY (DOCUMENTATION) [TD0547]

In addition to the activities specified by the CEM in accordance with Table 2, the evaluator shall perform the following activities.

*The evaluator shall examine the documentation outlined below provided by the developer to confirm that it contains all required information. This documentation is in addition to the documentation already required to be supplied in response to the EAs listed previously.*

The developer shall provide documentation identifying the list of software and hardware components that compose the TOE. Hardware components should identify at a minimum the processors used by the

TOE. Software components include applications, the operating system and other major components that are independently identifiable and reusable (outside of the TOE), for example a web server, protocol or cryptographic libraries, (independently identifiable and reusable components are not limited to the list provided in the example). This additional documentation is merely a list of the name and version number of the components and will be used by the evaluators in formulating vulnerability hypotheses during their analysis.

If the TOE is a distributed TOE then the developer shall provide:

- a. documentation describing the allocation of requirements between distributed TOE components as in [NDcPP, 3.4]
- b. a mapping of the auditable events recorded by each distributed TOE component as in [NDcPP, 6.3.3]
- c. additional information in the Preparative Procedures as identified in the refinement of AGD\_PRE.1 in additional information in the Preparative Procedures as identified in 3.4.1.2 and 3.5.1.2.

#### **Evaluator Findings:**

The evaluator collected this information from the developer which was used to feed into the Public Domain Search. Refer to evaluator findings in the evaluation activity below.

#### **Verdict:**

**PASS.**

#### **6.3.1.2 (5.6.1.2) EVALUATION ACTIVITY**

The evaluator formulates hypotheses in accordance with process defined in Appendix A. The evaluator documents the flaw hypotheses generated for the TOE in the report in accordance with the guidelines in Appendix A.3. The evaluator shall perform vulnerability analysis in accordance with Appendix A.2. The results of the analysis shall be documented in the report according to Appendix A.3.

#### **Evaluator Findings:**

The evaluator documented their analysis and testing of potential vulnerabilities with respect to this requirement. Public searches were performed against all keywords found within the Security Target and AGD that may be applicable to specific TOE components. This included protocols, TOE software version, and TOE hardware to ensure sufficient coverage under AVA. The evaluator searched the Internet for potential vulnerabilities in the TOE using the web sites listed below. The sources of the publicly available information are provided below:

- <https://nvd.nist.gov/view/vuln.search>
- <http://cve.mitre.org/cve>
- <https://www.cvedetails.com/vulnerability-search.php>
- <https://www.kb.cert.org/vuls/search/>
- [www.exploitsearch.net](http://www.exploitsearch.net)
- [www.securiteam.com](http://www.securiteam.com)
- <http://nessus.org/plugins/index.php?view=search>
- <http://www.zerodayinitiative.com/advisories>
- <https://www.exploit-db.com>
- <https://www.rapid7.com/db/vulnerabilities>
- <https://psirt.global.sonicwall.com/vuln-list>

The evaluator performed the public domain vulnerability searches using the following key words. The search was performed on **December 24, 2024**.

- Marvell CN9130
- Marvell 88F7040
- cpe:/:intel:xeon\_d-2123it:-
- cpe:/:intel:xeon\_d-2166nt:-
- cpe:/:intel:xeon\_d-2187nt:-
- cpe:/:intel:xeon\_silver\_4208:-
- cpe:/:sonicwall:sonicos:7.0.1
- cpe:/:openssl:openssl:1.1.1c
- cpe:/:sonicwall:tz670:-
- cpe:/:sonicwall:tz570:-
- cpe:/:sonicwall:tz570w:-
- cpe:/:sonicwall:tz570p:-
- cpe:/:sonicwall:tz470:-
- cpe:/:sonicwall:tz470w:-
- cpe:/:sonicwall:tz370:-
- cpe:/:sonicwall:tz370w:-
- cpe:/:sonicwall:tz270:-
- cpe:/:sonicwall:tz270w:-
- cpe:/:sonicwall:nsa\_2700:-
- cpe:/:sonicwall:nsa\_3700:-
- cpe:/:sonicwall:nsa\_4700:-
- cpe:/:sonicwall:nsa\_5700:-
- cpe:/:sonicwall:nsa\_6700:-
- cpe:/:sonicwall:nssp\_10700:-
- cpe:/:sonicwall:nssp\_11700:-
- cpe:/:sonicwall:nssp\_13700:-
- cpe:/:sonicwall:nsv\_270:-
- cpe:/:sonicwall:nsv\_470:-
- cpe:/:sonicwall:nsv\_870:-
- SonicOS/X

The evaluation lab examined each result provided from NVD and Exploit Search to determine if the current TOE version or component within the environment was vulnerable. Based upon the analysis, any issues found that were generated were patched in the TOE version and prior versions, mitigating the risk factor.

**Verdict:**

**PASS.**

## 7 DETAILED TEST CASES (TEST ACTIVITIES)

### 7.1 AUDIT

#### 7.1.1 FAU\_GEN.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the table of audit events and administrative actions listed above. This should include all instances of an event: for instance, if there are several different I&amp;A mechanisms for a system, the FIA_UIA_EXT.1 events must be generated for each mechanism. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. If HTTPS is implemented, the test demonstrating the establishment and termination of a TLS session can be combined with the test for an HTTPS session. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the guidance documentation, and that the fields in each audit record have the proper entries.</p> <p>Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.</p>
<b>Notes</b>	<p><b>NIT Technical Decision (TD0639) for Clarification for NTP MAC Keys.</b></p> <p>The SFRs FAU_GEN.1, FCS_CKM.4 and FPT_SKP_EXT.1 shall be applied to all cryptographic keys that are related to secure communication (i.e. related to FTP_TRP.1, FTP_ITC.1, FPT_ITT.1). The NTP requirements have been introduced in NDcPP V2.1 as a rather 'standalone' set of requirements with 'no audit requirements' specified in the ECD section for FCS_NTP_EXT.1 and no dependencies on FCS_CKM - in contrast to the corresponding sections for secure communication protocols like TLS. As NTP keys are not intended to be used for encryption of sensitive information, the level of protection is different compared to other pre-shared keys. It has therefore not been intended that NTP keys are treated as other pre-shared keys in the context of NDcPP.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Trigger each auditable event on the TOE. Verify that each audit record is generated and contains the required information.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should accurately generate audit records for all the required auditable events described in the ST under the FAU_GEN.1.1, FAU_GEN.1.2, FAU_GEN.1.1/VPN, FAU_GEN.1.2/VPN, FAU_GEN.1.1/IPS, FAU_GEN.1.2/IPS, FAU_GEN.2.1.</li> <li>• The TOE can generate audit records for establishment and termination of a channel for HTTPS/TLS.</li> </ul>

	<ul style="list-style-type: none"> <li>• The audit records generated should match the format specified in the guidance documentation.</li> <li>• Evidence- Audit logs generated for each SFR.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The audit records associated with each test case are recorded. Each required audit record is generated by the TOE. This meets the testing requirement.

### 7.1.2 FAU\_GEN.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of auditable events to TOE components in the Security Target. For all events involving more than one TOE component when an audit event is triggered, the evaluator has to check that the event has been audited on both sides (e.g. failure of building up a secure communication channel between the two components). This is not limited to error cases but includes also events about successful actions like successful build up/tear down of a secure communication channel between TOE components.</p> <p>Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.</p>
<b>Notes</b>	<p><b>NIT Technical Decision (TD0639) for Clarification for NTP MAC Keys.</b></p> <p>The SFRs FAU_GEN.1, FCS_CKM.4 and FPT_SKP_EXT.1 shall be applied to all cryptographic keys that are related to secure communication (i.e. related to FTP_TRP.1, FTP_ITC.1, FPT_ITT.1). The NTP requirements have been introduced in NDcPP V2.1 as a rather 'standalone' set of requirements with 'no audit requirements' specified in the ECD section for FCS_NTP_EXT.1 and no dependencies on FCS_CKM - in contrast to the corresponding sections for secure communication protocols like TLS. As NTP keys are not intended to be used for encryption of sensitive information, the level of protection is different compared to other pre-shared keys. It has therefore not been intended that NTP keys are treated as other pre-shared keys in the context of NDcPP.</p>
<b>Pass/Fail with Explanation</b>	The TOE is not distributed; hence, this activity is not applicable.

### 7.1.3 FAU\_GEN.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	This activity should be accomplished in conjunction with the testing of FAU_GEN.1.1.
<b>Pass/Fail with Explanation</b>	Pass. Testcase FAU_GEN.1 Test#1 covers this requirement.

#### 7.1.4 FAU\_GEN.2 TEST #2

Item	Data
<b>Test Assurance Activity</b>	For distributed TOEs the evaluator shall verify that where auditable events are instigated by another component, the component that records the event associates the event with the identity of the instigator. The evaluator shall perform at least one test on one component where another component instigates an auditable event. The evaluator shall verify that the event is recorded by the component as expected and the event is associated with the instigating component. It is assumed that an event instigated by another component can at least be generated for building up a secure channel between two TOE components. If for some reason (could be e.g. TSS or Guidance Documentation) the evaluator would come to the conclusion that the overall TOE does not generate any events instigated by other components, then this requirement shall be omitted.
<b>Pass/Fail with Explanation</b>	The TOE is not distributed; hence, this activity is not applicable.

#### 7.1.5 FAU\_STG\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:</p> <p>Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE during several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing. The evaluator shall verify that the TOE is</p>

	capable of transferring audit data to an external audit server automatically without administrator intervention.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the syslog server with port and certificates.</li> <li>• Configure the TOE to send logs to the Syslog server.</li> <li>• Verify the version of the audit server.</li> <li>• Login and logout from the TOE to generate logs.</li> <li>• Confirm that audit logs were sent to the syslog server.</li> <li>• Examine traffic captured by packet capture to ensure it is not plaintext.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should support the transfer of audit data without admin intervention.</li> <li>• The communication between TOE and Syslog server should be encrypted.</li> <li>• Packet Capture should show that traffic between TOE and the Syslog server is not sent in plaintext.</li> <li>• TOE logs should show a successful Syslog connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE passes all audit traffic to the remote audit server through a secure channel without admin interference. The evaluator accurately records the specific software used on the audit server, including the name and version. This meets the testing requirements.

7.1.6 FAU\_STG\_EXT.1 TEST #2 (A)

Item	Data
<b>Test Assurance Activity</b>	<p>Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:</p> <p>Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that:</p> <ol style="list-style-type: none"> <li>1) The audit data remains unchanged with every new auditable event that should be tracked but that the audit data is recorded again after the local storage for audit data is cleared (for the option ‘drop new audit data’ in FAU_STG_EXT.1.3).</li> </ol>
<b>Pass/Fail with Explanation</b>	N/A. The option ‘drop new audit data’ is not selected in the ST.

7.1.7 FAU\_STG\_EXT.1 TEST #2 (B)

Item	Data
<b>Test Assurance Activity</b>	<p>Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:</p> <p>Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that:</p> <p>2) The existing audit data is overwritten with every new auditable event that should be tracked according to the specified rule (for the option '<b>overwrite previous audit records</b>' in FAU_STG_EXT.1.3)</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Verify the log count in DB on the console before reaching its maximum limit of 10000.</li> <li>• Export the local audit logs into a CSV file and verify the log count.</li> <li>• Verify the timestamp of the last 2500 logs of the exported file to identify the oldest audit entries.</li> <li>• Wait until the buffer gets full and the log limit reaches 10000.</li> <li>• Once the buffer is full, verify that 25% of the audit logs are deleted.</li> <li>• Verify that the deleted audit logs are the oldest audit entries.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should successfully allow the overwriting of old logs by new ones.</li> <li>• Evidence – snapshot showing the oldest logs are overwritten by the new logs</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test is passed because once the limit was reached the oldest audit record was overwritten. This meets the testing requirements.

#### 7.1.8 FAU\_STG\_EXT.1 TEST #2 (C)

Item	Data
<b>Test Assurance Activity</b>	<p>Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:</p> <p>Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the</p>



	evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that: 3) The TOE behaves as specified (for the option 'other action' in FAU_STG_EXT.1.3).
<b>Pass/Fail with Explanation</b>	N/A. The option 'other action' is not selected in the ST.

#### 7.1.9 FAU\_STG\_EXT.1 TEST #3

Item	Data
<b>Test Assurance Activity</b>	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:  Test 3: If the TOE complies with FAU_STG_EXT.2/LocSpace the evaluator shall verify that the numbers provided by the TOE according to the selection for FAU_STG_EXT.2/LocSpace are correct when performing the tests for FAU_STG_EXT.1.3
<b>Pass/Fail with Explanation</b>	N/A. FAU_STG_EXT.2/LocSpace SFR is not claimed in ST.

#### 7.1.10 FAU\_STG\_EXT.1 TEST #4

Item	Data
<b>Test Assurance Activity</b>	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement:  Test 4: For distributed TOEs, Test 1 defined above should be applicable to all TOE components that forward audit data to an external audit server. For the local storage according to FAU_STG_EXT.1.2 and FAU_STG_EXT.1.3 the Test 2 specified above shall be applied to all TOE components that store audit data locally. For all TOE components that store audit data locally and comply with FAU_STG_EXT.2/LocSpace Test 3 specified above shall be applied. The evaluator shall verify that the transfer of audit data to an external audit server is implemented.
<b>Pass/Fail with Explanation</b>	N/A. The TOE is not distributed.

### 7.1.11 FPT\_STM\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: If the TOE supports direct <b>setting of the time by the Security Administrator</b> then the evaluator uses the guidance documentation to set the time. The evaluator shall then use an available interface to observe that the time was set correctly.</p> <p>If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Confirm the current time on the TOE.</li> <li>• Set a new time on the TOE via the remote GUI.</li> <li>• Verify that the new time is set.</li> <li>• Verify that an audit log is generated for the time change.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should allow time to be set manually via GUI.</li> <li>• Evidence: Snapshot should show updated time.</li> <li>• TOE should generate logs for the time change.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE allows the administrative user to configure the time on the TOE. This meets the testing requirements.</p>

### 7.1.12 FPT\_STM\_EXT.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: If the TOE supports the <b>use of an NTP server</b>; the evaluator shall use the guidance documentation to configure the NTP client on the TOE, and set up a communication path with the NTP server. The evaluator will observe that the NTP server has set the time to what is expected. If the TOE supports multiple protocols for establishing a connection with the NTP server, the evaluator shall perform this test using each supported protocol claimed in the guidance documentation.</p> <p>If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the</p>

	different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
<b>Pass/Fail with Explanation</b>	N/A. The ST does not select use of an NTP server.

#### 7.1.13 FPT\_STM\_EXT.1 TEST #3 [TD0632]

Item	Data
<b>Test Assurance Activity</b>	<p>Test 3: [conditional] If the TOE obtains time from the underlying VS, the evaluator shall record the time on the TOE, modify the time on the underlying VS, and verify the modified time is reflected by the TOE. If there is a delay between the setting the time on the VS and when the time is reflected on the TOE, the evaluator shall ensure this delay is consistent with the TSS and Guidance.</p> <p><b>TD0632 has been applied.</b></p>
<b>Pass/Fail with Explanation</b>	N/A. The ST does not select 'obtain time from underlying VS'.

#### 7.1.14 FTP\_ITC.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report.</p> <p>The evaluator shall perform the following tests:            Test 1: The evaluators shall ensure that communications using each protocol with each authorized IT entity is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target.</p>
<b>Notes</b>	<b>The NiT has issued a technical decision (TD0572) for Restricting FTP_ITC.1 to only IP address identifiers.</b>

	<p>DNS resolution is not mandatory to support FTP_ITC.1. There are no resolution mandates or requirements in RFC 6125.</p> <p>The use of the dnsName identifiers in X.509 certificates must be supported by TOEs that claim FCS_DTLSC_EXT or FCS_TLSC_EXT when associated with FTP_ITC.1. Inability to parse dnsName identifiers shall be considered a failure to meet these requirements.</p>
<b>Test Steps</b>	This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications.
<b>Expected Results</b>	Communication with the external audit server through IPsec VPN tunnel is successful.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications. As the tests show, communication with the external audit server through secure IPsec tunnel is successful.

7.1.15 FTP\_ITC.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report.</p> <p>The evaluator shall perform the following tests:            Test 2: For each protocol that the TOE can initiate as defined in the requirement, the evaluator shall follow the guidance documentation to ensure that in fact the communication channel can be initiated from the TOE.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target.</p>
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0572) for Restricting FTP_ITC.1 to only IP address identifiers.</b></p> <p>DNS resolution is not mandatory to support FTP_ITC.1. There are no resolution mandates or requirements in RFC 6125.</p>

	The use of the dnsName identifiers in X.509 certificates must be supported by TOEs that claim FCS_DTLSC_EXT or FCS_TLSC_EXT when associated with FTP_ITC.1. Inability to parse dnsName identifiers shall be considered a failure to meet these requirements.
<b>Test Steps</b>	This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications.
<b>Expected Results</b>	Connection with the external audit server through IPsec VPN tunnel is initiated by the TOE.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications. As the tests show, the communication with the external audit server is initiated by the TOE.

**7.1.16 FTP\_ITC.1 TEST #3**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report.</p> <p>The evaluator shall perform the following tests:            Test 3: The evaluator shall ensure, for each communication channel with an authorized IT entity, the channel data is not sent in plaintext.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target.</p>
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0572) for Restricting FTP_ITC.1 to only IP address identifiers.</b></p> <p>DNS resolution is not mandatory to support FTP_ITC.1. There are no resolution mandates or requirements in RFC 6125.</p> <p>The use of the dnsName identifiers in X.509 certificates must be supported by TOEs that claim FCS_DTLSC_EXT or FCS_TLSC_EXT when associated with FTP_ITC.1. Inability to parse dnsName identifiers shall be considered a failure to meet these requirements.</p>
<b>Test Steps</b>	This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications.

<b>Expected Results</b>	Connection with the external audit server through IPsec VPN tunnel is protected and no data is sent in plaintext.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered in conjunction with FAU_STG_EXT.1 Test #1 for audit server and FCS_IPSEC_EXT.1.1 Test #1 for VPN communications. As the tests show, all communication with the external audit server is protected by IPsec.

**7.1.17 FTP\_ITC.1 TEST #4**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report.</p> <p>The evaluator shall perform the following tests:  Test 4: Objective: The objective of this test is to ensure that the TOE reacts appropriately to any connection outage or interruption of the route to the external IT entities.</p> <p>The evaluator shall, for each instance where the TOE acts as a client utilizing a secure communication mechanism with a distinct IT entity, physically interrupt the connection of that IT entity for the following durations:</p> <ul style="list-style-type: none"> <li>i) A duration that exceeds the TOE’s application layer timeout setting,</li> <li>ii) A duration shorter than the application layer timeout but of sufficient length to interrupt the network link layer.</li> </ul> <p>The evaluator shall ensure that, when the physical connectivity is restored, communications are appropriately protected and no TSF data is sent in plaintext.</p> <p>In the case where the TOE is able to detect when the cable is removed from the device, another physical network device (e.g. a core switch) shall be used to interrupt the connection between the TOE and the distinct IT entity. The interruption shall not be performed at the virtual node (e.g. virtual switch) and must be physical in nature.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target.</p>
<b>Notes</b>	<b>The NiT has issued a technical decision (TD0572) for Restricting FTP_ITC.1 to only IP address identifiers.</b>

	<p>DNS resolution is not mandatory to support FTP_ITC.1. There are no resolution mandates or requirements in RFC 6125.</p> <p>The use of the dnsName identifiers in X.509 certificates must be supported by TOEs that claim FCS_DTLSC_EXT or FCS_TLSC_EXT when associated with FTP_ITC.1. Inability to parse dnsName identifiers shall be considered a failure to meet these requirements.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the TOE to connect with the remote peer IT entity.</li> </ul> <p>Short Disconnect:</p> <ul style="list-style-type: none"> <li>• Initiate the connection between the TOE and the peer.</li> <li>• Physically disrupt the connection for a short time, then test the connection. No data will go through, when connectivity is restored, the connection remains encrypted.</li> <li>• Verify successful connection and connection restoration via audit logs.</li> <li>• Verify short-duration connection disruption via packet capture.</li> </ul> <p>Long Disconnect:</p> <ul style="list-style-type: none"> <li>• Initiate the connection between the TOE and the peer.</li> <li>• Physically disrupt the connection for a long time, then test the connection. No data will go through, when connectivity is restored, the connection remains encrypted.</li> <li>• Verify successful connection and connection restoration via audit logs.</li> <li>• Verify long-duration connection disruption via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When physical connectivity with a remote peer IT entity is interrupted and then restored, the data exchanged between both entities should never be in plaintext.</li> <li>• Evidence - Packet capture should show connection reset and encrypted application data.</li> <li>• TOE log should show logs for successful connection and restored connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE does not send plaintext traffic when disconnected from the remote peer IT entity, regardless of the duration (short or long) of the disconnection. This meets the testing requirements.</p>

## 7.2 AUTH

### 7.2.1 FCS\_HTTPS\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>This test is now performed as part of FIA_X509_EXT.1/Rev testing.</p> <p>Tests are performed in conjunction with the TLS evaluation activities.</p> <p>If the TOE is an HTTPS client or an HTTPS server utilizing X.509 client authentication, then the certificate validity shall be tested in accordance with testing performed for FIA_X509_EXT.1.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE acts as an HTTPS Server only for WebGUI access. The TOE acts as a TLS Server without client authentication for both HTTPS and TLSS.</p>

### 7.2.2 FIA\_AFL.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application):</p> <p>Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE (and, if the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall also use the operational guidance to configure the time period after which access is re-enabled). The evaluator shall test that once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful.</p>
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0570) for clarification about FIA_AFL.1.</b></p> <ol style="list-style-type: none"><li>1. FIA_AFL.1 is a mandatory SFRs that the TOE will need to meet.</li><li>2. FIA_AFL.1 requires at least one remote administrative interface support password authentication.</li><li>3. If SSH is the TOE's only remote administrative interface, it needs to support password authentication. If there is another administrative interface (e.g. a web GUI) that supports password authentication, SSH does not need to support password authentication and, by extension, FIA_AFL.1.</li></ol> <p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p>



	<ol style="list-style-type: none"> <li>1. FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the TOE to allow up to 3 successive unsuccessful authentication attempts and set the time period after which access can be re-enabled.</li> <li>• Attempt to establish remote administrator access to the TOE using invalid credentials for 3 successive times.</li> <li>• Verify that an audit log has been generated indicating login failure.</li> <li>• Attempt to establish remote administrator access to the TOE using valid credentials.</li> <li>• Verify that this attempt is no longer successful and observe the logs indicating the authentication attempts limit is met/exceeded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not allow access to the device if an account fails authentication after a configured number of attempts.</li> <li>• TOE should show logs indicating the authentication attempts limit is met/exceeded.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE successfully locks out a user once the configured authentication attempts limit is reached, and authentication attempts with valid credentials are no longer successful. This meets the testing requirements.

7.2.3 FIA\_AFL.1 TEST #2A

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application):</p> <p>Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows:</p> <p>If the <b>administrator action</b> selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall confirm by testing that following the operational guidance and performing each action specified in the ST to re-enable the remote administrator’s access results in successful access (when using valid credentials for that administrator).</p>

<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0570) for clarification about FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_AFL.1 is a mandatory SFRs that the TOE will need to meet.</li> <li>2. FIA_AFL.1 requires at least one remote administrative interface support password authentication.</li> <li>3. If SSH is the TOE's only remote administrative interface, it needs to support password authentication. If there is another administrative interface (e.g. a web GUI) that supports password authentication, SSH does not need to support password authentication and, by extension, FIA_AFL.1.</li> </ol> <p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>
<b>Pass/Fail with Explanation</b>	<p>N/A. The <b>administrator action</b> selection in FIA_AFL.1.2 is not included in the ST</p>

#### 7.2.4 FIA\_AFL.1 TEST #2B

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application):</p> <p>Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows:</p> <p>If the <b>time period</b> selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall wait for just less than the time period configured in Test 1 and show that an authorisation attempt using valid credentials does not result in successful access. The evaluator shall then wait until just after the time period configured in Test 1 and show that an authorisation attempt using valid credentials results in successful access.</p>
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0570) for clarification about FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_AFL.1 is a mandatory SFRs that the TOE will need to meet.</li> <li>2. FIA_AFL.1 requires at least one remote administrative interface support password authentication.</li> </ol>

	<p>3. If SSH is the TOE's only remote administrative interface, it needs to support password authentication. If there is another administrative interface (e.g. a web GUI) that supports password authentication, SSH does not need to support password authentication and, by extension, FIA_AFL.1.</p> <p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Verify the configured lockout period.</li> <li>• Attempt to establish remote administrator access to the TOE using invalid credentials for 3 successive times.</li> <li>• Verify that an audit log has been generated indicating login failure.</li> <li>• Attempt to establish remote administrators access to the TOE using valid credentials just less than the configured time period.</li> <li>• Verify that this attempt is no longer successful and observe the logs.</li> <li>• Just after the configured time period, attempt to establish remote administrators access to the TOE using valid credentials.</li> <li>• Verify that this attempt is successful via TOE logs.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should allow a locked-out user to log in again after lockout time expires.</li> <li>• TOE should show account locked out logs and successful authentication logs once locked out time is completed.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE successfully rejects login with valid credentials till the lockout period and allows a locked-out user to log in again after the lockout time expires. This meets the testing requirements.</p>

7.2.5 FIA\_PMG\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests.</p> <p>Test 1: The evaluator shall compose passwords that meet the requirements in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of</p>

	passwords, the evaluator shall ensure that all characters, and a minimum length listed in the requirement are supported and justify the subset of those characters chosen for testing.
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Set the minimum password requirements. <ul style="list-style-type: none"> <li>○ Minimum 15-character length</li> <li>○ Minimum 1 upper case</li> <li>○ Minimum 1 lower case</li> <li>○ Minimum 1 digit</li> <li>○ Minimum 1 special character</li> </ul> </li> <li>• Attempt to create 15 characters password with username: good &amp; password: AB1CD7E!a@bc1de</li> <li>• Attempt to create 15 characters password with username: good1 &amp; password: FG2HI8J#f\$gh2ij</li> <li>• Attempt to create 15 characters password with username: good2 &amp; password: KL3MN9O%k^lm3no</li> <li>• Attempt to create 15 characters password with username: good3 &amp; password: PQ4RSOT&amp;p*qr4st</li> <li>• Attempt to create 15 characters password with username: good4 &amp; password: UV5WX1Y(u)vw5xy</li> <li>• Attempt to create 15 characters password with username: good5 &amp; password: ZA6BC2D!z@ab6cd</li> <li>• Verify all the usernames with correct password requirements are created.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• User accounts with passwords that meet requirements will be created.</li> <li>• TOE logs should show the successful creation of users.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE successfully creates user accounts with strong passwords. All characters claimed in the ST are supported by the TOE, and the passwords meet the minimum length requirement specified. This meets the testing requirements.

7.2.6 FIA\_PMG\_EXT.1 TEST #2

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall perform the following tests.</p> <p>Test 2: The evaluator shall compose passwords that do not meet the requirements in some way. For each password, the evaluator shall verify that the TOE does not support the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that the TOE enforces the allowed characters and the minimum length listed in the requirement and justify the subset of those characters chosen for testing.</p>
<p><b>Notes</b></p>	<p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>1. FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>
<p><b>Test Steps</b></p>	<ul style="list-style-type: none"> <li>• Set the minimum password requirements. <ul style="list-style-type: none"> <li>○ Minimum 15-character length</li> <li>○ Minimum 1 upper case</li> <li>○ Minimum 1 lower case</li> <li>○ Minimum 1 digit</li> <li>○ Minimum 1 special character</li> </ul> </li> <li>• Create an user1 having a password with a combination of lowercase, uppercase letters, numbers, and special characters (ABcd12!@).</li> <li>• Verify that the user was created successfully.</li> <li>• Attempt to change the password for user1 to a bad password that is less than 15 characters (UV5WX1Y(u)vw) and verify that it fails.</li> <li>• Attempt to change the password for user1 to a bad password with no lowercase letters (FG2HI8J#F\$GH2IJ) and verify that it fails.</li> <li>• Attempt to change the password for user1 to a bad password with no uppercase letters (ab1cd7ela@bc1de) and verify that it fails.</li> <li>• Attempt to change the password for user1 to a bad password with no numbers (KLmMNra%k^lmsno) and verify that it fails.</li> <li>• Attempt to change the password for user1 to a bad password with no special characters (PQ4RS0T2prqr4st) and verify that it fails.</li> </ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should only accept valid password combinations and generate an error when attempting to change the password of users with incorrect password combinations.</li> <li>• Evidence - screenshot showing error while changing password.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects users with bad passwords that do not meet the requirements. This meets the testing requirement.

### 7.2.7 FIA\_UIA\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method:</p> <p>Test 1: The evaluator shall use the guidance documentation to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct I&amp;A information results in the ability to access the system, while providing incorrect information results in denial of access.</p>
<b>Test Steps</b>	<p><u>Remote login – Web</u></p> <ul style="list-style-type: none"> <li>• Attempt to login from a remote web connection with incorrect credentials.</li> <li>• Verify that an audit record was generated showing login failure.</li> <li>• Attempt to login from a remote web connection with the correct credentials.</li> <li>• Verify that an audit record was generated showing login success.</li> </ul> <p><u>Local Console Login</u></p> <ul style="list-style-type: none"> <li>• Attempt to login from a local connection with incorrect credentials.</li> <li>• Verify that an audit record was generated showing login failure.</li> <li>• Attempt to login from a local connection with the correct credentials.</li> <li>• Verify that an audit record was generated showing login success.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should allow the user with correct credentials and reject the user with incorrect credentials.</li> <li>• TOE should generate logs for the successful and unsuccessful login attempts.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE successfully authenticates users with correct credentials and login fails when incorrect credentials are used. This meets the testing requirements.

### 7.2.8 FIA\_UIA\_EXT.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method:</p> <p>Test 2: The evaluator shall configure the services allowed (if any) according to the guidance documentation, and then determine the services available to an external remote entity. The evaluator shall determine that the list of services available is limited to those specified in the requirement.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Attempt to connect to the TOE remotely and verify that the only option presented before login is the access banner and links to TOE’s public knowledge-base web pages.</li> <li>• Verify that no other action is available for the users before successful login.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• No services except displaying a banner and links to public web pages should be available to a remote administrator attempting to login to the TOE remotely.</li> <li>• Evidence – Snap showing banner and links to public web pages.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE displays the access banner and provides links to knowledge-based websites before login. This meets the testing requirements.</p>

### 7.2.9 FIA\_UIA\_EXT.1 TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method:</p> <p>Test 3: For local access, the evaluator shall determine what services are available to a local administrator prior to logging in, and make sure this list is consistent with the requirement.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Attempt to connect to the TOE with some TOE commands and verify that the system commands are not available.</li> <li>• Verify authentication logs reflect failure.</li> <li>• Attempt to connect to the TOE locally and verify the only option presented is banner and then username/password entry.</li> <li>• Verify authentication logs reflect success.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE does not expose any services other than the ones meant to be exposed i.e. username/password entry and banner.</li> </ul>

	<ul style="list-style-type: none"> <li>Evidence – Snap showing only the username/password entry and banner is present before login.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The only option presented is the username/password entry, and banner to the administrator before login and no other system services are available to a local administrator prior to logging in via the directly connected console. This meets the testing requirements.

7.2.10 FIA\_UIA\_EXT.1 TEST #4

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method:</p> <p>Test 4: For distributed TOEs where not all TOE components support the authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2, the evaluator shall test that the components authenticate Security Administrators as described in the TSS.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since the TOE is not distributed.

7.2.11 FIA\_UAU\_EXT.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	Evaluation Activities for this requirement are covered under those for FIA_UIA_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA_UIA_EXT.1.
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> <li>FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.</li> </ol>



<b>Pass/Fail with Explanation</b>	Pass. Test cases under FIA_UIA_EXT.1 cover the requirements.
-----------------------------------	--

#### 7.2.12 FIA\_UAU.7 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following test for each method of local login allowed:  Test 1: The evaluator shall locally authenticate to the TOE. While making this attempt, the evaluator shall verify that at most obscured feedback is provided while entering the authentication information.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Connect to the TOE via the local console with incorrect authentication credentials and verify that at most obscured feedback is provided.</li> <li>• Verify authentication logs reflect failure.</li> <li>• Connect to the TOE via the local console with the correct authentication credentials and verify that most obscured feedback is provided.</li> <li>• Verify authentication logs reflect success.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not provide anything other than obscured feedback at the directly connected login console.</li> <li>• Evidence: screenshot showing password is obscured.</li> <li>• TOE logs should show successful/unsuccessful login attempts.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE meets password obscurity standards. This meets the testing requirements.

#### 7.2.13 FMT\_MOF.1/MANUALUPDATE TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to perform the update using a legitimate update image without prior authentication as Security Administrator (either by authentication as a user with no administrator privileges or without user authentication at all – depending on the configuration of the TOE). The attempt to update the TOE shall fail.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create a lower-privileged user.</li> <li>• Login as a user without Security Administrator privileges.</li> <li>• Attempt to update the device. This will fail as the lower privileged user has Read-Only Mode access.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• Users without Security Administrative access will not be able to perform the update using a legitimate update image.</li> </ul>

	<ul style="list-style-type: none"> <li>Evidence - screenshot showing an error message.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE does not allow users without Security Administrator privileges to update using a legitimate image. This meets the testing requirements.

#### 7.2.14 FMT\_MOF.1/MANUALUPDATE TEST #2

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to perform the update with prior authentication as Security Administrator using a legitimate update image. This attempt should be successful. This test case should be covered by the tests for FPT_TUD_EXT.1 already.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered in conjunction with FPT_TUD_EXT.1 Test #1.

#### 7.2.15 FMT\_MOF.1/SERVICES TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) without prior authentication as Security Administrator (either by authenticating as a user with no administrator privileges, if possible, or without prior authentication at all). The attempt to enable/disable this service/these services should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to enable/disable this service/these services can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Create a lower-privileged user.</li> <li>Log into the TOE as the lower privileged user.</li> <li>Attempt to perform TOE services (on-demand self-tests) on the TOE and verify the command is rejected.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The attempt to perform on-demand self-tests without authenticating as a security administrator should fail.</li> <li>Evidence - screenshot showing an error message.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. Users without prior authentication/privilege as security administrators cannot modify services. This meets the testing requirements.

#### 7.2.16 FMT\_MOF.1/SERVICES TEST #2

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) with prior authentication as Security Administrator. The attempt to enable/disable this service/these services should be successful.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Log into the TOE as the admin user.</li> <li>Attempt to perform TOE services (on demand self-tests) on the TOE and verify the command is successful.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The attempt to perform on demand self-tests with prior authentication as a security administrator should be successful.</li> <li>Evidence- Snapshot showing TOE self-tests</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. A security administrator can perform on demand self-tests. This meets the testing requirements.

#### 7.2.17 FMT\_MTD.1/COREDATA TEST #1

Item	Data
<b>Test Assurance Activity</b>	No separate testing for FMT_MTD.1/CoreData is required unless one of the management functions has not already been exercised under any other SFR.
<b>Pass/Fail with Explanation</b>	Pass. No separate testing for FMT_MTD.1/CoreData is required as all management functions have already been already exercised under claimed SFRs and there are no remaining functions to be tested.

#### 7.2.18 FMT\_MTD.1/CRYPTOKEYS TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to perform at least one of the related actions (modify, delete, generate/import) without prior authentication as Security Administrator (either by authentication as a non-administrative user, if supported, or without authentication at all). Attempts to perform related actions without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to manage cryptographic keys can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Connect to the TOE as an unprivileged user.</li> </ul>

	<ul style="list-style-type: none"> <li>Attempt to modify cryptographic keys i.e. delete/modify/generate/import CSR. This will fail as all such options are unavailable.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The attempt to manage the crypto keys without prior authenticating as a security administrator should fail.</li> <li>Evidence - screenshot showing options are disabled.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. Users without prior authentication/privilege as security administrators cannot modify cryptographic keys. This meets the testing requirements.

#### 7.2.19 FMT\_MTD.1/CRYPTOKEYS TEST #2

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall try to perform at least one of the related actions with prior authentication as Security Administrator. This attempt should be successful.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Connect to the TOE as privileged user.</li> <li>Attempt to generate cryptographic keys on the TOE and verify it is successful.</li> <li>Verify that an audit log was generated during the event.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The attempt to manage the crypto keys with prior authenticating as a security administrator should be successful.</li> <li>Evidence – TOE logs showing CSR generation.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. User with authentication as Security Administrator is able to generate cryptographic keys. This meets the testing requirements.

#### 7.2.20 FMT\_SMF.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator tests management functions as part of testing the SFRs identified in section 2.4.4. No separate testing for FMT_SMF.1 is required unless one of the management functions in FMT_SMF.1.1 has not already been exercised under any other SFR.
<b>Notes</b>	<p><b>The NiT has issued a technical decision (TD0571) for Guidance on how to handle FIA_AFL.1.</b></p> <ol style="list-style-type: none"> <li>FIA_UAU_EXT.2.1 applies solely to how the administrator logs in at the local console. Passwords were made selection based to allow TOEs that have difficulty determining whether an incoming connection is local or remote to provide a mechanism to prevent administrative lockout. As FIA_AFL.1 is a mandatory SFR, it is expected that the TOE provides at least one remote password-based authentication mechanism using credentials managed by the TOE.</li> </ol>

	2. FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 are all mandatory SFRs that the TOE will need to meet.
<b>Pass/Fail with Explanation</b>	Pass. Throughout the various security functionality testing of the TOE, FMT_SMF.1 Specification of Management Functions requirements have been met. This meets the testing requirements.

#### 7.2.21 FMT\_SMR.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this cPP be tested; for instance, if the TOE can be administered through a local hardware interface; SSH; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered in conjunction with the following tests: FIA_UIA_EXT.1 Test #2, FIA_UIA_EXT.1 Test #3, FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3 Test #1, FTA_SSL.4 Test #1 and FTA_TAB.1 Test #1.

#### 7.2.22 FTA\_SSL.3 TEST #1

Item	Data
<b>Test Assurance Activity</b>	For each method of remote administration, the evaluator shall perform the following test:  Test 1: The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a remote interactive session with the TOE. The evaluator then observes that the session is terminated after the configured time period.
<b>Test Steps</b>	<b>Timeout for 2 minutes:</b> <ul style="list-style-type: none"> <li>• Configure a 2-minute inactivity timeout period for remote administrative sessions.</li> <li>• Attempt to log in to the TOE via the web GUI.</li> <li>• Verify login time.</li> <li>• Verify the login time through the logs.</li> <li>• Let the session sit idle for 2 minutes and verify that the session was terminated.</li> </ul>

	<ul style="list-style-type: none"> <li>Verify through the logs that the user was logged out due to inactivity time expiring.</li> </ul> <p><b>Timeout for 5 minutes:</b></p> <ul style="list-style-type: none"> <li>Configure a 5-minute inactivity timeout period for remote administrative sessions.</li> <li>Attempt to log in to the TOE via the web GUI.</li> <li>Verify login time.</li> <li>Verify the login time through the logs.</li> <li>Let the session sit idle for 5 minutes and verify that the session was terminated.</li> <li>Verify through the logs that the user was logged out due to inactivity time expiring.</li> </ul> <p><b>Timeout for 15 minutes:</b></p> <ul style="list-style-type: none"> <li>Configure a 15-minute inactivity timeout period for remote administrative sessions.</li> <li>Attempt to log in to the TOE via the web GUI.</li> <li>Verify login time.</li> <li>Verify the login time through the logs.</li> <li>Let the session sit idle for 15 minutes and verify that the session was terminated.</li> <li>Verify through the logs that the user was logged out due to inactivity time expiring.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The remote administrative session should be terminated after the configured inactivity time period.</li> <li>TOE should generate logs for session timeout.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE disconnects users from web GUI after meeting the inactivity time limit. This meets the testing requirements.

7.2.23 FTA\_SSL.4 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>For each method of remote administration, the evaluator shall perform the following tests:</p> <p>Test 1: The evaluator initiates an interactive local session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Log into the TOE through a local administrative session.</li> <li>Verify the logs reflect login.</li> </ul>

	<ul style="list-style-type: none"> <li>Using the instructions provided in the guidance documentation, attempt to exit the session.</li> <li>Verify the logs reflect the logout.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The local session should be terminated once the exit command is entered.</li> <li>TOE should generate logs for logout.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE allows the user to terminate the directly connected administrative session. This meets the testing requirements.

#### 7.2.24 FTA\_SSL.4 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>For each method of remote administration, the evaluator shall perform the following tests:</p> <p>Test 2: The evaluator initiates an interactive remote session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Log into the TOE through a remote administrative session.</li> <li>Verify the logs reflect login.</li> <li>Log out of the device.</li> <li>Verify the logs reflect the logout.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The remote session should be terminated once the logout action is performed.</li> <li>TOE should generate logs for logout.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE allows the user to terminate the remote administrative session. This meets the testing requirements.

#### 7.2.25 FTA\_SSL\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following test:</p> <p>Test 1: The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a local interactive session with the TOE. The evaluator then observes that the session is either locked or terminated after the configured time period. If locking was selected from the component, the evaluator then ensures that reauthentication is needed when trying to unlock the session.</p>
<b>Test Steps</b>	<b>Timeout for 2 minutes:</b>

	<ul style="list-style-type: none"> <li>• Configure an inactivity time out period of 2 minutes on local administrative sessions.</li> <li>• Attempt to login to the TOE locally.</li> <li>• Verify login time using log.</li> <li>• Let the session sit idle for 2 minutes and verify that the session was terminated.</li> <li>• Verify through the logs that the user was logged out due to the session timed out.</li> </ul> <p><b>Timeout for 5 minutes:</b></p> <ul style="list-style-type: none"> <li>• Configure an inactivity time out period of 5 minutes on local administrative sessions.</li> <li>• Attempt to login to the TOE locally.</li> <li>• Verify login time using log.</li> <li>• Let the console sit idle for 5 minutes and verify that the session was terminated.</li> <li>• Verify through the logs that the user was logged out due to the session timed out.</li> </ul> <p><b>Timeout for 15 minutes:</b></p> <ul style="list-style-type: none"> <li>• Configure an inactivity time out period of 15 minutes on local administrative sessions.</li> <li>• Attempt to login to the TOE locally.</li> <li>• Verify login time using log.</li> <li>• Let the console sit idle for 15 minutes and verify that the session was terminated.</li> <li>• Verify through the logs that the user was logged out due to the session timed out.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The local interactive session should be terminated after the configured time period.</li> <li>• TOE logs should show session termination.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE ends the user session on local console after the inactivity time limit is reached. This meets the testing requirements.

7.2.26 FTA\_TAB.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall also perform the following test:</p> <p>Test 1: The evaluator follows the guidance documentation to configure a notice and consent warning message. The evaluator shall then, for each method of access specified</p>



	in the TSS, establish a session with the TOE. The evaluator shall verify that the notice and consent warning message is displayed in each instance.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure a notice and consent warning message on the TOE.</li> <li>• Verify that the audit records reflect the configuration steps.</li> <li>• Log into the TOE via the local console and verify that the warning message is displayed.</li> <li>• Log into the TOE via Web GUI and verify that the warning message is displayed.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The notice and consent warning message should be displayed in each instance of login (local and remote).</li> <li>• Evidence - screenshot showing banners.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. An access banner can be set for all the methods that can be used to access the device. This meets the testing requirements.

#### 7.2.27 FTP\_TRP.1/ADMIN TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: The evaluators shall ensure that communications using each specified (in the guidance documentation) remote administration method is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Attempt to establish an HTTPS session from a remote administrator (Web GUI).</li> <li>• Verify audit logs that the user is successfully logged in to the TOE.</li> <li>• Capture the traffic between the devices and verify that the connection was successful, and traffic is not sent in plaintext.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should encrypt the traffic successfully.</li> <li>• Evidence – Packet capture showing successful connection.</li> <li>• TOE logs should show a successful login.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. Users can access the TOE via a TLS connection and traffic is not sent in plaintext. This meets the testing requirements.

#### 7.2.28 FTP\_TRP.1/ADMIN TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall ensure, for each communication channel, the channel data is not sent in plaintext.</p> <p>For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. This test is covered in conjunction with FTP_TRP.1/Admin Test #1.</p>

## 7.3.1 FCS\_CKM.1 RSA

Item	Data
<b>Test Assurance Activity</b>	<p>Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported).</p> <p><b>Key Generation for FIPS PUB 186-4 RSA Schemes</b></p> <p>The evaluator shall verify the implementation of RSA Key Generation by the TOE using the Key Generation test. This test verifies the ability of the TSF to correctly produce values for the key components including the public verification exponent <math>e</math>, the private prime factors <math>p</math> and <math>q</math>, the public modulus <math>n</math> and the calculation of the private signature exponent <math>d</math>.</p> <p>Key Pair generation specifies 5 ways (or methods) to generate the primes <math>p</math> and <math>q</math>. These include:</p> <ul style="list-style-type: none"> <li>a) Random Primes: <ul style="list-style-type: none"> <li>• Provable primes</li> <li>• Probable primes</li> </ul> </li> <li>b) Primes with Conditions: <ul style="list-style-type: none"> <li>• Primes <math>p_1, p_2, q_1, q_2, p</math> and <math>q</math> shall all be provable primes</li> <li>• Primes <math>p_1, p_2, q_1</math>, and <math>q_2</math> shall be provable primes and <math>p</math> and <math>q</math> shall be probable primes</li> <li>• Primes <math>p_1, p_2, q_1, q_2, p</math> and <math>q</math> shall all be probable primes</li> </ul> </li> </ul> <p>To test the key generation method for the Random Provable primes method and for all the Primes with Conditions methods, the evaluator must seed the TSF key generation routine with sufficient data to deterministically generate the RSA key pair. This includes the random seed(s), the public exponent of the RSA key, and the desired key length. For each key length supported, the evaluator shall have the TSF generate 25 key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: RSA KeyGen  Key size / Modulus: 2048, 3072 4096  CAVP #: A5110, A2583, A4982</p>

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

7.3.2 FCS\_CKM.1 ECC

Item	Data
<b>Test Assurance Activity</b>	<p>Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported).</p> <p><b>Key Generation for Elliptic Curve Cryptography (ECC)</b>  <i>FIPS 186-4 ECC Key Generation Test</i>            For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator shall submit the generated key pairs to the public key verification (PKV) function of a known good implementation.</p> <p><i>FIPS 186-4 Public Key Verification (PKV) Test</i>            For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall generate 10 private/public key pairs using the key generation function of a known good implementation and modify five of the public key values so that they are incorrect, leaving five values unchanged (i.e., correct). The evaluator shall obtain in response a set of 10 PASS/FAIL values.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: ECDSA KeyGen and ECDSA KeyVer            Curves: P-256, P-384, P-521            CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.3 FCS\_CKM.1 FFC – FIPS PUB 186-4

Item	Data
<b>Test Assurance Activity</b>	Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products.

Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported).

### **Key Generation for Finite-Field Cryptography (FFC)**

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

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

- Primes  $q$  and  $p$  shall both be provable primes
- Primes  $q$  and field prime  $p$  shall both be probable primes

and two ways to generate the cryptographic group generator  $g$ :

- Generator  $g$  constructed through a verifiable process
- Generator  $g$  constructed through an unverifiable process.

The Key generation specifies 2 ways to generate the private key  $x$ :

- $\text{len}(q)$  bit output of RBG where  $1 \leq x \leq q-1$
- $\text{len}(q) + 64$  bit output of RBG, followed by a mod  $q-1$  operation and a  $+1$  operation, where  $1 \leq x \leq q-1$ .

The security strength of the RBG must be at least that of the security offered by the FFC parameter set.

To test the cryptographic and field prime generation method for the provable primes method and/or the group generator  $g$  for a verifiable process, the evaluator must seed the TSF parameter generation routine with sufficient data to deterministically generate the parameter set.

For each key length supported, the evaluator shall have the TSF generate 25 parameter sets and key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation. Verification must also confirm

- $g \neq 0,1$
- $q$  divides  $p-1$
- $g^q \bmod p = 1$
- $g^x \bmod p = y$

for each FFC parameter set and key pair.

<b>Pass/Fail with Explanation</b>	N/A. Algorithm DSA KeyGen is not claimed in the ST.
-----------------------------------	---

#### 7.3.4 FCS\_CKM.1 FFC – “SAFE-PRIME” GROUPS [TD0580]

Item	Data
<b>Test Assurance Activity</b>	<p>Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported).</p> <p><b>FFC Schemes using “safe-prime” groups</b> Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1.</p> <p><b>TD0580 has been applied.</b></p>
<b>Pass/Fail with Explanation</b>	<p>Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1.</p> <p>Additionally, Algorithm: Safe Prime Key Generation and Safe Primes Key Verification Safe prime Groups: modp-2048 CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

#### 7.3.5 FCS\_CKM.2 RSA

Item	Data
<b>Test Assurance Activity</b>	<p><b>RSA-based key establishment</b></p> <p>The evaluator shall verify the correctness of the TSF’s implementation of RSAES-PKCS1-v1_5 by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses RSAES-PKCS1-v1_5.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. This testing was performed in conjunction with FTP_TRP.1/Admin Test #1 and FTP_ITC.1 Test #1 to demonstrate correct operation.</p>

Item	Data
<p><b>Test Assurance Activity</b></p>	<p><b>Key Establishment Schemes</b></p> <p>The evaluator shall verify the implementation of the key establishment schemes of the supported by the TOE using the applicable tests below.</p> <p><b>SP800-56A Key Establishment Schemes</b></p> <p>The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag.</p> <p><i>Function Test</i></p> <p>The Function test verifies the ability of the TOE to implement the key agreement schemes correctly. To conduct this test the evaluator shall generate or obtain test vectors from a known good implementation of the TOE supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of one set of domain parameter values (FFC) or the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested.</p> <p>The evaluator shall obtain the DKM, the corresponding TOE's public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and TOE id fields.</p> <p>If the TOE does not use a KDF defined in SP 800-56A, the evaluator shall obtain only the public keys and the hashed value of the shared secret.</p> <p>The evaluator shall verify the correctness of the TSF's implementation of a given scheme by using a known good implementation to calculate the shared secret value, derive the keying material DKM, and compare hashes or MAC tags generated from these values.</p>

	<p>If key confirmation is supported, the TSF shall perform the above for each implemented approved MAC algorithm.</p> <p><i>Validity Test</i></p> <p>The Validity test verifies the ability of the TOE to recognize another party’s valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator’s public keys, the TOE’s public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields.</p> <p>The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties’ static public keys, both parties’ ephemeral public keys and the TOE’s static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass).</p> <p>The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE’s results with the results using a known good implementation verifying that the TOE detects these errors.</p>
<p><b>Pass/Fail with Explanation</b></p>	<p>Algorithm: KAS-ECC-SSC Sp800-56Ar3  Curves: P-256, P-384, P-521  CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.7 FCS\_CKM.2 SP800-56A - FFC

Item	Data
<p><b>Test Assurance Activity</b></p>	<p><b>Key Establishment Schemes</b></p> <p>The evaluator shall verify the implementation of the key establishment schemes of the supported by the TOE using the applicable tests below.</p>



### **SP800-56A Key Establishment Schemes**

The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag.

#### *Function Test*

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

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

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

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

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

#### *Validity Test*

The Validity test verifies the ability of the TOE to recognize another party's valid and invalid key agreement results with or without key confirmation. To conduct this test, the

	<p>evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator’s public keys, the TOE’s public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields.</p> <p>The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties’ static public keys, both parties’ ephemeral public keys and the TOE’s static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass).</p> <p>The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE’s results with the results using a known good implementation verifying that the TOE detects these errors.</p>
<p><b>Pass/Fail with Explanation</b></p>	<p>Algorithm: KAS-FFC-SSC Sp800-56Ar3            Generation Method: DH-14 (MODP-2048)            CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.8 FCS\_CKM.2 FCC SAFE-PRIME

Item	Data
<p><b>Test Assurance Activity</b></p>	<p><b>FFC Schemes using “safe-prime” groups</b></p> <p>The evaluator shall verify the correctness of the TSF’s implementation of safe-prime groups by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses safe-prime groups. This test must be performed for each safe-prime group that each protocol uses.</p>
<p><b>Pass/Fail with Explanation</b></p>	<p>This test has been successfully tested in FTP_TRP.1/Admin, FTP_ITC.1 that uses safe-prime groups. The evaluator tested each protocol and verified the successful connection.</p>

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

### 7.3.9 FCS\_CKM.4

Item	Data
<b>Test Assurance Activity</b>	There are no test assurance activities.
<b>Notes</b>	<p><b>NIT Technical Decision (TD0639) for Clarification for NTP MAC Keys.</b></p> <p>The SFRs FAU_GEN.1, FCS_CKM.4 and FPT_SKP_EXT.1 shall be applied to all cryptographic keys that are related to secure communication (i.e. related to FTP_TRP.1, FTP_ITC.1, FPT_ITT.1). The NTP requirements have been introduced in NDcPP V2.1 as a rather 'standalone' set of requirements with 'no audit requirements' specified in the ECD section for FCS_NTP_EXT.1 and no dependencies on FCS_CKM - in contrast to the corresponding sections for secure communication protocols like TLS. As NTP keys are not intended to be used for encryption of sensitive information, the level of protection is different compared to other pre-shared keys. It has therefore not been intended that NTP keys are treated as other pre-shared keys in the context of NDcPP.</p>
<b>Pass/Fail with Explanation</b>	<p>N/A. There are no test assurance activities for this SFR.</p> <p>Based on these findings, this assurance activity is considered satisfied.</p>

### 7.3.10 FCS\_COP.1/DATAENCRYPTION AES-CBC

Item	Data
<b>Test Assurance Activity</b>	<p><b>AES-CBC Known Answer Tests</b></p> <p>There are four Known Answer Tests (KATs), described below. In all KATs, the plaintext, ciphertext, and IV values shall be 128-bit blocks. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.</p> <p><b>KAT-1.</b> To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 10 plaintext values and obtain the ciphertext value that results from AES-CBC encryption of the given plaintext using a key value of all zeros and an IV of all zeros. Five plaintext</p>

values shall be encrypted with a 128-bit all-zeros key, and the other five shall be encrypted with a 256-bit all-zeros key.

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using 10 ciphertext values as input and AES-CBC decryption.

**KAT-2.** To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 10 key values and obtain the ciphertext value that results from AES-CBC encryption of an all-zeros plaintext using the given key value and an IV of all zeros. Five of the keys shall be 128-bit keys, and the other five shall be 256-bit keys.

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using an all-zero ciphertext value as input and AES-CBC decryption.

**KAT-3.** To test the encrypt functionality of AES-CBC, the evaluator shall supply the two sets of key values described below and obtain the ciphertext value that results from AES encryption of an all-zeros plaintext using the given key value and an IV of all zeros. The first set of keys shall have 128 128-bit keys, and the second set shall have 256 256-bit keys. Key  $i$  in each set shall have the leftmost  $i$  bits be ones and the rightmost  $N-i$  bits be zeros, for  $i$  in  $[1,N]$ .

To test the decrypt functionality of AES-CBC, the evaluator shall supply the two sets of keys and ciphertext value pairs described below and obtain the plaintext value that results from AES-CBC decryption of the given ciphertext using the given key and an IV of all zeros. The first set of key/ciphertext pairs shall have 128 128-bit key/ciphertext pairs, and the second set of key/ciphertext pairs shall have 256 256-bit key/ciphertext pairs. Key  $i$  in each set shall have the leftmost  $i$  bits be ones and the rightmost  $N-i$  bits be zeros, for  $i$  in  $[1,N]$ . The ciphertext value in each pair shall be the value that results in an all-zeros plaintext when decrypted with its corresponding key.

**KAT-4.** To test the encrypt functionality of AES-CBC, the evaluator shall supply the set of 128 plaintext values described below and obtain the two ciphertext values that result from AES-CBC encryption of the given plaintext using a 128-bit key value of all zeros with an IV of all zeros and using a 256-bit key value of all zeros with an IV of all zeros, respectively. Plaintext value  $i$  in each set shall have the leftmost  $i$  bits be ones and the rightmost  $128-i$  bits be zeros, for  $i$  in  $[1,128]$ .

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using ciphertext values of the same form as the plaintext in the encrypt test as input and AES-CBC decryption.

### AES-CBC Multi-Block Message Test

The evaluator shall test the encrypt functionality by encrypting an i-block message where  $1 < i \leq 10$ . The evaluator shall choose a key, an IV and plaintext message of length i blocks and encrypt the message, using the mode to be tested, with the chosen key and IV. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key and IV using a known good implementation.

The evaluator shall also test the decrypt functionality for each mode by decrypting an i-block message where  $1 < i \leq 10$ . The evaluator shall choose a key, an IV and a ciphertext message of length i blocks and decrypt the message, using the mode to be tested, with the chosen key and IV. The plaintext shall be compared to the result of decrypting the same ciphertext message with the same key and IV using a known good implementation.

### AES-CBC Monte Carlo Tests

The evaluator shall test the encrypt functionality using a set of 200 plaintext, IV, and key 3-tuples. 100 of these shall use 128 bit keys, and 100 shall use 256 bit keys. The plaintext and IV values shall be 128-bit blocks. For each 3-tuple, 1000 iterations shall be run as follows:

```
# Input: PT, IV, Key
for i = 1 to 1000:
    if i == 1:
        CT[1] = AES-CBC-Encrypt(Key, IV, PT)
        PT = IV
    else:
        CT[i] = AES-CBC-Encrypt(Key, PT)
        PT = CT[i-1]
```

The ciphertext computed in the 1000th iteration (i.e., CT[1000]) is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation.

The evaluator shall test the decrypt functionality using the same test as for encrypt, exchanging CT and PT and replacing AES-CBC-Encrypt with AESCBC-Decrypt.

#### Pass/Fail with Explanation

Algorithm: AES CBC  
Key size: 128, 192, 256  
CAVP #: A5110, A2583, A4982

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

7.3.11 FCS\_COP.1/DATAENCRYPTION AES-GCM

Item	Data
<b>Test Assurance Activity</b>	<p><b>AES-GCM Test</b></p> <p>The evaluator shall test the authenticated encrypt functionality of AES-GCM for each combination of the following input parameter lengths:</p> <p><b>128 bit and 256 bit keys</b></p> <ul style="list-style-type: none"> <li>a) <b>Two plaintext lengths.</b> One of the plaintext lengths shall be a nonzero integer multiple of 128 bits, if supported. The other plaintext length shall not be an integer multiple of 128 bits, if supported.</li> <li>a) <b>Three AAD lengths.</b> One AAD length shall be 0, if supported. One AAD length shall be a non-zero integer multiple of 128 bits, if supported. One AAD length shall not be an integer multiple of 128 bits, if supported.</li> <li>b) <b>Two IV lengths.</b> If 96 bit IV is supported, 96 bits shall be one of the two IV lengths tested.</li> </ul> <p>The evaluator shall test the encrypt functionality using a set of 10 key, plaintext, AAD, and IV tuples for each combination of parameter lengths above and obtain the ciphertext value and tag that results from AES-GCM authenticated encrypt. Each supported tag length shall be tested at least once per set of 10. The IV value may be supplied by the evaluator or the implementation being tested, as long as it is known.</p> <p>The evaluator shall test the decrypt functionality using a set of 10 key, ciphertext, tag, AAD, and IV 5-tuples for each combination of parameter lengths above and obtain a Pass/Fail result on authentication and the decrypted plaintext if Pass. The set shall include five tuples that Pass and five that Fail.</p> <p>The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: AES GCM</p> <p>Key size: 128, 256</p> <p>CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.12 FCS\_COP.1/DATAENCRYPTION AES-CTR

Item	Data
<p><b>Test Assurance Activity</b></p>	<p><b>AES-CTR Known Answer Tests</b></p> <p>The Counter (CTR) mode is a confidentiality mode that features the application of the forward cipher to a set of input blocks, called counters, to produce a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. Since the Counter Mode does not specify the counter that is used, it is not possible to implement an automated test for this mode. The generation and management of the counter is tested through FCS_SSH*_EXT.1.4. If CBC and/or GCM are selected in FCS_COP.1/DataEncryption, the test activities for those modes sufficiently demonstrate the correctness of the AES algorithm. If CTR is the only selection in FCS_COP.1/DataEncryption, the AES-CBC Known Answer Test, AESGCM Known Answer Test, or the following test shall be performed (all of these tests demonstrate the correctness of the AES algorithm):</p> <p>There are four Known Answer Tests (KATs) described below to test a basic AES encryption operation (AES-ECB mode). For all KATs, the plaintext, IV, and ciphertext values shall be 128-bit blocks. The results from each test may either be obtained by the validator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.</p> <p>KAT-1 To test the encrypt functionality, the evaluator shall supply a set of 5 plaintext values for each selected keysize and obtain the ciphertext value that results from encryption of the given plaintext using a key value of all zeros.</p> <p>KAT-2 To test the encrypt functionality, the evaluator shall supply a set of 5 key values for each selected keysize and obtain the ciphertext value that results from encryption of an all zeros plaintext using the given key value.</p> <p>KAT-3 To test the encrypt functionality, the evaluator shall supply a set of key values for each selected keysize as described below and obtain the ciphertext values that result from AES encryption of an all zeros plaintext using the given key values. A set of 128 128-bit keys, a set of 192 192-bit keys, and/or a set of 256 256-bit keys. Key <i>i</i> in each set shall have the leftmost <i>i</i> bits be ones and the rightmost <i>N-i</i> bits be zeros, for <i>i</i> in [1, <i>N</i>].</p>

	<p>KAT-4 To test the encrypt functionality, the evaluator shall supply the set of 128 plaintext values described below and obtain the ciphertext values that result from encryption of the given plaintext using each selected keysize with a key value of all zeros (e.g. 256 ciphertext values will be generated if 128 bits and 256 bits are selected and 384 ciphertext values will be generated if all key sizes are selected). Plaintext value <math>i</math> in each set shall have the leftmost bits be ones and the rightmost <math>128-i</math> bits be zeros, for <math>i</math> in <math>[1, 128]</math>.</p> <p><b>AES-CTR Multi-Block Message Test</b></p> <p>The evaluator shall test the encrypt functionality by encrypting an <math>i</math>-block message where <math>1 \leq i \leq 10</math> (test shall be performed using AES-ECB mode). For each <math>i</math> the evaluator shall choose a key and plaintext message of length <math>i</math> blocks and encrypt the message, using the mode to be tested, with the chosen key. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key using a known good implementation. The evaluator shall perform this test using each selected keysize.</p> <p><b>AES-CTR Monte-Carlo Test</b></p> <p>The evaluator shall test the encrypt functionality using 100 plaintext/key pairs. The plaintext values shall be 128-bit blocks. For each pair, 1000 iterations shall be run as follows:</p> <pre># Input: PT, Key for i = 1 to 1000:     CT[i] = AES-ECB-Encrypt(Key, PT) PT = CT[i]</pre> <p>The ciphertext computed in the 1000th iteration is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation. The evaluator shall perform this test using each selected keysize.</p> <p>There is no need to test the decryption engine.</p>
<b>Pass/Fail with Explanation</b>	N/A. Algorithm AES CTR is not claimed in the ST.

7.3.13 FCS\_COP.1/SIGGEN ECDSA

Item	Data
<b>Test Assurance Activity</b>	<b>ECDSA Algorithm Tests</b> <b>ECDSA FIPS 186-4 Signature Generation Test</b>



	<p>For each supported NIST curve (i.e., P-256, P-384 and P-521) and SHA function pair, the evaluator shall generate 10 1024-bit long messages and obtain for each message a public key and the resulting signature values R and S. To determine correctness, the evaluator shall use the signature verification function of a known good implementation.</p> <p><b>ECDSA FIPS 186-4 Signature Verification Test</b></p> <p>For each supported NIST curve (i.e., P-256, P-384 and P-521) and SHA function pair, the evaluator shall generate a set of 10 1024-bit message, public key and signature tuples and modify one of the values (message, public key or signature) in five of the 10 tuples. The evaluator shall obtain in response a set of 10 PASS/FAIL values.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: ECDSA SigGen, ECDSA SigVer</p> <p>Curves: P-256, P-384, P-521</p> <p>CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.14 FCS\_COP.1/SIGGEN RSA

Item	Data
<b>Test Assurance Activity</b>	<p><b>RSA Signature Algorithm Tests</b></p> <p><b>Signature Generation Test</b></p> <p>The evaluator generates or obtains 10 messages for each modulus size/SHA combination supported by the TOE. The TOE generates and returns the corresponding signatures.</p> <p>The evaluator shall verify the correctness of the TOE’s signature using a trusted reference implementation of the signature verification algorithm and the associated public keys to verify the signatures.</p> <p><b>Signature Verification Test</b></p> <p>For each modulus size/hash algorithm selected, the evaluator generates a modulus and three associated key pairs, (d, e). Each private key d is used to sign six pseudorandom messages each of 1024 bits using a trusted reference implementation of the signature generation algorithm. Some of the public keys, e, messages, or signatures are altered so that signature verification should fail. For both the set of original messages and the set of altered messages: the modulus, hash algorithm, public key e values, messages, and signatures are forwarded to the TOE, which then attempts to verify the signatures and returns the verification results.</p>

	The evaluator verifies that the TOE confirms correct signatures on the original messages and detects the errors introduced in the altered messages.
<b>Pass/Fail with Explanation</b>	<p>Algorithm: RSA SigGen, RSA SigVer  Key size / Modulus: 2048, 3072, 4096  CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.15 FCS\_COP.1/HASH

Item	Data
<b>Test Assurance Activity</b>	<p>The TSF hashing functions can be implemented in one of two modes. The first mode is the byte-oriented mode. In this mode the TSF only hashes messages that are an integral number of bytes in length; i.e., the length (in bits) of the message to be hashed is divisible by 8. The second mode is the bit-oriented mode. In this mode the TSF hashes messages of arbitrary length. As there are different tests for each mode, an indication is given in the following sections for the bit-oriented vs. the byte-oriented testmacs.</p> <p>The evaluator shall perform all of the following tests for each hash algorithm implemented by the TSF and used to satisfy the requirements of this PP.</p> <p><b>Short Messages Test - Bit-oriented Mode</b>  The evaluators devise an input set consisting of m+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m bits. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.</p> <p><b>Short Messages Test - Byte-oriented Mode</b>  The evaluators devise an input set consisting of m/8+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m/8 bytes, with each message being an integral number of bytes. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.</p> <p><b>Selected Long Messages Test - Bit-oriented Mode</b>  The evaluators devise an input set consisting of m messages, where m is the block length of the hash algorithm (e.g. 512 bits for SHA-256). The length of the ith message is m +</p>

	<p>99*i, where <math>1 \leq i \leq m</math>. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.</p> <p><b>Selected Long Messages Test - Byte-oriented Mode</b></p> <p>The evaluators devise an input set consisting of <math>m/8</math> messages, where <math>m</math> is the block length of the hash algorithm (e.g. 512 bits for SHA-256). The length of the <math>i</math>th message is <math>m + 8*99*i</math>, where <math>1 \leq i \leq m/8</math>. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.</p> <p><b>Pseudorandomly Generated Messages Test</b></p> <p>This test is for byte-oriented implementations only. The evaluators randomly generate a seed that is <math>n</math> bits long, where <math>n</math> is the length of the message digest produced by the hash function to be tested. The evaluators then formulate a set of 100 messages and associated digests by following the algorithm provided in Figure 1 of [SHAVS]. The evaluators then ensure that the correct result is produced when the messages are provided to the TSF.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: SHA-1, SHA2-256, SHA2-384, SHA2-512 CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.16 FCS\_COP.1/KEYEDHASH

Item	Data
<b>Test Assurance Activity</b>	For each of the supported parameter sets, the evaluator shall compose 15 sets of test data. Each set shall consist of a key and message data. The evaluator shall have the TSF generate HMAC tags for these sets of test data. The resulting MAC tags shall be compared to the result of generating HMAC tags with the same key and message data using a known good implementation.
<b>Pass/Fail with Explanation</b>	<p>Algorithm: HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512 CAVP #: A5110, A2583, A4982</p> <p>Pass. Based on these findings, this assurance activity is considered satisfied.</p>

7.3.17 FCS\_RBG\_EXT.1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform 15 trials for the RNG implementation. If the RNG is configurable, the evaluator shall perform 15 trials for each configuration.</p> <p>If the RNG has prediction resistance enabled, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) generate a second block of random bits (4) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The next two are additional input and entropy input for the first call to generate. The final two are additional input and entropy input for the second call to generate. These values are randomly generated. “generate one block of random bits” means to generate random bits with number of returned bits equal to the Output Block Length (as defined in NIST SP800-90A).</p> <p>If the RNG does not have prediction resistance, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) reseed, (4) generate a second block of random bits (5) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The fifth value is additional input to the first call to generate. The sixth and seventh are additional input and entropy input to the call to reseed. The final value is additional input to the second generate call.</p> <p>The following paragraphs contain more information on some of the input values to be generated/selected by the evaluator.</p> <p><b>Entropy input:</b> the length of the entropy input value must equal the seed length.</p> <p><b>Nonce:</b> If a nonce is supported (CTR_DRBG with no Derivation Function does not use a nonce), the nonce bit length is one-half the seed length.</p> <p><b>Personalization string:</b> The length of the personalization string must be &lt;= seed length. If the implementation only supports one personalization string length, then the same length can be used for both values. If more than one string length is support, the evaluator shall use personalization strings of two different lengths. If the implementation does not use a personalization string, no value needs to be supplied.</p> <p><b>Additional input:</b> the additional input bit lengths have the same defaults and restrictions as the personalization string lengths.</p>
<b>Pass/Fail with Explanation</b>	<p>Algorithm: Hash DRBG (SHA2-256)  Mode: Hash_DRBG  CAVP #: A5110, A2583, A4982</p>

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

## 7.4 TLSS

### 7.4.1 FCS\_TLSS\_EXT.1.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.</li> <li>• Verify the required ciphersuite with packet capture.</li>   <li>• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA.</li> <li>• Verify the required ciphersuite with packet capture.</li> </ul>

	<ul style="list-style-type: none"> <li>Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA.</li> <li>Verify the required ciphersuite with packet capture.</li> </ul> <ul style="list-style-type: none"> <li>Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256.</li> <li>Verify the required ciphersuite with packet capture.</li> </ul> <ul style="list-style-type: none"> <li>Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384.</li> <li>Verify the required ciphersuite with packet capture.</li> </ul> <ul style="list-style-type: none"> <li>Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256.</li> <li>Verify the required ciphersuite with packet capture.</li> </ul> <ul style="list-style-type: none"> <li>Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384.</li> <li>Verify the required ciphersuite with packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>TOE should successfully establish the TLS connection with claimed cipher suites.</li> <li>Packet captures should show the successful establishment of TLS connection with configured ciphersuites.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE can establish a TLS connection using each of the ciphersuites specified by the requirement. This meets the test requirements.

7.4.2 FCS\_TLSS\_EXT.1.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	Test 2: The evaluator shall send a Client Hello to the server with a list of ciphersuites that does not contain any of the ciphersuites in the server's ST and verify that the server denies the connection. Additionally, the evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the server denies the connection.
<b>Test Steps</b>	Unclaimed ciphersuite: <ul style="list-style-type: none"> <li>Using the acumen-tls tool as a client, attempt to establish a TLS connection to the TOE using an unsupported ciphersuite in the Client Hello: - TLS_RSA_WITH_3DES_EDE_CBC_SHA.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify the failure logs on TOE showing failure due to no shared cipher.</li> <li>• Verify the connection fails via packet capture.</li> </ul> <p>TLS_NULL_WITH_NULL_NULL ciphersuite:</p> <ul style="list-style-type: none"> <li>• Using the acumen-tls tool as a client, attempt to establish a TLS connection to the TOE using TLS_NULL_WITH_NULL_NULL ciphersuite in the client hello and verify the connection fails.</li> <li>• Verify the failure logs on TOE showing failure due to no shared cipher.</li> <li>• Verify the connection fails via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject the connection when an unsupported ciphersuite is presented.</li> <li>• Packet capture should show handshake failure with unsupported ciphersuites.</li> <li>• The TOE log should show handshake failure due to no shared cipher.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects TLS connections with the unsupported ciphersuites. This meets the testing requirement.

7.4.3 FCS\_TLSS\_EXT.1.1 TEST #3A

Item	Data
<b>Test Assurance Activity</b>	Test 3: The evaluator shall perform the following modifications to the traffic: <ul style="list-style-type: none"> <li>a) Modify a byte in the Client Finished handshake message, and verify that the server rejects the connection and does not send any application data.</li> </ul>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Run the acumen-tlss tool as a client with a modified client finished message and wait for the connection, the connection should fail.</li> <li>• Verify the failure logs on the device showing SSL Error: decryption failed.</li> <li>• Verify the unsuccessful connection via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject a connection when the byte in the client’s finished handshake message is modified.</li> <li>• Packet capture should show connection failure when the Client Finished handshake message is modified.</li> <li>• TOE logs should show SSL Error: decryption failed.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects the connection after receiving the modified Client Handshake message. This meets the test requirements.

7.4.4 FCS\_TLSS\_EXT.1.1 TEST #3B

Item	Data
------	------



<p><b>Test Assurance Activity</b></p>	<p>Test 3: The evaluator shall perform the following modifications to the traffic:</p> <p>b) (Test Intent: The intent of this test is to ensure that the server's TLS implementation immediately makes use of the key exchange and authentication algorithms to: a) Correctly encrypt (D)TLS Finished message and b) Encrypt every (D)TLS message after session keys are negotiated.)</p> <p>The evaluator shall use one of the claimed ciphersuites to complete a successful handshake and observe transmission of properly encrypted application data. The evaluator shall verify that no Alert with alert level Fatal (2) messages were sent.</p> <p>The evaluator shall verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. The evaluator shall examine the Finished message (encrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 11 22 33 44 55...) and confirm that it does not contain unencrypted data (unencrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 14 00 00 0c...), by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages. There is a chance that an encrypted Finished message contains a hexadecimal value of '14' at the position where a plaintext Finished message would contain the message type code '14'. If the observed Finished message contains a hexadecimal value of '14' at the position where the plaintext Finished message would contain the message type code, the test shall be repeated three times in total. In case the value of '14' can be observed in all three tests it can be assumed that the Finished message has indeed been sent in plaintext and the test has to be regarded as 'failed'. Otherwise it has to be assumed that the observation of the value '14' has been due to chance and that the Finished message has indeed been sent encrypted. In that latter case the test shall be regarded as 'passed'.</p>
<p><b>Test Steps</b></p>	<ul style="list-style-type: none"> <li>• Initiate a connection to the TOE with acumen-tls tool as a client.</li> <li>• Verify that no Alert with alert level Fatal (2) messages were sent.</li> <li>• Verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message.</li> <li>• Examine the Finished message and confirm that it does not contain unencrypted data by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• TOE should reject a connection when text is not encrypted otherwise it should succeed.</li> </ul>

	<ul style="list-style-type: none"> <li>Packet capture should show the message is encrypted hence the connection is successful.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The Finished message contains Hexadecimal 16 and is sent immediately after Hexadecimal 14 in the ChangeCipherSpec message. The first byte of the encrypted Finished message does not equal hexadecimal 14. This meets the testing requirement.

#### 7.4.5 FCS\_TLSS\_EXT.1.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall send a Client Hello requesting a connection for all mandatory and selected protocol versions in the SFR (e.g. by enumeration of protocol versions in a test client) and verify that the server denies the connection for each attempt.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Use the acumen-tls tool as a client to initiate a connection to the TOE and verify the connections fails for all the non-supported SSL and TLS versions.</li> <li>Attempt an SSL v2.0 connection to the TOE.</li> <li>Verify the failed connection with the logs.</li> <li>Verify that the connection was denied with the packet capture.</li> <li>Attempt an SSL v3.0 connection to the TOE.</li> <li>Verify the failed connection with the logs.</li> <li>Verify that the connection was denied with the packet capture.</li> <li>Attempt an TLS v1.0 connection to the TOE.</li> <li>Verify the failed connection with the logs.</li> <li>Verify that the connection was denied with the packet capture.</li> <li>Attempt an TLS v1.1 connection to the TOE.</li> <li>Verify the failed connection with the logs.</li> <li>Verify that the connection was denied with the packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The TOE should reject a connection when a client requests a connection with the unsupported TLS/SSL versions.</li> <li>TOE logs should show connection failure due to an unknown protocol.</li> <li>Packet capture should show a connection reset due to an unsupported protocol version.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects all SSLv2, SSLv3, TLS v1.0, and TLS v1.1 connection attempts. This meets the testing requirement.

#### 7.4.6 FCS\_TLSS\_EXT.1.3 TEST #1A

Item	Data
------	------

<b>Test Assurance Activity</b>	<p>Test 1: [conditional] If ECDHE ciphersuites are supported:</p> <p>a) The evaluator shall repeat this test for each supported elliptic curve. The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single supported elliptic curve specified in the Elliptic Curves Extension. The Evaluator shall verify (through a packet capture or instrumented client) that the TOE selects the same curve in the Server Key Exchange message and successfully establishes the connection.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Connect to the TOE using secp256r1 and verify that it is successful.</li> <li>• Verify with packet capture that the connection is established using the curve secp256r1.</li> <li>• Connect to the TOE using secp384r1 and verify that it is successful.</li> <li>• Verify with packet capture that the connection is established using the curve secp384r1.</li> <li>• Connect to the TOE using secp521r1 and verify that it is successful.</li> <li>• Verify with packet capture that the connection is established using the curve secp521r1.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The connection should be successful when a supported ECDHE cipher and elliptic curve are configured.</li> <li>• Packet capture should show a successful connection and the supported elliptic curve used.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE was able to make a connection using each supported elliptic curve. This meets the test requirements.</p>

7.4.7 FCS\_TLSS\_EXT.1.3 TEST #1B

Item	Data
<b>Test Assurance Activity</b>	<p>Test 1: [conditional] If ECDHE ciphersuites are supported:</p> <p>b) The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single unsupported elliptic curve (e.g. secp192r1 (0x13)) specified in RFC4492, chap. 5.1.1. The evaluator shall verify that the TOE does not send a Server Hello message and the connection is not successfully established.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Run the acumen-tls tool as a client, establish a connection to TOE over TLS using the supported ciphersuite and unsupported elliptical curve and verify the connection fails.</li> <li>• Verify the log on the device showing failure due to no shared cipher.</li> <li>• Verify the packet capture showing connection failure.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject the connection when the supported cipher and the unsupported elliptic curve are configured.</li> </ul>

	<ul style="list-style-type: none"> <li>• Packet capture should show connection failure with the unsupported elliptic curve.</li> <li>• Logs should show handshake failure due to no shared cipher.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects a connection with unsupported elliptic curves. This meets the testing requirements.

#### 7.4.8 FCS\_TLSS\_EXT.1.3 TEST #2

Item	Data
<b>Test Assurance Activity</b>	Test 2: [conditional] If DHE ciphersuites are supported, the evaluator shall repeat the following test for each supported parameter size. If any configuration is necessary, the evaluator shall configure the TOE to use a supported Diffie-Hellman parameter size. The evaluator shall attempt a connection using a supported DHE ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a Server Key Exchange Message where p Length is consistent with the message are the ones configured Diffie-Hellman parameter size(s).
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since DHE ciphersuites are not supported by the TOE.

#### 7.4.9 FCS\_TLSS\_EXT.1.3 TEST #3

Item	Data
<b>Test Assurance Activity</b>	Test 3: [conditional] If RSA key establishment ciphersuites are supported, the evaluator shall repeat this test for each RSA key establishment key size. If any configuration is necessary, the evaluator shall configure the TOE to perform RSA key establishment using a supported key size (e.g. by loading a certificate with the appropriate key size). The evaluator shall attempt a connection using a supported RSA key establishment ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a certificate whose modulus is consistent with the configured RSA key size.
<b>Pass/Fail with Explanation</b>	N/A. The RSA key establishment ciphersuites are not supported.

#### 7.4.10 FCS\_TLSS\_EXT.1.4 TEST #1 [TD0569]

Item	Data
------	------

<b>Test Assurance Activity</b>	<p><i>Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).</i></p> <p>Test 1 [conditional]: If the TOE does not support session resumption based on session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2) or session tickets according to RFC5077, the evaluator shall perform the following test:</p> <ol style="list-style-type: none"> <li>a) The client sends a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket.</li> <li>b) The client verifies the server does not send a NewSessionTicket handshake message (at any point in the handshake).</li> <li>c) The client verifies the Server Hello message contains a zero-length session identifier or passes the following steps: Note: The following steps are only performed if the ServerHello message contains a non-zero length SessionID.</li> <li>d) The client completes the TLS handshake and captures the SessionID from the ServerHello.</li> <li>e) The client sends a ClientHello containing the SessionID captured in step d). This can be done by keeping the TLS session in step d) open or start a new TLS session using the SessionID captured in step d).</li> <li>f) The client verifies the TOE (1) implicitly rejects the SessionID by sending a ServerHello containing a different SessionID and by performing a full handshake (as shown in Figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.</li> </ol> <p>Remark: If multiple contexts are supported for session resumption, the session ID or session ticket may be obtained in one context for resumption in another context. It is possible that one or more contexts may only permit the construction of sessions to be reused in other contexts but not actually permit resumption themselves. For contexts which do not permit resumption, the evaluator is required to verify this behaviour subject to the description provided in the TSS. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.</p> <p><b>TD0569 has been applied.</b></p>
<b>Pass/Fail with Explanation</b>	<p>N/A. This test is not applicable since the TOE supports session resumption based on session tickets.</p>

7.4.11 FCS\_TLSS\_EXT.1.4 TEST #2A [TD0569]

Item	Data
------	------

<b>Test Assurance Activity</b>	<p>Test 2 [conditional]: If the TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2), the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):</p> <p>a) The evaluator shall conduct a successful handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then initiate a new TLS connection and send the previously captured session ID to show that the TOE resumed the previous session by responding with ServerHello containing the same SessionID immediately followed by ChangeCipherSpec and Finished messages (as shown in Figure 2 of RFC 4346 or RFC 5246).</p> <p>Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.</p> <p><b>TD0569 has been applied.</b></p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-tlss tool to connect to the TOE.</li> <li>• Verify packet capture contains two TLS handshakes with the TOE and the same session ID is sent through the next session's server hello.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should establish a successful TLS client connection when the session ID of the previous session is sent in the server Hello.</li> <li>• The packet capture should show that the same session ID is sent through the next session's server hello thus, establishing a successful connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE responds with an abbreviated handshake when the session ID is reused. This meets the testing requirements.</p>

7.4.12 FCS\_TLSS\_EXT.1.4 TEST #2B [TD0569]

Item	Data
<b>Test Assurance Activity</b>	<p>Test 2 [conditional]: If the TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2), the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):</p>

	<p>b) The evaluator shall initiate a handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then, within the same handshake, generate or force an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message thereby disrupting the handshake. The evaluator shall then initiate a new Client Hello using the previously captured session ID, and verify that the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID and performing a full handshake (as shown in figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.</p> <p>Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.</p> <p><b>TD0569 has been applied.</b></p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-tlss tool to connect to the TOE.</li> <li>• Verify packet capture contains session ID in the Server Hello message and within the same handshake, generate an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message.</li> <li>• Verify session ID in the new client hello matches the session ID in the previous Server hello and the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID.</li> <li>• Verify handshake failure logs.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should close the TLS client connection established by the ‘acumen-tlss tool’ when it sends an alert message.</li> <li>• The packet capture should show that the TOE implicitly rejects the previous session ID by performing a full handshake, sending a new session ID, and allowing the flow of application data.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE implicitly rejects the modified session ID by performing a full handshake, sending a new session ID, and allowing the flow of application data. This meets the testing requirements.</p>

Item	Data
<b>Test Assurance Activity</b>	<p>Test 3 [conditional]: If the TOE supports session tickets according to RFC5077, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):</p> <p>a) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the session ticket in the ClientHello. The evaluator shall confirm that the TOE responds with an abbreviated handshake described in section 3.1 of RFC 5077 and illustrated with an example in figure 2. Of particular note: if the server successfully verifies the client's ticket, then it may renew the ticket by including a NewSessionTicket handshake message after the ServerHello in the abbreviated handshake (which is shown in figure 2). This is not required, however as further clarified in section 3.3 of RFC 5077.</p> <p>Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.</p> <p><b>TD0556 and TD0569 has been applied.</b></p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-tlss tool to connect to the TOE.</li> <li>• Verify packet capture contains two TLS handshakes with the TOE and the same session ticket is sent through the next session's client hello.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should establish a successful TLS client connection when the session ticket of the previous session is sent in ClientHello.</li> <li>• The packet capture should verify that the same session ticket is sent through the next session's client hello thus, establishing a successful connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE responds with an abbreviated handshake when the session ticket is reused. This meets the testing requirements.

#### 7.4.14 FCS\_TLSS\_EXT.1.4 TEST #3B [TD0569]

Item	Data
------	------



<p><b>Test Assurance Activity</b></p>	<p>Test 3 [conditional]: If the TOE supports session tickets according to RFC5077, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS):</p> <ul style="list-style-type: none"> <li>b) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator will then modify the session ticket and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake (as shown in figure 3 or 4 of RFC 5077), or (2) terminates the connection in some way that prevents the flow of application data.</li> </ul> <p>Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.</p> <p><b>TD0569 has been applied.</b></p>
<p><b>Test Steps</b></p>	<ul style="list-style-type: none"> <li>• Use the acumen-tlss tool to connect to the TOE.</li> <li>• Verify packet capture contains two TLS handshakes with the TOE.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• The TOE should close the TLS client connection that is established by the ‘acumen-tlss tool’ which sends the modified session ticket.</li> <li>• The packet capture should show TOE implicitly rejects the modified session ticket by performing a full handshake by sending a new session ticket and allowing the flow of application data.</li> </ul>
<p><b>Pass/Fail with Explanation</b></p>	<p>Pass. TOE implicitly rejects the modified session ticket by performing a full handshake by sending a new session ticket and allowing the flow of application data. This meets the testing requirements.</p>

## 7.5.1 FPT\_TST\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>It is expected that at least the following tests are performed:</p> <ol style="list-style-type: none"> <li>Verification of the integrity of the firmware and executable software of the TOE</li> <li>Verification of the correct operation of the cryptographic functions necessary to fulfil any of the SFRs.</li> </ol> <p>Although formal compliance is not mandated, the self-tests performed should aim for a level of confidence comparable to:</p> <ol style="list-style-type: none"> <li>[FIPS 140-2], chap. 4.9.1, Software/firmware integrity test for the verification of the integrity of the firmware and executable software. Note that the testing is not restricted to the cryptographic functions of the TOE.</li> <li>[FIPS 140-2], chap. 4.9.1, Cryptographic algorithm test for the verification of the correct operation of cryptographic functions. Alternatively, national requirements of any CCRA member state for the security evaluation of cryptographic functions should be considered as appropriate.</li> </ol> <p>The evaluator shall either verify that the self-tests described above are carried out during initial start-up or that the developer has justified any deviation from this.</p> <p>For distributed TOEs the evaluator shall perform testing of self-tests on all TOE components according to the description in the TSS about which self-test are performed by which component.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Login to the TOE through the Console using credentials.</li> <li>Restart the TOE using the command “restart”.</li> <li>Observe the boot process for self-tests. (Self-tests during Power ON)</li> <li>Verify through logs that the self-test has been completed successfully.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The TOE should execute all claimed self-tests during bootup.</li> <li>Evidence (screenshot or CLI output) and log showing successful self-tests.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE successfully executes self-test. This meets the testing requirement.

## 7.5.2 FPT\_TUD\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following tests:

	<p>Test 1: The evaluator performs the version verification activity to determine the current version of the product. If a trusted update can be installed on the TOE with a delayed activation, the evaluator shall also query the most recently installed version (for this test the TOE shall be in a state where these two versions match). The evaluator obtains a legitimate update using procedures described in the guidance documentation and verifies that it is successfully installed on the TOE. For some TOEs loading the update onto the TOE and activation of the update are separate steps ('activation' could be performed e.g. by a distinct activation step or by rebooting the device). In that case the evaluator verifies after loading the update onto the TOE but before activation of the update that the current version of the product did not change but the most recently installed version has changed to the new product version. After the update, the evaluator performs the version verification activity again to verify the version correctly corresponds to that of the update and that current version of the product and most recently installed version match again.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Verify the current firmware version of the TOE.</li> <li>• Click on the 'Upload Firmware' option and upload the image on the TOE.</li> <li>• Check if the image is uploaded on the TOE.</li> <li>• Verify that after loading the new image onto the TOE but before activation of the new image the current version of the product did not change.</li> <li>• Boot the uploaded image and wait for some time until the booting is done.</li> <li>• After the update, perform the version verification activity again to verify the version correctly corresponds to that of the update and that the current version of the product and the most recently installed version match again.</li> <li>• Verify through logs that the firmware is upgraded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should successfully update the current version with the new version after verifying the integrity of the new image.</li> <li>• Evidence - Screenshot showing new version post upgrade.</li> <li>• TOE logs should show successful image installation.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE can be successfully updated. This meets the testing requirements.</p>

7.5.3 FPT\_TUD\_EXT.1 TEST #2 (A)

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall perform the following tests:</p> <p>Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates:</p> <ol style="list-style-type: none"> <li>1) A modified version (e.g. using a hex editor) of a legitimately signed update</li> </ol> <p>If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<p><b>Test Steps</b></p>	<ul style="list-style-type: none"> <li>• Verify the current firmware version on the TOE.</li> <li>• Upload the modified image on the TOE and verify that it fails.</li> <li>• Verify the update failure with logs.</li> <li>• Verify that the TOE firmware version has not changed.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• The TOE should detect and reject the modified image for software update.</li> <li>• Evidence (screenshot or CLI output) showing the old version before and after the update attempt.</li> <li>• TOE logs should show the failure of the software update.</li> </ul>
<p><b>Pass/Fail with Explanation</b></p>	<p>Pass. The TOE can detect and reject the modified image. This meets the testing requirements.</p>

7.5.4 FPT\_TUD\_EXT.1 TEST #2 (B)

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall perform the following tests:</p> <p>Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates:</p> <p>2) An image that has not been signed</p> <p>If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<p><b>Test Steps</b></p>	<ul style="list-style-type: none"> <li>• Verify the current firmware version on the TOE.</li> <li>• Upload an image with no signature on the TOE and verify that it fails.</li> <li>• Verify the update failure with logs.</li> <li>• Verify that the TOE firmware version has not changed.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• The TOE should detect and reject the image without signature for software update.</li> <li>• Evidence (screenshot or CLI output) showing the old version before and after the update attempt.</li> </ul>

	<ul style="list-style-type: none"> <li>TOE logs should show the failure of the software update.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE can detect and reject the image without signature. This meets the testing requirements.

#### 7.5.5 FPT\_TUD\_EXT.1 TEST #2 (C)

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates:</p> <p>3) An image signed with an invalid signature (e.g. by using a different key as expected for creating the signature or by manual modification of a legitimate signature)</p> <p>If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Verify the current firmware version on the TOE.</li> <li>Upload the image with the invalid signature on the TOE and verify that it fails.</li> <li>Verify the update failure with logs.</li> <li>Verify that the TOE firmware version has not changed.</li> </ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should detect and reject the image with an invalid signature for software update.</li> <li>• Evidence (e.g., screenshot or CLI output) showing the old version before and after the update attempt.</li> <li>• TOE logs should show the failure of the software update.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE can detect and reject the image with an invalid signature. This meets the testing requirements.

### 7.5.6 FPT\_TUD\_EXT.1 TEST #3 (A)

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 3 [conditional]: If the TOE itself verifies a hash value over an image against a published hash value (i.e. reference value) that has been imported to the TOE from outside such that the TOE itself authorizes the installation of an image to update the TOE, the following test shall be performed (otherwise the test shall be omitted. If the published hash is provided to the TOE by the Security Administrator and the verification of the hash value over the update file(s) against the published hash is performed by the TOE, then the evaluator shall perform the following tests. The evaluator first confirms that no update is pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test.</p> <ol style="list-style-type: none"> <li>1) The evaluator obtains or produces an illegitimate update such that the hash of the update does not match the published hash. The evaluator provides the published hash value to the TOE and calculates the hash of the update either on the TOE itself (if that functionality is provided by the TOE), or else outside the TOE. The evaluator confirms that the hash values are different, and attempts to install the update on the TOE, verifying that this fails because of the difference in hash values (and that the failure is logged). Depending on the implementation of the TOE, the TOE might not allow the Security Administrator to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE</li> </ol> <p>If the TOE allows delayed activation of updates, the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs. Depending on the point in time when the attempted update is rejected, the most recently installed version might or might not be updated. The evaluator shall verify that the TOE</p>

	<p>handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt.</p> <p>If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since the TOE does not verify images based on published hash values.

7.5.7 FPT\_TUD\_EXT.1 TEST #3 (B)

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 3 [conditional]: If the TOE itself verifies a hash value over an image against a published hash value (i.e. reference value) that has been imported to the TOE from outside such that the TOE itself authorizes the installation of an image to update the TOE, the following test shall be performed (otherwise the test shall be omitted. If the published hash is provided to the TOE by the Security Administrator and the verification of the hash value over the update file(s) against the published hash is performed by the TOE, then the evaluator shall perform the following tests. The evaluator first confirms that no update is pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test.</p>



	<p>2) The evaluator uses a legitimate update and tries to perform verification of the hash value without providing the published hash value to the TOE. The evaluator confirms that this attempt fails. Depending on the implementation of the TOE it might not be possible to attempt the verification of the hash value without providing a hash value to the TOE, e.g. if the hash value needs to be handed over to the TOE as a parameter in a command line message and the syntax check of the command prevents the execution of the command without providing a hash value. In that case the mechanism that prevents the execution of this check shall be tested accordingly, e.g. that the syntax check rejects the command without providing a hash value, and the rejection of the attempt is regarded as sufficient verification of the correct behaviour of the TOE in failing to verify the hash. The evaluator then attempts to install the update on the TOE (in spite of the unsuccessful hash verification) and confirms that this fails. Depending on the implementation of the TOE, the TOE might not allow to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE</p> <p>If the TOE allows delayed activation of updates, the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs. Depending on the point in time when the attempted update is rejected, the most recently installed version might or might not be updated. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt.</p> <p>If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped.</p> <p>The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).</p> <p>For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since the TOE does not verify images based on published hash values.

## 7.6.1 FIA\_X509\_EXT.1.1/REV TEST #1A

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 1a: The evaluator shall present the TOE with a valid chain of certificates (terminating in a trusted CA certificate) as needed to validate the leaf certificate to be used in the function and shall use this chain to demonstrate that the function succeeds. Test 1a shall be designed in a way that the chain can be 'broken' in Test 1b by either being able to remove the trust anchor from the TOEs trust store, or by setting up the trust store in a way that at least one intermediate CA certificate needs to be provided, together with the leaf certificate from outside the TOE, to complete the chain (e.g. by storing only the root CA certificate in the trust store).</p>
<b>Test Steps</b>	<p>The option for using X.509 certificates for self-testing, as well as FPT_TUD_EXT.2, is not selected in the ST.</p> <p>Using RSA Certificates:</p> <ul style="list-style-type: none"> <li>• Import the CA, ICA certificate for TOE and ICA certificate for the PEER.</li> <li>• Import a TOE's leaf certificate signed by the TOE's ICA certificate into TOE.</li> <li>• Configure the TOE for IKE/IPsec connection with the PEER (Strongswan).</li> <li>• Configure the PEER (Strongswan) for IKE/IPsec connection with the TOE.</li> <li>• Enable the OCSP responder using OpenSSL.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is established due to a valid chain of certificates.</li> <li>• Verify that the connection is successful via audit logs.</li> <li>• Verify that the connection is successful via packet capture.</li> </ul> <p>Using ECDSA Certificate:</p> <ul style="list-style-type: none"> <li>• Import the CA, ICA certificate for TOE and ICA certificate for the PEER.</li> </ul>

	<ul style="list-style-type: none"> <li>• Import a TOE's leaf certificate signed by the TOE's ICA certificate into TOE.</li> <li>• Configure the TOE for IKE/IPsec connection with the PEER (Strongswan).</li> <li>• Configure the PEER (Strongswan) for IKE/IPsec connection with the TOE.</li> <li>• Enable the OCSP responder using OpenSSL.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is established due to a valid chain of certificates.</li> <li>• Verify that the connection is successful via audit logs.</li> <li>• Verify that the connection is successful via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When a complete certificate chain is present, the TOE should establish a successful TLS connection.</li> <li>• Packet capture and logs should show a successful connection as a complete chain of certificates is present on the TOE.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE can establish a successful connection when a complete certificate trust chain is present. This meets the test requirements.

#### 7.6.2 FIA\_X509\_EXT.1.1/REV TEST #1B

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 1b: The evaluator shall then 'break' the chain used in Test 1a by either removing the trust anchor in the TOE's trust store used to terminate the chain, or by removing one of the intermediate CA certificates (provided together with the leaf certificate in Test 1a) to complete the chain. The evaluator shall show that an attempt to validate this broken chain fails.</p>
<b>Test Steps</b>	<p>The option for using X.509 certificates for self-testing, as well as FPT_TUD_EXT.2, is not selected in the ST.</p> <p>This test is performed in continuation with Test #1a. Using RSA Certificates:</p>

	<ul style="list-style-type: none"> <li>• Delete VM’s intermediate certificate from the TOE’s truststore.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is not established due to an invalid chain of certificates.</li> <li>• Verify that the connection fails via audit logs.</li> <li>• Verify that the connection failed via packet capture.</li> </ul> <p>Using EC Certificates:</p> <ul style="list-style-type: none"> <li>• Delete VM’s intermediate certificate from the TOE’s truststore.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is not established due to an invalid chain of certificates.</li> <li>• Verify that the connection fails via audit logs.</li> <li>• Verify that the connection failed via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should reject the connection when an incomplete certificate trust chain is present.</li> <li>• TOE logs and the packet capture should show an unsuccessful IPsec connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects the connection when an incomplete certificate trust chain is present. This meets the test requirements.

7.6.3 FIA\_X509\_EXT.1.1/REV TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.</p>
<b>Test Steps</b>	<p>The option for using X.509 certificates for self-testing, as well as FPT_TUD_EXT.2, is not selected in the ST.</p> <p><b>PART1:</b></p>

	<p>Pass expired end entity through IPSEC connection:</p> <ul style="list-style-type: none"> <li>• Generate a PEER certificate that has expired.</li> <li>• Initiate an IPsec connection and verify that the connection fails since the PEER certificate is expired.</li> <li>• Verify on the TOE that no tunnel is established.</li> <li>• Verify connection failure logs.</li> <li>• Verify the unsuccessful connection via Packet capture.</li> </ul> <p><b>PART2:</b></p> <p>Try to load the expired certificate on the TOE:</p> <ul style="list-style-type: none"> <li>• Import the expired end entity on to the TOE.</li> <li>• Verify that the TOE rejects the upload attempt and issues logs for the same.</li> <li>• Create an expired ICA using XCA.</li> <li>• Import the expired ICA onto the TOE.</li> <li>• Verify that the TOE rejects the upload attempt and issues logs for the same.</li> </ul> <p><b>PART3:</b></p> <p>Try to use a certificate which expired while on the TOE:</p> <ul style="list-style-type: none"> <li>• Import a certificate that will expire in the near future, on to the TOE.</li> <li>• Use the imported certificate in an IPsec connection and verify that the connection is successful.</li> <li>• Terminate the IPsec tunnel and wait for the certificate to expire while on the TOE.</li> <li>• Attempt to use the expired certificate in an IPsec connection and verify that the connection fails.</li> <li>• Verify the connection failure logs.</li> <li>• Verify the connection failure via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not accept a connection because of the expired certificate.</li> <li>• The TOE should generate an error when importing an expired certificate.</li> <li>• TOE logs should show connection failure due to an expired server certificate.</li> <li>• Packet capture should show connection failure as an expired server certificate is used.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE rejects the connection when an expired certificate is presented and generates an error when an attempt is made to upload an expired certificate. This meets the test requirements.</p>

7.6.4 FIA\_X509\_EXT.1.1/REV TEST #3

Item	Data
------	------

<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 3: The evaluator shall test that the TOE can properly handle revoked certificates— conditional on whether CRL or OCSP is selected; if both are selected, then a test shall be performed for each method. The evaluator shall test revocation of the peer certificate and revocation of the peer intermediate CA certificate i.e. the intermediate CA certificate should be revoked by the root CA. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails. Revocation checking is only applied to certificates that are not designated as trust anchors. Therefore, the revoked certificate(s) used for testing shall not be a trust anchor.</p>
<p><b>Test Steps</b></p>	<p>The option for using X.509 certificates for self-testing, as well as FPT_TUD_EXT.2, is not selected in the ST.</p> <p>The CRL is not selected in the ST, TOE only supports revocation checking using OCSP.</p> <p>Successful Connection:</p> <ul style="list-style-type: none"> <li>• Import the CA, ICA certificate for TOE and ICA certificate for the PEER.</li> <li>• Import a TOE’s leaf certificate signed by the TOE’s ICA certificate into TOE.</li> <li>• Configure the TOE for IKE/IPsec connection with the PEER (Strongswan).</li> <li>• Configure the PEER (Strongswan) for IKE/IPsec connection with the TOE.</li> <li>• Generate and export the index file. Verify that all the certificates are valid in the OCSP responder index file.</li> <li>• Enable the OCSP responder using OpenSSL.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is established due to a valid chain of certificates.</li> <li>• Verify that the connection is successful via audit logs.</li> <li>• Verify that the connection is successful via packet capture.</li> </ul> <p>Revoked PEER End Entity certificate:</p> <ul style="list-style-type: none"> <li>• Revoke the PEER end entity certificate.</li> </ul>

	<ul style="list-style-type: none"> <li>• Generate and export the index file. Verify that the peer certificate is revoked in the OCSP responder index file.</li> <li>• Enable the OCSP responder using openssl.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is unsuccessful due to a revoked certificate.</li> <li>• Verify that the connection is unsuccessful via audit logs.</li> <li>• Verify that the connection is unsuccessful via packet capture.</li> </ul> <p>Revoked ICA certificate:</p> <ul style="list-style-type: none"> <li>• Unrevoked the previously revoked PEER End Entity and revoke the ICA certificate.</li> <li>• Generate and export the index file. Verify that the peer intermediate certificate is revoked in the OCSP responder index file.</li> <li>• Enable the OCSP responder using openssl.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is unsuccessful due to a revoked certificate.</li> <li>• Verify that the connection is unsuccessful via audit logs.</li> <li>• Verify that the connection is unsuccessful via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• A successful connection should be established when connecting with valid certificates.</li> <li>• The connection should fail when connecting with a revoked peer certificate.</li> <li>• The connection should fail when connecting with a revoked intermediate certificate.</li> <li>• TOE logs should show the OSCP checking status.</li> <li>• Packet capture should show whether the connection is successful or unsuccessful.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The CRL is not selected in the ST, TOE only supports revocation checking using OCSP. It successfully connects with unrevoked certificates and rejects connections with revoked certificates. This meets the testing requirements.

7.6.5 FIA\_X509\_EXT.1.1/REV TEST #4

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for

	<p>FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 4: If OCSP is selected, the evaluator shall configure the OCSP server or use a man-in-the-middle tool to present a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.</p>
<b>Test Steps</b>	<p>The option for using X.509 certificates for self-testing, as well as FPT_TUD_EXT.2, is not selected in the ST.</p> <p>The CRL is not selected in the ST, TOE only supports revocation checking using OCSP.</p> <ul style="list-style-type: none"> <li>• Configure the CA signing the OCSP to use a signing certificate that does not have the OCSP signing key usage bit set.</li> <li>• Attempt a connection with the peer (will fail).</li> <li>• Verify that the tunnel is not established.</li> <li>• Verify the failure of validation of an OCSP response via an OCSP responder.</li> <li>• Verify that certificate validation failed because the signer CA certificate does not have the OCSP signing EKU via TOE logs.</li> <li>• Verify the unsuccessful IPsec connection with the help of packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should reject the connection when the CA signing the OCSP does not have the OCSP signing key usage bit set.</li> <li>• Evidence (screenshot or CLI output) showing the addition of certificates.</li> <li>• Log should show an unsuccessful connection.</li> <li>• Packet capture should show an unsuccessful connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The CRL is not selected in the ST, TOE only supports revocation checking using OCSP. The TOE rejects connections when the delegated signer certificate in OCSP is invalid and does not have OCSP-signer EKU. This meets the testing requirements.</p>

7.6.6 FIA\_X509\_EXT.1.1/REV TEST #5

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of</p>



	<p>X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 5: The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-strongswan tool to initiate an IPsec connection after modifying a byte in the first 8 bytes of the certificate and verify that the connection fails.</li> <li>• Verify on the TOE that no tunnel is established.</li> <li>• Verify via logs that an error is generated while establishing the IPsec tunnel.</li> <li>• Verify via packet capture that the session is not established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject the connection when the first 8 bytes of the certificate are modified.</li> <li>• TOE should generate error logs when a certificate with modified bytes is presented.</li> <li>• Packet capture should show connection failure due to a certificate with modified bytes being presented.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. TOE rejects connections when the first 8 bytes of the certificate are modified. This meets the test requirements.</p>

7.6.7 FIA\_X509\_EXT.1.1/REV TEST #6

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p>

	Test 6: The evaluator shall modify any byte in the certificate signatureValue field (see RFC5280 Sec. 4.1.1.3), which is normally the last field in the certificate, and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-strongswan tool to initiate an IPsec connection after modifying any byte in the certificate's signatureValue field and verify that the connection fails.</li> <li>• Verify on the TOE that no tunnel is established.</li> <li>• Verify via logs that an error is generated while establishing the IPsec tunnel.</li> <li>• Verify via packet capture that the session is not established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject the connection when any byte of the certificate is modified.</li> <li>• TOE should generate error logs when a certificate with modified bytes is presented.</li> <li>• Packet capture should show connection failure due to a certificate with modified bytes being presented.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects connections when presented with a certificate that has any byte modified in the certificate's signatureValue field. This meets the test requirements.

#### 7.6.8 FIA\_X509\_EXT.1.1/REV TEST #7

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for selftesting is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:</p> <p>Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The hash of the certificate will not validate.)</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Use the acumen-strongswan tool to initiate an IPsec connection after modifying any byte in the certificate's public key and verify that the connection fails.</li> <li>• Verify on the TOE that no tunnel is established.</li> <li>• Verify via logs that an error is generated while establishing the IPsec tunnel.</li> <li>• Verify via packet capture that the session is not established.</li> </ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should reject the connections when the public key of the certificate is modified.</li> <li>• TOE should generate error logs while establishing the IPsec tunnel.</li> <li>• Packet capture should show the session is not established.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects connections when any byte is the public key of the certificate is modified. This meets the test requirements.

#### 7.6.9 FIA\_X509\_EXT.1.1/REV TEST #8A [TD0527]

Item	Data
<b>Test Assurance Activity</b>	<p><b>(Conditional on support for a minimum certificate path length of three certificates)</b>  <b>(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)</b>  <b>(Conditional on TOE ability to process CA certificates presented in certificate message)</b></p> <p>Test 8a: The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a valid chain of EC certificates (terminating in a trusted CA certificate), where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.</p> <p><b>TD0527 (12/1 Update) has been applied.</b></p>
<b>Pass/Fail with Explanation</b>	N/A. The TOE does not accept the presented PEER ICA certificate. Refer to test case FIA_X509_EXT.1.1/REV TEST #1B for evidence.

#### 7.6.10 FIA\_X509\_EXT.1.1/REV TEST #8B [TD0527]

Item	Data
<b>Test Assurance Activity</b>	<p><b>(Conditional on support for a minimum certificate path length of three certificates)</b>  <b>(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)</b>  <b>(Conditional on TOE ability to process CA certificates presented in certificate message)</b></p> <p>Test 8b: The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from</p>

	<p>outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a chain of EC certificates (terminating in a trusted CA certificate), where the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key information field, and is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.</p> <p><b>TD0527 (12/1 Update) has been applied.</b></p>
<b>Pass/Fail with Explanation</b>	N/A. The TOE does not accept the presented PEER ICA certificate. Refer to test case FIA_X509_EXT.1.1/REV TEST #1B for evidence.

7.6.11 FIA\_X509\_EXT.1.1/REV TEST #8C [TD0527]

Item	Data
<b>Test Assurance Activity</b>	<p><b>(Conditional on support for a minimum certificate path length of three certificates)</b>  <b>(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)</b></p> <p>Test 8c: The evaluator shall establish a subordinate CA certificate, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is accepted into the TOE's trust store. The evaluator shall then establish a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is rejected, and not added to the TOE's trust store.</p> <p><b>TD0527 (12/1 Update) has been applied.</b></p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Intermediate certificate is modified with a named curve with an explicit format in the public key information field using the x509-mod tool.</li> <li>• Configure the TOE for the root certificate as a trust anchor.</li> <li>• Attempt to add the modified Intermediate certificate on the TOE.</li> <li>• Verify that the TOE discards the certificate.</li> <li>• Verify error logs on the device showing the ICA certificate has an invalid public key.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject the certificate when the public key information is modified in the intermediate certificate.</li> <li>• Evidence (screenshot or CLI output) showing rejection of the certificates.</li> </ul>

<b>Pass/Fail with Explanation</b>	Pass. The evaluator verified that when the public key information is modified in the intermediate certificate and is loaded to the TOE's trust store, TOE does not accept such a certificate. This meets the testing requirements.
-----------------------------------	--

7.6.12 FIA\_X509\_EXT.1.2/REV TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted.</p> <p>The goal of the following tests is to verify that the TOE accepts a certificate as a CA certificate only if it has been marked as a CA certificate by using basicConstraints with the CA flag set to True (and implicitly tests that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation).</p> <p>For each of the following tests the evaluator shall create a chain of at least three certificates: a self-signed root CA certificate, an intermediate CA certificate and a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain).</p> <p>Test 1: The evaluator shall ensure that at least one of the CAs in the chain does not contain the basicConstraints extension. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points:</p> <ul style="list-style-type: none"> <li>i) as part of the validation of the leaf certificate belonging to this chain;</li> <li>ii) when attempting to add a CA certificate without the basicConstraints extension to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).</li> </ul> <p>The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection is distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to repeat the</p>

	tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create an Intermediate CA with no basic constraints.</li> <li>• Import the created Intermediate CA onto the TOE.</li> <li>• Ensure the TOE shows the certificate as invalid.</li> <li>• Verify logs showing invalid certificate has been uploaded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should reject certificates signed by a CA that do not contain the basicConstraints extension.</li> <li>• TOE should generate error logs showing the certificate does not contain the basicConstraints extension</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects certificates signed by a CA that do not contain the basicConstraints extension. This meets the test requirements.

7.6.13 FIA\_X509\_EXT.1.2/REV TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted.</p> <p>The goal of the following tests is to verify that the TOE accepts a certificate as a CA certificate only if it has been marked as a CA certificate by using basicConstraints with the CA flag set to True (and implicitly tests that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation).</p> <p>For each of the following tests the evaluator shall create a chain of at least three certificates: a self-signed root CA certificate, an intermediate CA certificate and a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain).</p>

	<p>Test 2: The evaluator shall ensure that at least one of the CA certificates in the chain has a basicConstraints extension in which the CA flag is set to FALSE. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points:</p> <ul style="list-style-type: none"> <li>i) As part of the validation of the leaf certificate belonging to this chain;</li> <li>ii) When attempting to add a CA certificate with the CA flag set to FALSE to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).</li> </ul> <p>The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection is distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to repeat the tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create an Intermediate CA with False basic constraints.</li> <li>• Import the created Intermediate CA onto the TOE.</li> <li>• Ensure the TOE shows the certificate as invalid.</li> <li>• Verify logs showing invalid certificate has been uploaded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should reject certificates signed by a CA that contain the basicConstraints extension set to False.</li> <li>• TOE should generate error logs showing the certificate contains the basicConstraints extension set to False.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE rejects certificates signed by a CA that contains basicConstraints extension set to false. This meets the test requirements.</p>

7.6.14 FIA\_X509\_EXT.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following test for each trusted channel:</p> <p>The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the guidance documentation to determine that all supported administrator-configurable options behave in their documented manner.</p>
<b>Test Steps</b>	<p>Successful Connection:</p> <ul style="list-style-type: none"> <li>• Import the CA, ICA certificate for TOE and ICA certificate for the PEER.</li> <li>• Import a TOE's leaf certificate signed by the TOE's ICA certificate into TOE.</li> </ul>

	<ul style="list-style-type: none"> <li>• Configure the TOE for IKE/IPsec connection with the PEER (Strongswan).</li> <li>• Configure the PEER (Strongswan) for IKE/IPsec connection with the TOE.</li> <li>• Enable the OSCP responder using OpenSSL.</li> <li>• Attempt to establish a connection between the TOE and PEER (Strongswan) and verify that the connection is established due to a valid chain of certificates.</li> <li>• Verify that the connection is successful via audit logs.</li> <li>• Verify that the connection is successful via packet capture.</li> </ul> <p>Manipulation of the environment:</p> <ul style="list-style-type: none"> <li>• Stop the OSCP responder from interrupting from listening.</li> <li>• Initiate the connection and verify that it fails.</li> <li>• Verify the failed connection via logs.</li> <li>• Verify the failed connection via Packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should establish a connection when certificate validation is successful.</li> <li>• The TOE should reject the certificate when validation checking of the certificate is not available.</li> <li>• Evidence (screenshot or CLI output) showing the configuration of OSCP and manipulation of the environment.</li> <li>• Log should show an unsuccessful connection.</li> <li>• Packet capture should show an unsuccessful connection.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE rejects certificates it cannot verify via OSCP when the responder is down. This meets the testing requirements.

7.6.15 FIA\_X509\_EXT.3 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: The evaluator shall use the guidance documentation to cause the TOE to generate a Certification Request. The evaluator shall capture the generated message and ensure that it conforms to the format specified. The evaluator shall confirm that the Certification Request provides the public key and other required information, including any necessary user-input information.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• From the TOE, generate a CSR.</li> <li>• Examine the CSR contents and ensure the CSR contains the following fields: <ul style="list-style-type: none"> <li>○ Public Key</li> <li>○ Common Name</li> <li>○ Organization</li> <li>○ Country</li> </ul> </li> </ul>



<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should be able to generate a CSR with all the requisite information as mentioned in the ST.</li> <li>• Evidence – snapshot showing required fields are configured</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE is able to generate a CSR with all of the requisite information. This meets the testing requirements.

#### 7.6.16 FIA\_X509\_EXT.3 TEST #2

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall demonstrate that validating a response message to a Certification Request without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the certificate response message and demonstrate that the function succeeds.</p>
<b>Test Steps</b>	<p>Part I</p> <ul style="list-style-type: none"> <li>• From the TOE, generate a CSR request.</li> <li>• Generate a signed certificate based on the generated CSR from an external intermediate CA.</li> <li>• Ensure that the full trust chain for the signed CA is not present on the TOE.</li> <li>• Attempt to load the signed certificate on the TOE.</li> <li>• The certificate is uploaded with the status Not verified because the full trust chain of the CA is not present.</li> <li>• As the certificate is not valid, it is not available for authentication use.</li> </ul> <p>Part II</p> <ul style="list-style-type: none"> <li>• Add the intermediary certificates to the TOE certificate store to ensure that the signing CA now has a full certificate path.</li> <li>• Verify uploaded signed certificate is validated once the full trust chain of the CA is present.</li> <li>• As the certificate is valid, it is available for authentication use.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not validate a signed CSR if the full trust chain is not present. When a full trust chain is present, the TOE should validate the signed CSR.</li> <li>• TOE should generate logs for certificate installation.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE only installs a CSR response signed by a CA with a full trust path and does not validate a signed CSR if the full trust chain is not present. This meets the testing requirements.

## 7.7 FIREWALL

### 7.7.1 FAU\_GEN.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	In addition to the Evaluation Activities specified in the Supporting Document for the Base-PP, the evaluator shall perform tests to demonstrate that audit records are generated for the auditable events as specified in Table 2 of the PP-Module and, if the optional SFR FFW_RUL_EXT.2 is claimed by the TOE, Table 3.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered under FAU_GEN.1 Test#1 from audit module. The optional SFR FFW_RUL_EXT.2 is not claimed by the TOE.

### 7.7.2 FFW\_RUL\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would otherwise be denied by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization.
<b>Test Steps</b>	<p>IPv4:</p> <ul style="list-style-type: none"><li>• Configure a deny rule to drop traffic for specific source and destination IP addresses on TOE.</li><li>• Keep sending traffic which matches the configured deny rule on the TOE.</li><li>• Verify with packet capture that traffic is being denied.</li><li>• Reboot the TOE while the ping is in progress.</li><li>• Continue sending traffic that matches the configured deny rule on the TOE.</li><li>• Verify with packet capture that traffic is being denied through the TOE.</li></ul> <p>IPv6:</p> <ul style="list-style-type: none"><li>• Configure a deny rule to drop traffic for specific source and destination IP addresses on TOE.</li><li>• Keep sending traffic which matches the configured deny rule on the TOE.</li><li>• Verify with packet capture that traffic is being denied.</li><li>• Reboot the TOE while the ping is in progress.</li><li>• Continue sending traffic that matches the configured deny rule on the TOE.</li><li>• Verify with packet capture that traffic is being denied through the TOE.</li></ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not allow any network traffic that is denied by the ruleset to pass through while it is being initialized.</li> <li>• TOE logs and Packet Capture should show that denied traffic is not passed through the TOE during TOE initialization.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. IPv4 and IPv6 packets that would otherwise be denied by the ruleset are not passed through the TOE during initialization. This meets the testing requirements.

7.7.3 FFW\_RUL\_EXT.1 TEST #2

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would be permitted by the ruleset should be sourced and be directed at a host. The evaluator shall verify using a packet sniffer that none of the generated network traffic is permitted through the firewall during initialization and is only permitted once initialization is complete.</p> <p>Note: The remaining testing associated with application of the ruleset is addressed in the subsequent test evaluation activities.</p>
<b>Test Steps</b>	<p><b>IPv4:</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to permit traffic for specific source and destination IP addresses on TOE.</li> <li>• Keep sending traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being allowed.</li> <li>• Reboot the TOE while the ping is in progress.</li> <li>• Continue sending the required traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being denied through the TOE while initializing.</li> <li>• Continue sending the required traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being allowed through the TOE once initialization is completed.</li> </ul> <p><b>IPv6:</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to permit traffic for specific source and destination IP addresses on TOE.</li> <li>• Keep sending traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being allowed.</li> </ul>

	<ul style="list-style-type: none"> <li>• Reboot the TOE while the ping is in progress.</li> <li>• Continue sending the required traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being denied through the TOE while initializing.</li> <li>• Continue sending the required traffic that matches the configured permit rule on the TOE.</li> <li>• Verify with packet capture that traffic is being allowed through the TOE once initialization is completed.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not allow any network traffic that is permitted by the ruleset to pass through while it is being initialized.</li> <li>• TOE logs and Packet Capture should show that permitted traffic is not passed through the TOE during TOE initialization.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. IPv4 and IPv6 packets that would otherwise be allowed by the ruleset are not passed through the TOE during initialization. This meets the testing requirements.

7.7.4 FFW\_RUL\_EXT.1.2/FFW\_RUL\_EXT.1.3/FFW\_RUL\_EXT.1.4 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>Test 1: The evaluator shall use the instructions in the guidance documentation to test that state full packet filter firewall rules can be created that permit, drop, and log packets for each of the following attributes:</p> <ul style="list-style-type: none"> <li>• ICMPv4 <ul style="list-style-type: none"> <li>○ Type</li> <li>○ Code</li> </ul> </li> <li>• ICMPv6 <ul style="list-style-type: none"> <li>○ Type</li> <li>○ Code</li> </ul> </li> <li>• IPv4 <ul style="list-style-type: none"> <li>○ Source address</li> <li>○ Destination Address</li> <li>○ Transport Layer Protocol</li> </ul> </li> <li>• IPv6 <ul style="list-style-type: none"> <li>○ Source address</li> <li>○ Destination Address</li> <li>○ Transport Layer Protocol and where defined by the ST author,</li> <li>○ Extension Header Type, Extension Header Fields</li> </ul> </li> <li>• TCP <ul style="list-style-type: none"> <li>○ Source Port</li> <li>○ Destination Port</li> </ul> </li> <li>• UDP <ul style="list-style-type: none"> <li>○ Source Port</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>○ Destination Port</li> </ul> <p>Note that these test activities should be performed in conjunction with those of FFW_RUL_EXT.1.9 where the effectiveness of the rules is tested. The test activities for FFW_RUL_EXT.1.9 define the protocol/attribute combinations required to be tested. If those combinations are configured manually, that will fulfil the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.</p>
<p><b>Test Steps</b></p>	<p><b><u>ICMPv4:</u></b></p> <p><b>Type</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to accept ICMPv4 type 8 code 0 packets but drop ICMPv4 type 3 code 0 packets.</li> <li>• Generate and send traffic that matches the applied filter.</li> <li>• Verify through logs that the ICMPV4 packets are dropped or accepted according to the applied rules based on type.</li> <li>• Verify the ICMPV4 packets are dropped or accepted according to the applied filter using Packet Capture.</li> </ul> <p><b>Code</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to accept ICMPv4 type 3 code 0 packets but drop ICMPv4 type 3 code 1 packets.</li> <li>• Generate and send traffic that matches the applied filter.</li> <li>• Verify through logs that the ICMPV4 packets are dropped or accepted according to the rules applied based on type.</li> <li>• Verify the ICMPV4 packets are dropped or accepted according to the filter applied using Packet Capture.</li> </ul> <p><b><u>IPv4:</u></b></p> <p><b>Source address</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to drop and accept traffic with specified IPv4 source addresses.</li> <li>• Generate and send traffic that matches the applied filter.</li> <li>• Verify the IPV4 packets are dropped or accepted according to the filter applied using logs.</li> <li>• Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture.</li> </ul>

### **Destination address**

- Configure a filter to drop and accept traffic with specified IPv4 destination addresses.
- Generate and send traffic that matches the applied filter.
- Verify the IPV4 packets are dropped or accepted according to the filter applied using logs.
- Verify the IPV4 packets are dropped or accepted according to the filter applied using Packet Capture.

### **Transport Layer Protocol**

- Configure a filter to drop and accept traffic with a specified IPv4 transport layer protocol.
- Generate and send traffic that matches the applied filter.
- Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using logs.
- Verify the IPV4 protocol packets are dropped or accepted according to the filter applied using Packet Capture.

### **ICMPV6:**

#### **Type**

- Configure a filter to accept and drop ICMPV6 packets according to its type.
- Generate and send traffic that matches the created filter.
- Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on type.
- Verify through packet capture that the ICMPV6 packets are dropped or accepted according to the rules applied based on type.

#### **Code**

- Configure a filter to accept and drop ICMPV6 packets according to its code.
- Generate and send traffic that matches the created filter.
- Verify through logs that the ICMPV6 packets are dropped or accepted according to the rules applied based on code.
- Verify through packet capture that the ICMPV6 packets are dropped or accepted according to the rules applied based on code.

### **IPv6:**

#### **Source address**

- Configure a filter to drop and accept traffic with specified IPv6 source addresses.
- Generate and send traffic that matches the applied filter.
- Verify the IPV6 packets are dropped or accepted according to the filter applied using logs.
- Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture.

#### **Destination address**

- Configure a filter to drop and accept traffic with specified IPv6 destination addresses.
- Generate and send traffic that matches the applied filter.
- Verify the IPV6 packets are dropped or accepted according to the filter applied using logs.
- Verify the IPV6 packets are dropped or accepted according to the filter applied using Packet Capture.

#### **Transport layer protocol**

- Configure a filter to drop and accept traffic with a specified IPv6 transport layer protocol.
- Generate and send traffic that matches the applied filter.
- Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using logs.
- Verify the IPV6 protocol packets are dropped or accepted according to the filter applied using Packet Capture.

#### **TCP:**

##### **Source Port**

- Configure a filter to drop and accept traffic according to specified source ports.
- Generate and send traffic that matches the applied filter.
- Verify the TCP packets are dropped or accepted according to the filter applied using logs.
- Verify the TCP packets are dropped or accepted according to the filter applied using packet capture.

##### **Destination Port**

- Configure a filter to drop and accept traffic according to specified destination ports.
- Generate and send traffic that matches the applied filter.

	<ul style="list-style-type: none"> <li>• Verify the TCP packets are dropped or accepted according to the filter applied using logs.</li> <li>• Verify the TCP packets are dropped or accepted according to the filter applied using packet capture.</li> </ul> <p><b>UDP:</b></p> <p><b>Source Port</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to drop and accept traffic according to specified source ports.</li> <li>• Generate and send traffic that matches the applied filter.</li> <li>• Verify the UDP packets are dropped or accepted according to the filter applied using logs.</li> <li>• Verify the UDP packets are dropped or accepted according to the filter applied using packet capture.</li> </ul> <p><b>Destination Port</b></p> <ul style="list-style-type: none"> <li>• Configure a filter to drop and accept traffic according to specified destination ports.</li> <li>• Generate and send traffic that matches the applied filter.</li> <li>• Verify the UDP packets are dropped or accepted according to the filter applied using logs.</li> <li>• Verify the UDP packets are dropped or accepted according to the filter applied using packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should allow successful creation of stateful packet filter firewall rules that can permit, drop, and log packets based on each specified attribute, including: <ul style="list-style-type: none"> <li>○ IPv4: Source address, Destination address, and Transport Layer Protocol</li> <li>○ IPv6: Source address, Destination address, Transport Layer Protocol</li> <li>○ TCP: Source Port and Destination Port</li> <li>○ UDP: Source Port and Destination Port</li> <li>○ ICMPv4: Type and Code</li> <li>○ ICMPv6: Type and Code</li> </ul> </li> <li>• Packet captures and TOE logs should show traffic getting accepted or dropped according to configured filter attributes.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. For ICMPv4, ICMPv6, TCP, UDP, IPv4 and IPv6, TOE successfully implemented full packet filter firewall rules that permit, drop, and log packets for each of the specified attributes. This meets the testing requirements.

7.7.5 FFW\_RUL\_EXT.1.2/FFW\_RUL\_EXT.1.3/FFW\_RUL\_EXT.1.4 TEST #2

Item	Data
------	------



<b>Test Assurance Activity</b>	<p>Test 2: Repeat the test assurance activity above to ensure that state full traffic filtering rules can be defined for each distinct network interface type supported by the TOE.</p> <p>Note that these test activities should be performed in conjunction with those of FFW_RUL_EXT.1.9 where the effectiveness of the rules is tested. The test activities for FFW_RUL_EXT.1.9 define the protocol/attribute combinations required to be tested. If those combinations are configured manually, that will fulfil the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. This test requirement has been covered as part of FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #1 (Only one interface type is claimed).</p>

7.7.6 FFW\_RUL\_EXT.1.5 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 1: The evaluator shall configure the TOE to permit and log TCP traffic. The evaluator shall initiate a TCP session. While the TCP session is being established, the evaluator shall introduce session establishment packets with incorrect flags to determine that the altered traffic is not accepted as part of the session (i.e., a log event is generated to show the ruleset was applied). After a TCP session is successfully established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and destination ports, sequence number, flags) one at a time in order to verify that the altered packets are not accepted as part of the established session.</p>
<b>Test Steps</b>	<p>IPv4:</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• The following script was used to send traffic for this test.</li> <li>• Verify that SYN packet was sent by the script first and a SYN-ACK packet was received.</li> <li>• Before the script sends an ACK packet to complete the TCP handshake, the script sends a few packets with flags that is not an ACK and verify that they are not passed through the firewall via PCAP and logs.</li> <li>• Verify that the script sends an ACK packet, and the TCP session is successfully established via packet capture.</li> <li>• Verify the logs generated for a successful connection and policy check.</li> <li>• Alter the connection by sending a modified source address.</li> </ul>

- Verify via packet capture that the packet is not accepted due to the modified source address.
- Verify via logs that the packet is not accepted due to the modified source address.
- Alter the connection by sending a modified destination address.
- Verify via packet capture that the packet is not accepted due to the modified destination address.
- Verify via logs that the packet is not accepted due to the modified destination address.
- Alter the connection by sending a modified source port.
- Verify via packet capture that the packet is not accepted due to the modified source port.
- Verify via logs that the packet is not accepted due to the modified source port.
- Alter the connection by sending a modified destination port.
- Verify via packet capture that the packet is not accepted due to the modified destination port.
- Verify via logs that the packet is not accepted due to the modified destination port.
- Alter the connection by sending a modified sequence number.
- Verify via packet capture that the packet is not accepted due to the modified sequence number.
- Verify via logs that the packet is not accepted due to the modified sequence number.
- Send a bad TCP flag which does not belong to the established TCP session.
- Verify via packet capture that the connection is not accepted due to TCP flag which is not a part of the session.
- Verify via logs that the packet is not accepted.

#### IPv6:

- Configure the TOE to allow all traffic.
- The following script was used to send traffic for this test.
- Run the script for establishing a successful TCP session.
- Verify with the logs generated for a successful TCP session
- Verify that the TCP session is successfully established via packet capture.
- If SYN+ACK is received send a new packet with a flag that is not part of 3-way handshake.
- After the TCP session is successful, send a new packet with a flag that is not an ACK and verify that they are not passed through the firewall via logs.
- After the TCP session is successful, send a new packet with a flag that is not an ACK and verify that they are not passed through the firewall via packet capture.
- Alter the connection by sending a modified source address.
- Verify via packet capture that the packet is not accepted due to the modified source address.

	<ul style="list-style-type: none"> <li>• Verify via logs that the packet is not accepted due to the modified source address.</li> <li>• Alter the connection by sending a modified destination address.</li> <li>• Verify via packet capture that the packet is not accepted due to the modified destination address.</li> <li>• Verify via logs that the packet is not accepted due to the modified destination address.</li> <li>• Alter the connection by sending a modified source port.</li> <li>• Verify via packet capture that the packet is not accepted due to the modified source port.</li> <li>• Verify via logs that the packet is not accepted due to the modified source port.</li> <li>• Alter the connection by sending a modified destination port.</li> <li>• Verify via packet capture that the packet is not accepted due to the modified destination port.</li> <li>• Verify via logs that the packet is not accepted due to the modified destination port.</li> <li>• Alter the connection by sending a modified sequence number.</li> <li>• Verify via packet capture that the packet is not accepted due to the modified sequence number.</li> <li>• Verify via logs that the packet is not accepted due to the modified sequence number.</li> <li>• Send a bad TCP flag which does not belong to the established TCP session.</li> <li>• Verify via packet capture that the connection is not accepted due to TCP flag which is not a part of the session.</li> <li>• Verify via logs that the packet is not accepted.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should successfully permit the initiation of TCP sessions. During the session establishment, any TCP packets with incorrect or unexpected flags should be rejected by the TOE.</li> <li>• After the TCP session is successfully established, any alterations to the session-determining attributes (e.g., source/destination addresses, ports, sequence number, flags) should result in the TOE rejecting the altered packets.</li> <li>• TOE logs should show traffic getting permitted according to configured filter attributes.</li> <li>• Packet Capture should show traffic getting permitted according to configured filter attributes.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For both IPv4 and IPv6, the TOE does not accept altered packets (source and destination addresses, source and destination ports, sequence number, flags) after a TCP session is successfully established. This meets testing requirements.</p>

7.7.7 FFW\_RUL\_EXT.1.5 TEST #2

Item	Data
------	------

<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 2: The evaluator shall terminate the TCP session established per Test 1 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.</p>
<b>Test Steps</b>	<p>IPv4</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a TCP session then terminate the session.</li> <li>• Send a packet that matches the former TCP session.</li> <li>• Verify that the logs show the TCP packet is similar to the former session and that it is subjected to the ruleset.</li> <li>• Verify via the packet capture that the TCP traffic sent matches the former TCP session.</li> </ul> <p>IPv6</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a TCP session then terminate the session.</li> <li>• Send a packet that matches the former TCP session and that it is subjected to the ruleset.</li> <li>• Verify that the logs show the TCP packet is similar to the former session.</li> <li>• Verify via the packet capture that the TCP traffic sent matches the former TCP session.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not forward any TCP packet matching the former session through the TOE without subjecting it to the ruleset.</li> <li>• TOE logs should show TCP traffic sent matching the former TCP session getting logged.</li> <li>• Packet Capture should show TCP traffic sent matches the former TCP session.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For both IPv4 and IPv6, any packet matching a former TCP session is not forwarded through the TOE without being subject to the ruleset, which meets the testing requirements.</p>

7.7.8 FFW\_RUL\_EXT.1.5 TEST #3

Item	Data
<b>Test Assurance Activity</b>	The following tests shall be run using IPv4 and IPv6.

	Test 3: The evaluator shall expire (i.e., reach timeout) the TCP session established per Test 1 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
<b>Test Steps</b>	<p>IPv4</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a TCP session then wait for the session to expire.</li> <li>• Send a packet that matches the former TCP session.</li> <li>• Verify that the logs show the TCP packet is similar to the former session and that it is subjected to the ruleset.</li> <li>• Verify via the packet capture that the TCP traffic sent matches the expired TCP session.</li> </ul> <p>IPv6</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a TCP session then wait for the session to expire.</li> <li>• Send a packet that matches the former TCP session.</li> <li>• Verify that the logs show the TCP packet is similar to the former session and that it is subjected to the ruleset.</li> <li>• Verify via the packet capture that the TCP traffic sent matches the expired TCP session.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not forward any TCP packet matching the former expired session through the TOE without subjecting it to the ruleset.</li> <li>• TOE logs should show TCP traffic sent matching expired TCP session getting logged and subjected to the configured ruleset.</li> <li>• Packet Capture should show TCP traffic sent matches the expired TCP session.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. For both IPv4 and IPv6, any TCP packet matching the former expired session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.

7.7.9 FFW\_RUL\_EXT.1.5 TEST #4

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 4: The evaluator shall configure the TOE to permit and log UDP traffic. The evaluator shall establish a UDP session. Once a UDP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, source and</p>

	<p>destination ports) one at a time in order to verify that the altered packets are not accepted as part of the established session.</p>
<p><b>Test Steps</b></p>	<p><b>IPv4:</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a UDP session and send 10 packets with the set of the original attributes.</li> <li>• Verify through the TOE logs that the UDP session is opened and matches the configured policy.</li> <li>• Verify through the packet capture that UDP packets are received at the other end.</li> <li>• Modify the source IP address in the crafted packet to a different value and send the altered packet to the same destination IP and port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified source address.</li> <li>• Modify the destination IP address in the crafted packet to a different value and send the altered packet with the same source IP and port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified destination address.</li> <li>• Modify the source port in the packet to a different value and send the altered packet to the original IP addresses and destination port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified source port.</li> <li>• Modify the destination port in the packet to a different value and send the altered packet to the original IP addresses and source port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified destination port.</li> </ul> <p><b>IPv6:</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a UDP session and send 10 packets with the set of the original attributes.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify through the TOE logs that the UDP session is opened and matches the configured policy.</li> <li>• Verify through the packet capture that UDP packets are received at the other end.</li> <li>• Modify the source IP address in the crafted packet to a different value and send the altered packet to the same destination IP and port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified source address.</li> <li>• Modify the destination IP address in the crafted packet to a different value and send the altered packet with the same source IP and port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified destination address.</li> <li>• Modify the source port in the packet to a different value and send the altered packet to the original IP addresses and destination port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified source port.</li> <li>• Modify the destination port in the packet to a different value and send the altered packet to the original IP addresses and source port.</li> <li>• Observe and verify that the TOE does not accept the altered packet as part of the established session and logs the packet as a new session.</li> <li>• Verify through packet capture that the UDP packet is sent with a modified destination port.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• After establishing the UDP session, the evaluator alters each of the session-determining attributes (source address, destination address, source port, destination port) one at a time. For each alteration, the TOE should not accept the connection as part of the original session..</li> <li>• TOE logs should show UDP traffic getting permitted according to configured filter attributes.</li> <li>• Packet Capture should show UDP traffic getting permitted according to configured filter attributes.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For both IPv4 and IPv6, the TOE does not accept altered packets (source and destination addresses, source and destination ports) after a UDP session is successfully established. This meets testing requirements.</p>

#### 7.7.10 FFW\_RUL\_EXT.1.5 TEST #5

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 5: The evaluator shall expire (i.e., reach timeout) the UDP session established per Test 4 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.</p>
<b>Test Steps</b>	<p>IPv4</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a UDP session then wait for the session to expire.</li> <li>• Send a packet that matches the former UDP session.</li> <li>• Verify that the logs show the UDP packet is similar to the former expired session.</li> <li>• Verify via the packet capture that the UDP traffic sent matches the expired UDP session.</li> </ul> <p>IPv6</p> <ul style="list-style-type: none"> <li>• Configure the TOE to allow all traffic.</li> <li>• Establish a UDP session then wait for the session to expire.</li> <li>• Send a packet that matches the former UDP session.</li> <li>• Verify that the logs show the UDP packet is similar to the former expired session.</li> <li>• Verify via the packet capture that the UDP traffic sent matches the expired UDP session.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should not forward any UDP packet matching the former expired session through the TOE without subjecting it to the ruleset.</li> <li>• TOE logs should show UDP traffic sent matching expired UDP session getting logged and subjected to the configured ruleset.</li> <li>• Packet Capture should show UDP traffic sent matches the expired UDP session.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For both IPv4 and IPv6, any UDP packet matching the former expired session is not forwarded through the TOE without being subject to the ruleset. This meets the testing requirements.</p>

#### 7.7.11 FFW\_RUL\_EXT.1.5 TEST #6

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p>



	Test 6: If ICMP is selected, the evaluator shall configure the TOE to permit and log ICMP traffic. The evaluator shall establish a session for ICMP as defined in the TSS. Once an ICMP session is established, the evaluator shall alter each of the session determining attributes (source and destination addresses, other attributes chosen in FFW_RUL_EXT.1.5) one at a time in order to verify that the altered packets are not accepted as part of the established session.
<b>Pass/Fail with Explanation</b>	N/A. ICMP is not selected in the ST.

#### 7.7.12 FFW\_RUL\_EXT.1.5 TEST #7

Item	Data
<b>Test Assurance Activity</b>	The following tests shall be run using IPv4 and IPv6.  Test 7: If applicable, the evaluator shall terminate the ICMP session established per Test 6 as described in the TSS. The evaluator shall then immediately send a packet matching the former session definition in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
<b>Pass/Fail with Explanation</b>	N/A. ICMP is not selected in the ST.

#### 7.7.13 FFW\_RUL\_EXT.1.5 TEST #8

Item	Data
<b>Test Assurance Activity</b>	The following tests shall be run using IPv4 and IPv6.  Test 8: The evaluator shall expire (i.e., reach timeout) the ICMP session established per Test 6 as described in the TSS. The evaluator shall then send a packet matching the former session in order to ensure it is not forwarded through the TOE without being subject to the ruleset.
<b>Pass/Fail with Explanation</b>	N/A. ICMP is not selected in the ST.

#### 7.7.14 FFW\_RUL\_EXT.1.6 TEST #1

Item	Data
------	------

<p><b>Test Assurance Activity</b></p>	<p>Both IPv4 and IPv6 shall be tested for items a), b), c), d), and e) of the SFR element. Both IPv4 and IPv6 shall be tested for item i) unless the rule definition is specific to IPv4 or IPv6. Note: f), g), and h) are specific to IPv4 or IPv6 and shall be tested accordingly</p> <p>Test 1: The evaluator shall test each of the conditions for automatic packet rejection in turn. In each case, the TOE should be configured to allow all network traffic and the evaluator shall generate a packet or packet fragment that is to be rejected. The evaluator shall use packet captures to ensure that the unallowable packet or packet fragment is not passed through the TOE.</p>
<p><b>Test Steps</b></p>	<p>IPv4</p> <p>Create a filter to allow all types of traffic through the TOE.</p> <ol style="list-style-type: none"> <li>1. Packets which are invalid fragments. <ul style="list-style-type: none"> <li>• By default, the TOE drops any packets that are invalid fragments when NDPP mode is enabled.</li> <li>• Send invalid fragments to the TOE.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> </li>   <li>2. Fragments that cannot be completely re-assembled. <ul style="list-style-type: none"> <li>• By default, the TOE drops fragments that cannot be completely re-assembled when NDPP mode is enabled.</li> <li>• Modify traffic to match the applied filter.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> </li>   <li>3. Packets where the source address is defined as being on a broadcast network. <ul style="list-style-type: none"> <li>• By default, the TOE drop packets where the source address is defined as being on a broadcast network when NDPP mode is enabled.</li> <li>• Send traffic where the source address is defined as being on a broadcast network.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> </li>   <li>4. Packets where the source address is defined as being on a multicast network. <ul style="list-style-type: none"> <li>• By default, the TOE drop packets where the source address is defined as being on a multicast network when NDPP mode is enabled.</li> <li>• Send traffic where the source address is defined as being on a multicast network.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> </li> </ol>

5. Packets where the source address is defined as being a loopback address.
  - Send traffic where the source address is defined as being on a loopback address.
  - Verify through logs that the traffic is rejected.
  - Verify through Packet Capture that the traffic is rejected.
  
6. Packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address “reserved for future use” (i.e. 240.0.0.0/4).
  - By default, the TOE drops packets where the source or destination address of the network packet is defined as an address “reserved for future use”.
  - Send traffic with a source address matching an unspecified address and reserved for further use.
  - Verify through logs that the traffic is rejected.
  - Verify through Packet Capture that the traffic is rejected.
  
7. Packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified.
  - Enable setting to log traffic with packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified.
  - Send traffic with IP options: Loose Source Routing, Strict Source Routing, or Record Route.
  - Verify through logs that the traffic is rejected.
  - Verify through Packet Capture that the traffic is rejected.

Other packets defined in FFW\_RUL\_EXT.1.6- No other rules defined

IPv6:

Create a filter to allow all types of traffic through the TOE.

1. Packets which are invalid fragments.
  - By default, the TOE drops any packets that are invalid fragments when NDPP mode is enabled.
  - Send invalid fragments to the TOE.
  - Verify through logs that the traffic is rejected.
  - Verify through Packet Capture that the traffic is rejected.
  
2. Fragments that cannot be completely re-assembled.
  - By default, the TOE drops fragments that cannot be completely re-assembled when NDPP mode is enabled.
  - Modify traffic to match the applied filter.
  - Verify through logs that the traffic is rejected.
  - Verify through Packet Capture that the traffic is rejected.

	<p>3. Packets where the source address is defined as being on a broadcast network.</p> <ul style="list-style-type: none"> <li>• By default, the TOE drop packets where the source address is defined as being on a broadcast network when NDPP mode is enabled.</li> <li>• Send traffic where the source address is defined as being on a broadcast network.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> <p>4. Packets where the source address is defined as being on a multicast network.</p> <ul style="list-style-type: none"> <li>• By default, the TOE drop packets where the source address is defined as being on a multicast network when NDPP mode is enabled.</li> <li>• Send traffic where the source address is defined as being on a multicast network.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> <p>5. Packets where the source address is defined as being a loopback address.</p> <ul style="list-style-type: none"> <li>• Send traffic where the source address is defined as being on a loopback address.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul> <p>6. Packets where the source or destination address of the network packet is defined as an “unspecified address” or an address “reserved for future definition and use” (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6.</p> <ul style="list-style-type: none"> <li>• By default, the TOE drops packets where the source or destination address of the network packet is defined as an address “reserved for future use”.</li> <li>• Send traffic with a source address matching unspecified address and reserved for further use.</li> <li>• Verify through logs that the traffic is rejected.</li> <li>• Verify through Packet Capture that the traffic is rejected.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should reject unallowable or fragmented packets.</li> <li>• TOE logs should show that unallowed packets or packet fragments are denied by it.</li> <li>• Packet Capture should show unallowed packets and packet fragments not passing through TOE.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For IPv4 and IPv6 each of the conditions (invalid fragment, reassembled fragments, broadcast network source address, multicast network source address, loopback address, unspecified or reserved address, and packets with IPv4 options) are rejected and logged automatically. This meets the testing requirements.</p>

#### 7.7.15 FFW\_RUL\_EXT.1.6 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>Both IPv4 and IPv6 shall be tested for items a), b), c), d), and e) of the SFR element. Both IPv4 and IPv6 shall be tested for item i) unless the rule definition is specific to IPv4 or IPv6. Note: f), g), and h) are specific to IPv4 or IPv6 and shall be tested accordingly</p> <p>Test 2: For each of the cases above, the evaluator shall use any applicable guidance to enable dropped packet logging or counting. In each case above, the evaluator shall ensure that the rejected packet or packet fragment was recorded (either logged or an appropriate counter incremented).</p>
<b>Pass/Fail with Explanation</b>	Pass. The requirements of this test have been completed as part of testing for FFW_RUL_EXT.1.6 Test #1.

#### 7.7.16 FFW\_RUL\_EXT.1.7 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 1: The evaluator shall configure the TOE to drop and log network traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received. The evaluator shall generate suitable network traffic to match the configured rule and verify that the traffic is dropped, and a log message generated.</p>
<b>Test Steps</b>	<p>When the NDPP mode is enabled, there is no need to configure any filter to drop or allow traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received.</p> <p>IPv4</p> <ul style="list-style-type: none"> <li>• Send traffic where the source address of the packet matches that of the TOE network interface.</li> <li>• Verify that traffic was denied via packet capture.</li> <li>• Verify that traffic was denied via TOE logs.</li> </ul> <p>IPv6</p> <ul style="list-style-type: none"> <li>• Send traffic where the source address of the packet matches that of the TOE network interface.</li> <li>• Verify that traffic was denied via packet capture.</li> <li>• Verify that traffic was denied via TOE logs.</li> </ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should drop traffic with a source address matching the network interface.</li> <li>• TOE logs should show traffic with a source address matching the TOE network interface getting dropped.</li> <li>• Packet Capture should show traffic with a source address matching the TOE network interface getting dropped.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. For both IPv4 and Ipv6, the TOE drops and logs network traffic where the source address of the packet matches that of the TOE network interface upon which the traffic was received. This meets testing requirements.

7.7.17 FFW\_RUL\_EXT.1.7 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The following tests shall be run using IPv4 and IPv6.</p> <p>Test 2: The evaluator shall configure the TOE to drop and log network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted, e.g. if the TOE believes that network 192.168.1.0/24 is reachable through interface 2, network traffic with a source address from the 192.168.1.0/24 network should be generated and sent to an interface other than interface 2. The evaluator shall verify that the network traffic is dropped, and a log message generated.</p>
<b>Test Steps</b>	<p>IPv4:</p> <ul style="list-style-type: none"> <li>• Ping from VM1 to the X3 interface of the TOE and verify that the network is reachable.</li> <li>• Verify through the logs that the packets are allowed and the network is reachable.</li> <li>• Verify through the packet capture that the ping response is received and network traffic is reachable.</li> <li>• Ping from VM1 to the X0 interface (other than interface X3) of the TOE and verify the packets are dropped.</li> <li>• Verify via logs that the packets are dropped to an interface other than interface X3.</li> <li>• Verify via packet capture that the packets are dropped to an interface other than interface X3.</li> </ul> <p>IPv6:</p> <ul style="list-style-type: none"> <li>• Ping from VM1 to the X3 interface of the TOE and verify that the network is reachable.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify through the logs that the packets are allowed and the network is reachable.</li> <li>• Verify through the packet capture that the ping response is received and network traffic is reachable.</li> <li>• Ping from VM1 to the X0 interface (other than interface X3) of the TOE and verify the packets are dropped.</li> <li>• Verify via logs that the packets are dropped to an interface other than interface X3.</li> <li>• Verify via packet capture that the packets are dropped to an interface other than interface X3.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should drop and log network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted.</li> <li>• TOE logs should show that traffic with a source address not matching the network reachability information of the targeted interface is dropped.</li> <li>• Packet Capture should show that traffic with a source address not matching the network reachability information of the interface to which it is targeted getting dropped.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. For both IPv4 and IPv6, the TOE drops and logs network traffic where the source IP address of the packet fails to match the network reachability information of the interface to which it is targeted. This meets the testing requirements.

**7.7.18 FFW\_RUL\_EXT.1.8 TEST #1 [TD0545]**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>Test 1: If the TOE implements a mechanism that ensures that no conflicting rules can be configured, the evaluator shall try to configure two conflicting rules and verify that the TOE rejects the conflicting rule(s). It is important to verify that the mechanism is implemented in the TOE but not in the non-TOE environment. If the TOE does not implement a mechanism that ensures that no conflicting rules can be configured, the evaluator shall devise two equal stateful traffic filtering rules with alternate operations – permit and drop. The rules should then be deployed in two distinct orders and in each case the evaluator shall ensure that the first rule is enforced in both cases by generating applicable packets and using packet capture and logs for confirmation.</p> <p><b>TD0545 has been applied.</b></p>
<b>Test Steps</b>	<p>IPv4</p> <ul style="list-style-type: none"> <li>• Configure a filter to allow destination address.</li> </ul>

	<ul style="list-style-type: none"> <li>Configure a filter to deny the same destination address and verify that TOE generates an error.</li> <li>Verify TOE generates configuration failure logs.</li> </ul> <p>IPv6</p> <ul style="list-style-type: none"> <li>Configure a filter to allow destination address.</li> <li>Configure a filter to deny the same destination address and verify that TOE generates an error.</li> <li>Verify TOE generates configuration failure logs.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>TOE should implement a mechanism that ensures that no conflicting rules can be configured.</li> <li>TOE logs showing configuration failure.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For both IPv4 and IPv6, TOE implements a mechanism that ensures that no conflicting rules can be configured, the evaluator tried to configure two conflicting rules and verified that the TOE rejects the conflicting rule(s). This meets the testing requirement.</p>

7.7.19 FFW\_RUL\_EXT.1.8 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>Test 2: The evaluator shall repeat the procedure above, except that the two rules should be devised where one is a subset of the other (e.g., a specific address vs. a network segment). Again, the evaluator should test both orders to ensure that the first is enforced regardless of the specificity of the rule.</p>
<b>Test Steps</b>	<p><b>IPv4</b></p> <ul style="list-style-type: none"> <li>Configure two firewall rules to allow packets with a destination network and deny packets with a destination address that is a subset of that network. Let the allow rule to be the first and the deny rule to be the second in order.</li> <li>Send traffic with the destination address specified in the deny rule.</li> <li>Verify through the firewall logs that the traffic with the destination address specified in the deny rule is allowed.</li> <li>Verify through a packet capture that the traffic with the destination address specified in the deny rule is allowed.</li> <li>Swap the order of the allow and deny rules.</li> <li>Send traffic with the destination address specified in the deny rule.</li> <li>Verify through the firewall logs that the traffic with the destination address specified in the deny rule is denied.</li> <li>Verify through a packet capture that the traffic with the destination address specified in the deny rule is denied.</li> </ul>



	<p><b>IPv6</b></p> <ul style="list-style-type: none"> <li>• Configure two firewall rules to allow packets with a destination network and deny packets with a destination address that is a subset of that network. Let the allow rule to be the first and the deny rule to be the second in order.</li> <li>• Send traffic with the destination address specified in the deny rule.</li> <li>• Verify through the firewall logs that the traffic with the destination address specified in the deny rule is allowed.</li> <li>• Verify through a packet capture that the traffic with the destination address specified in the deny rule is allowed.</li> <li>• Swap the order of the allow and deny rules.</li> <li>• Send traffic with the destination address specified in the deny rule.</li> <li>• Verify through the firewall logs that the traffic with the destination address specified in the deny rule is denied.</li> <li>• Verify through a packet capture that the traffic with the destination address specified in the deny rule is denied.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should enforce the first rule in the firewall filter.</li> <li>• TOE logs and packet capture should show that the packets are allowed or denied as per the configured rule.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. For both IPv4 and IPv6, the TOE enforces the first rule in the firewall filter. This meets the testing requirement.

7.7.20 FFW\_RUL\_EXT.1.9 TEST #1

Item	Data
<b>Test Assurance Activity</b>	For each attribute in FFW_RUL_EXT.1.2, the evaluator shall construct a test to demonstrate that the TOE can correctly compare the attribute from the packet header to the ruleset, and shall demonstrate both the permit and deny for each case. It shall also be verified that a packet is dropped if no matching rule can be identified for the packet. The evaluator shall check the log in each case to confirm that the relevant rule was applied. The evaluator shall record a packet capture for each test to demonstrate the correct TOE behaviour.
<b>Pass/Fail with Explanation</b>	Pass. This test has been completed as part of FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 TEST #1.

7.7.21 FFW\_RUL\_EXT.1.10 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The following tests shall be run using IPv4 and IPv6.

	<p>Test 1: The evaluator shall define a TCP half-open connection limit on the TOE. The evaluator shall generate TCP SYN requests to pass through the TOE to the target system using a randomised source IP address and common destination IP address. The number of SYN requests should exceed the TCP half-open threshold defined on the TOE. TCP SYN-ACK messages should not be acknowledged. The evaluator shall verify through packet capture that once the defined TCP half-open threshold has been reached, subsequent TCP SYN packets are not transmitted to the target system. The evaluator shall verify that when the configured threshold is reached that, depending upon the selection, either a log entry is generated or a counter is incremented.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Configure the TOE to limit the amount of half-open TCP connections.</li> </ul> <p>IPv4:</p> <ul style="list-style-type: none"> <li>Apply the configuration to the TOE's interface.</li> <li>Send traffic to the TOE from random source addresses.</li> <li>Verify that when the configured threshold is reached a log entry is generated or a counter is incremented.</li> <li>Verify via packet capture that the connection drops after reaching the threshold.</li> </ul> <p>IPv6:</p> <ul style="list-style-type: none"> <li>Configure the TOE to limit the amount of half-open TCP connections.</li> <li>Apply the configuration to the TOE's interface.</li> <li>Send traffic to the TOE from random source addresses.</li> <li>Verify that when the configured threshold is reached a log entry is generated or a counter is incremented.</li> <li>Verify via packet capture that the connection drops after reaching the threshold.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>Once the defined TCP half-open threshold is reached, the TOE should block subsequent TCP SYN packets from being transmitted.</li> <li>TOE should generate log entry and increments counter to show the half-open TCP connections after the configured threshold has been reached.</li> <li>Packet Capture should show that the TOE not responding to the half-open TCP connections after the configured threshold has been reached.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. For IPv4 and IPv6, randomized source TCP SYN packets are not transmitted by the TOE. When the configured threshold is reached, a counter is incremented by the TOE. This meets the testing requirement.</p>

7.7.22 FMT\_SMF.1/FFW TEST #1

Item	Data
------	------

<b>Test Assurance Activity</b>	The evaluation activities specified for FMT_SMF.1 in the Supporting Document for the Base-PP shall be applied in the same way to the newly added management functions defined in FMT_SMF.1/FFW in the FW Module.
<b>Pass/Fail with Explanation</b>	Pass. FMT_SMF.1 TEST #1 satisfies the requirement.

## 7.8 IPS

### 7.8.1 FAU\_GEN.1/IPS

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall test that the interfaces used to configure the IPS policies yield expected IPS data in association with the IPS policies. A number of IPS policy combination and ordering scenarios need to be configured and tested by attempting to pass both allowed and anomalous network traffic matching configured IPS policies in order to trigger all required IPS events.</p> <p>Note the following:</p> <ul style="list-style-type: none"> <li>This activity should have been addressed with a combination of the Test EAs for the other IPS requirements.</li> </ul> <p>As part of testing this activity, the evaluator shall also ensure that the audit data generated to address this SFR can be handled in the manner that FAU_STG_EXT.1 requires for all audit data.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Trigger each auditable event on the TOE. Verify that each audit record is generated and contains the required information.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The TOE should accurately generate audit records for all the required auditable events described in the ST under the FAU_GEN.1.1/IPS, FAU_GEN.1.2/IPS.</li> <li>The audit records generated should match the format specified in the guidance documentation.</li> <li>Evidence- Audit logs generated for each SFR.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. This test is covered under FAU_GEN.1 Test#1 from audit module.

### 7.8.2 FMT\_SMF.1/IPS TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following tests:

	<p>Test 1: The evaluator shall use the operational guidance to create a signature and enable it on an interface. The evaluator shall then generate traffic that would be successfully triggered by the signature. The evaluator should observe the TOE applying the corresponding reaction in the signature.</p> <p>Other testing for this SFR is performed in conjunction with the EAs for IPS_ABD_EXT.1 and IPS_SBD_EXT.1.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create a custom signature on the TOE to deny a specific type of traffic, such as a UDP packet with the payload 'acumen'.</li> <li>• Create an app rule to reset/drop the traffic for the custom signature created.</li> <li>• Using Scapy, create and send a UDP packet with the payload 'acumen,' which was defined as a custom signature in the TOE.</li> <li>• Verify via logs that the packets with the payload 'acumen' have been dropped.</li> <li>• Verify via packet capture that the payload 'acumen' was sent from LAN VM to WAN VM and that it has been dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should successfully apply the corresponding reaction in the signature.</li> <li>• TOE logs and packet capture should show that traffic matching the configured signature is dropped by TOE according to the applied policy.</li> <li>• TOE detects and logs traffic matching the configured signature and drops the traffic according to the applied policy.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The evaluator has used the operational guidance to create a signature and enable it on an interface. The evaluator has then generated traffic that successfully triggers the signature. The evaluator has observed the TOE applying the corresponding reaction to the signature. This meets testing requirements.</p>

7.8.3 FMT\_SMF.1/IPS TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall then disable the signature and attempt to regenerate the same traffic and ensure that the TOE allows the traffic to pass with no reaction.</p> <p>Other testing for this SFR is performed in conjunction with the EAs for IPS_ABD_EXT.1 and IPS_SBD_EXT.1.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Disable the signature from Test #1.</li> <li>• Send modified traffic that matches the disabled configured filter.</li> <li>• Verify through a packet capture that the traffic was appropriately allowed to pass through the TOE without the reaction specified in Test #1.</li> </ul>

<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should permit traffic matching the configured signature when the applied IDP policy is disabled.</li> <li>• TOE logs and packet capture should show traffic matching the configured signature being permitted through TOE when the applied IDP policy is disabled.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. After disabling the signature, the TOE allows the same traffic to pass through it with no reaction. This meets the testing requirements.

#### 7.8.4 FMT\_SMF.1/IPS TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 3: The evaluator shall use the operational guidance to import signatures and repeat the test conducted in Test 1.</p> <p>Other testing for this SFR is performed in conjunction with the EAs for IPS_ABD_EXT.1 and IPS_SBD_EXT.1.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Import a preconfigured signature and enable it to detect a specified attack. Load the customsignature.txt file from the system onto TOE.</li> <li>• Apply the signature to the TOE's interface by creating an app rule.</li> <li>• Create and send a packet with payload 'Acumensec' using Scapy.</li> <li>• Verify via logs that the packets with the payload 'Acumensec' have been dropped due to the created App rule.</li> <li>• Verify via packet capture that the payload 'Acumensec' was sent from LAN VM to WAN VM and that it has been dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should detect and log traffic matching the configured imported signature and drop the traffic according to the applied policy.</li> <li>• TOE logs and packet capture should show that traffic matching the configured imported signature is dropped by TOE according to the applied policy.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE imported signatures, which once enabled, detect traffic matching the signature on the configured interface. This meets the testing requirements.

#### 7.8.5 IPS\_ABD\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following tests:

	<p>Test 1: The evaluator shall use the instructions in the operational guidance to configure baselines or anomaly-based rules for each attributes specified in IPS_ABD_EXT.1.1. The evaluator shall send traffic that does not match the baseline or matches the anomaly-based rule and verify the TOE applies the configured reaction. This shall be performed for each attribute in IPS_ABD_EXT.1.1.</p>
<p><b>Test Steps</b></p>	<p><b><u>IPv4 – version</u></b></p> <p><b>Part1: Access rule without any Schedules Object</b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv4 version and define the data type as numeric with a value of 4.</li> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was allowed.</li> <li>• Verify through packet capture that response was received for ICMP packets.</li> </ul> <p><b>Part2: Access rule in scheduled time</b></p> <ul style="list-style-type: none"> <li>• Create a Schedule so any packet which is sent during this schedule is allowed.</li> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic and select the created schedule.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was allowed.</li> <li>• Verify through packet capture that response was received for ICMP packets.</li> </ul> <p><b>Part3: Access rule in outside scheduled time</b></p> <ul style="list-style-type: none"> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic and select the created schedule.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped because it was sent after the scheduled time.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <ul style="list-style-type: none"> <li>• Create another schedule from 11:00 to 17:00. Any packet that will be sent outside this time frame will get dropped.</li> </ul> <p><b><u>IPv6 – version.</u></b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv6 version and define the data type as numeric with a value of 6.</li> </ul>

- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 17:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **ICMPv4 – type.**

- Create a Packet Dissection object with the header field set to ICMP type and define the data type as numeric with a value of 8.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 17:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **ICMPv6 – type.**

- Create a Packet Dissection object with the header field set to ICMPv6 type and define the data type as numeric with a value of 129.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 17:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **TCP - Sequence Number.**

- Create a Packet Dissection object with the header field set to Sequence Number and define the data type as numeric with a value of 1234.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 17:00.
- Verify through packet capture that the TCP connection was reset.

	<p><b><u>UDP – length.</u></b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to packet length and define the data type as numeric with a value of 8.</li> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS_deny,” which was created earlier.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 17:00.</li> <li>• Verify through packet capture that no response was received for UDP packets.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should detect and log traffic matching the baselines or anomaly-based rules for particular attributes.</li> <li>• TOE logs and packet capture should show that traffic matching the baselines or anomaly-based rules for particular attributes are treated according to the applied policy.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The evaluator successfully used the instructions in the operational guidance to configure baselines or anomaly-based rules for each attribute specified in IPS_ABD_EXT.1.1. The evaluator then sent traffic that did not match the baseline or matched the anomaly-based rule. The TOE correctly applied the configured reaction for each attribute in IPS_ABD_EXT.1.1 This meets the testing requirements.</p>

7.8.6 IPS\_ABD\_EXT.1 TEST #2

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall repeat the test above to ensure that baselines or anomaly-based rules can be defined for each distinct network interface type supported by the TOE.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Before performing this test, a schedule was created in Object -&gt;Match Objects -&gt; Schedules from 01:00 to 07:00. So, any packet which was sent outside this time frame was dropped.</li> </ul> <p><b><u>IPv4 – version:</u></b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv4 version and define the data type as numeric with a value of 4.</li> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS_deny,” which was created earlier.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.</li> </ul>



- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **IPv6 – Version:**

- Create a Packet Dissection object with the header field set to IPv6 version and define the data type as numeric with a value of 6.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **ICMPv4 -type:**

- Create a Packet Dissection object with the header field set to ICMP type and define the data type as numeric with a value of 9.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **ICMPv6 – type:**

- Create a Packet Dissection object with the header field set to ICMPv6 type and define the data type as numeric with a value of 129.
- Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS\_deny,” which was created earlier.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.
- Verify through packet capture that no response was received for ICMP packets.

#### **TCP – Sequence number:**

- Create a Packet Dissection object with the header field set to Sequence Number and define the data type as numeric with a value of 3636.

	<ul style="list-style-type: none"> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS_deny,” which was created earlier.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.</li> <li>• Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.</li> <li>• Verify through packet capture that the TCP connection was reset.</li> </ul> <p><b><u>UDP – Length:</u></b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to packet length and define the data type as numeric with a value of 8.</li> <li>• Add a Packet Dissection object to the Firewall Access rule that allows traffic. Select the schedule object as “IPS_deny,” which was created earlier.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.</li> <li>• Verify through logs that the packet was dropped because it was sent after the scheduled time, which was set for 7:00.</li> <li>• Verify through packet capture that no response was received for UDP packets.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should detect and log traffic matching the baselines or anomaly-based rules for particular attribute.</li> <li>• TOE logs and packet capture should show that traffic matching the baselines or anomaly-based rules for particular attributes are treated according to applied policy.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The evaluator successfully used the instructions in the operational guidance to configure baselines or anomaly-based rules for each attribute specified in IPS_ABD_EXT.1.1. The evaluator then sent traffic that did not match the baseline or matched the anomaly-based rule. The TOE correctly applied the configured reaction for each attribute in IPS_ABD_EXT.1.1. This meets the testing requirement.</p>

7.8.7 IPS\_IPB\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: The evaluator shall use the instructions in the operational guidance to create a known-bad address list. Using a single IP address, a list of addresses or a range of addresses from that list, the evaluator shall attempt to send traffic through the TOE that would otherwise be allowed by the TOE and observe the TOE automatically drops that traffic.</p>

<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Before creating the access rule, the evaluator can ping the known-bad addresses through the TOE.</li> <li>• Verify via TOE logs that ICMP packets with a known-bad addresses were allowed.</li> <li>• Create a single entry of a known-bad address.</li> <li>• Create an access rule to block this created known-bad address.</li> <li>• Verify logs generated for the creation of an access rule.</li> <li>• Ping the IP addresses and verify the ICMP packets to the specified known-bad address have been denied, while ICMP packets to other addresses have been allowed.</li> <li>• Verify through TOE logs that only the ICMP packets to the specified known-bad address have been denied, while ICMP packets to other addresses have been allowed.</li> <li>• Verify via packet capture that only the ICMP packets to the specified known-bad address have been denied, while ICMP packets to other addresses have been allowed.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should drop traffic matching the configured know bad address list, which would otherwise be allowed.</li> <li>• TOE logs should show that traffic matching the single IP address, a list of addresses or a range of addresses in the known-bad address list are dropped by TOE.</li> <li>• Packet Capture should show traffic matching the single IP address, a list of addresses or a range of addresses in the known-bad address list are dropped by TOE.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE drops traffic matching the configured know bad address list, which would otherwise be allowed. This meets the testing requirements.

7.8.8 IPS\_IPB\_EXT.1 TEST #2

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall use the instructions in the operational guidance to create a known-good address list. Using a single IP address, a list of addresses or a range of addresses from that list, the evaluator shall attempt to send traffic that would otherwise be denied by the TOE and observe the TOE automatically allowing traffic.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Before creating the access rule, the evaluator can ping the known-good address through the TOE.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify via TOE logs that ICMP packets with known-good addresses were allowed.</li> <li>• Create a single entry of a known-good address.</li> <li>• Create an access rule to allow only the created known-good address.</li> <li>• Verify logs generated for the creation of an access rule.</li> <li>• Ping the IP addresses and verify that ICMP packets to the specified known-good address have been permitted, while ICMP packets to other addresses have been denied.</li> <li>• Verify through TOE logs that ICMP packets to the specified known-good address have been permitted, while ICMP packets to other addresses have been denied.</li> <li>• Verify via packet capture that ICMP packets to the specified known-good address have been permitted, while ICMP packets to other addresses have been denied.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should permit traffic matching the configured know-good address and address range.</li> <li>• TOE logs show that after the implementation of a known-good address list onto a policy, traffic matching the single IP address, a list of addresses or a range of addresses in the address list is permitted by TOE, while ICMP packets to other addresses have been denied.</li> <li>• Packet Capture shows that after implementation of a known-good address list onto a policy, traffic matching the single IP address, a list of addresses or a range of addresses in the address list is permitted by TOE.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE permits traffic matching the configured know-good address and address range. This meets the testing requirements.

7.8.9 IPS\_IPB\_EXT.1 TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 3: The evaluator shall add conflicting IP addresses to each list and ensure that the TOE handles conflicting traffic in a manner consistent with the precedence in IPS_NTA_EXT.1.1.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create two conflicting access rules on the TOE.</li> <li>• Ping from the LAN VM to the WAN VM and verify that the packets are dropped.</li> <li>• Verify through the logs that the packets are dropped due to the configured access rule.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> <li>• Change the priority of access rules. Set the allow all traffic rule as the highest priority and set the block ICMP requests rule as priority 2.</li> </ul>

	<ul style="list-style-type: none"> <li>• Ping from the LAN VM to the WAN VM and verify that the ICMP packets were allowed</li> <li>• Verify through the logs that the packets are allowed as access rule priority was changed.</li> <li>• Verify through packet capture that a response was received for ICMP packets.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should handle conflicting rules in an Administrator-defined order.</li> <li>• TOE logs should show that it handles conflicting traffic according to the order in which rules are applied.</li> <li>• Packet Capture should show that it handles conflicting traffic according to the order in which rules are applied.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE handles conflicting rules in an Administrator-defined order. This meets the testing requirements.

#### 7.8.10 IPS\_NTA\_EXT.1.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There are no test EAs for this element.
<b>Pass/Fail with Explanation</b>	There are no test EAs for this element.

#### 7.8.11 IPS\_NTA\_EXT.1.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There are no test EAs for this element.
<b>Pass/Fail with Explanation</b>	There are no test EAs for this element.

#### 7.8.12 IPS\_NTA\_EXT.1.3 TEST #1

Item	Data
<b>Test Assurance Activity</b>	Testing for this element is performed in conjunction with testing where promiscuous and inline interfaces are tested.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by IPS_SBD_EXT.1 Test#1.

7.8.13 IPS\_SBD\_EXT.1.1 TEST #1 [TD0722]

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: The evaluator shall use the instructions in the operational guidance to test that packet header signatures can be created and/or configured with the selected and/or configured reactions specified in IPS_SBD_EXT.1.5 for each of the attributes listed below. Each attribute shall be individually assigned to its own unique signature:</p> <ul style="list-style-type: none"> <li>• IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; IP Options; and, if selected, type of services (ToS).</li> <li>• IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and, if selected, traffic class and/or flow label.</li> <li>• ICMP: Type; Code; Header Checksum; and, if selected, other Header fields (varies based on the ICMP type and code).</li> <li>• ICMPv6: Type; Code; and Header Checksum.</li> <li>• TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.</li> <li>• UDP: Source port; destination port; length; and UDP checksum.</li> </ul> <p>The evaluator shall generate traffic to trigger a signature and shall then use a packet sniffer to capture traffic that ensures the reactions of each rule are performed as expected.</p> <p><b>TD0722 has been applied.</b></p>
<p><b>Test Steps</b></p>	<p><b>IPv4</b></p> <p><b>Version:</b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv4 version and define the data type as numeric with a value of 4.</li> <li>• Add Packet Dissection Filter to Firewall Access rule that allows traffic.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped due to the created Packet Dissection object.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul>

**Header Length:**

- Create a Packet Dissection object with the header field set to Header Length and define the data type as numeric with a value of 20.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Packet Length:**

- Create a Packet Dissection object with the header field set to Packet length and define the data type as numeric with a value of 128.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**ID:**

- Create a Packet Dissection object with the header field set to Identity and define the data type as numeric with a value of 4.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**IP Flag (DF):**

- Create a Packet Dissection object with the header field set to Flag and define the data type as Bitset with a value Don't Fragment.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**IP Flag (MF):**

- Create a Packet Dissection object with the header field set to Flag and define the data type as Bitset with a value More Fragments.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Fragment Offset:**

- Create a Packet Dissection object with the header field set to Fragment Offset and define the data type as numeric with a value of 1480.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Time to Live (TTL):**

- Create a Packet Dissection object with the header field set to TTL and define the data type as numeric with a value of 64.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Send modified traffic to match the packet dissection rule. Send the packets from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Protocol:**

- Create a Packet Dissection object with the header field set to Protocol and define the data type as numeric with a value of 1.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Header Checksum:**

- Create a Packet Dissection object with the header field set to Header Checksum and define the data type as Numeric (Hex) with a value of 5e49.



- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Source Address:**

- Create an Address Object for the source address to be blocked.
- Select the created address object in the access rule.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created policy.
- Verify through packet capture that no response was received for ICMP packets.

**Destination Address:**

- Create an Address Object for the destination address to be blocked.
- Select the created address object in the access rule.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created policy.
- Verify through packet capture that no response was received for ICMP packets.

**IP Option:**

- Create a Packet Dissection object with the header field set to Option Type(8 bits) and define the data type as numeric with a value 1.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**IPv6**

**IPv6 Version:**

- Create a Packet Dissection object with the header field set to IPv6 Version and define the data type as numeric with a value 6.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.

- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **IPv6 Payload Length:**

- Create a Packet Dissection object with the header field set to IPv6 Payload Length and define the data type as numeric with a value of 20.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **IPv6 Next Header:**

- Create a Packet Dissection object with the header field set to IPv6 Next Header and define the data type as numeric with a value 6.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **IPv6 Hop Limit:**

- Create a Packet Dissection object with the header field set to IPv6 Hop Limit and define the data type as numeric with a value of 60.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **IPv6 Source Address:**

- Create an Address Object for the IPV6 source address to be blocked.
- Create an access rule to deny any service from the IPv6 source address selected in an object.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.

- Verify through logs that the packet was dropped due to the created policy.
- Verify through packet capture that a TCP session was reset.

#### **IPv6 Destination Address:**

- Create an Address Object for the IPV6 destination address to be blocked.
- Create an access rule to deny any service from the IPv6 destination address selected in an object.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created policy.
- Verify through packet capture that a TCP session was reset.

#### **IPv6 Routing Header:**

##### **Type:**

- Create a Packet Dissection object with the header field set to Routing Header Type and define the data type as numeric with a value 2.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was captured for malformed packets.

##### **Segment Left:**

- Create a Packet Dissection object with the header field set to IPv6 Routing Segments left and define the data type as numeric with a value 1.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was captured for malformed packets.

##### **Routing Header – Length:**

- Create a Packet Dissection object with the header field set to IPv6 Routing Header Length and define the data type as numeric with a value 0.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.

- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was captured for malformed packets.

#### **IPv6 - Traffic Class:**

- Create a Packet Dissection object with the header field set to IPv6 Traffic Class and define the data type as numeric with a value 2.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **Flow Label:**

- Create a Packet Dissection object with the header field set to IPv6 Flow Label and define the data type as numeric with a value 2.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **For ICMP(v4)**

##### **Type:**

- Create a Packet Dissection object with the header field set to ICMP Type and define the data type as numeric with a value 9.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

##### **ICMP(v4) – Code:**

- Create a Packet Dissection object with the header field set to ICMP code and define the data type as numeric with a value 100.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**ICMPv4- Header Checksum:**

- Create a Packet Dissection object with the header field set to ICMP Checksum and define the data type as numeric with a value 5e49.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**ICMPv4 – Rest of Header:**

- Create a Packet Dissection object with the header field set to ICMP identifier and define the data type as numeric with a value 1234.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Type:**

- Create a Packet Dissection object with the header field set to ICMPv6 Type and define the data type as numeric with a value of 129.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Code:**

- Create a Packet Dissection object with the header field set to ICMPv6 code and define the data type as numeric with a value 0.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**Header Checksum:**

- Create a Packet Dissection object with the header field set to ICMPv6 Checksum and define the data type as numeric with a value 1234.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

**For TCP**

**Source port:**

- Create a Service Object for the TCP source port.
- Create an access rule to deny the created TCP source port service object.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created access rule.
- Verify through packet capture that a TCP session was reset.

**TCP Destination Port:**

- Create a Service Object for the TCP destination port.
- Create an access rule to deny the created TCP destination port service object.
- Craft a packet using the Scapy tool to match the configured access rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created access rule.
- Verify through packet capture that a TCP session was reset.

**TCP- Sequence Number:**

- Create a Packet Dissection object with the header field set to Sequence Number and define the data type as numeric with a value 1234.

- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

**TCP- Offset:**

- Create a Packet Dissection object with the header field set to Data Offset and define the data type as numeric with a value 5.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

**TCP SYN Flag:**

- Create a Packet Dissection object with the header field set to Flag and define the data type as Bitset with a value SYN.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

**TCP – Window:**

- Create a Packet Dissection object with the header field set to Window and define the data type as numeric with a value 256.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

**TCP – Checksum:**

- Create a Packet Dissection object with the header field set to TCP Checksum and define the data type as numeric (Hex) with a value 91e7.

- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **TCP- Urgent Pointer:**

- Create a Packet Dissection object with the header field set to Urgent Pointer and define the data type as Range with a range 0 to 10.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **TCP -options:**

- Create a Packet Dissection object with the header field set to Options -- MSS and define the data type as Range with a range 512 to 2000.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **TCP – Acknowledgement Number:**

- Create a Packet Dissection object with the header field set to Ack Number and define the data type as Range with a range 0 to 10.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **TCP – Reserved:**



- Create a Packet Dissection object with the header field set to Reserved and define the data type as Range with a range 0 to 255.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **For UDP**

##### **UDP Source Port:**

- Create a Service Object for the UDP source port.
- Create an access rule to deny the created UDP source port service object.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created access rule.
- Verify through packet capture that UDP packets were dropped.

##### **UDP Destination Port:**

- Create a Service Object for the UDP destination port.
- Create an access rule to deny the created UDP destination port service object.
- Craft a packet using the Scapy tool to match the configured filter. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created access rule.
- Verify through packet capture that UDP packets were dropped.

##### **UDP Length:**

- Create a Packet Dissection object with the header field set to Packet Length and define the data type as Numeric with a value 8.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that UDP packets were dropped.

##### **UDP Checksum:**

- Create a Packet Dissection object with the header field set to UDP Checksum and define the data type as Numeric (Hex) with a value e2dc.

	<ul style="list-style-type: none"> <li>• Add Packet Dissection Filter to Firewall Access rule that allows traffic.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped due to the created Packet Dissection object.</li> <li>• Verify through packet capture that UDP packets were dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When TOE is triggered with traffic matching configured signatures, it should react in the expected way by dropping the traffic.</li> <li>• TOE logs and packet capture should the traffic is dropped due to configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE is triggered with traffic matching configured signatures and reacts in the expected way by dropping the traffic. This meets the testing requirements.

7.8.14 IPS\_SBD\_EXT.1.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall repeat the test above to ensure that signature-based IPS policies can be defined for each distinct network interface type capable of applying signatures as supported by the TOE.</p>
<b>Test Steps</b>	<p><b>IPv4 - Version</b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv4 version and define the data type as numeric with a value of 4.</li> <li>• Add Packet Dissection Filter to Firewall Access rule that allows traffic.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.</li> <li>• Verify through logs that the packet was dropped due to the created Packet Dissection object.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <p><b>IPv6- Version</b></p> <ul style="list-style-type: none"> <li>• Create a Packet Dissection object with the header field set to IPv6 Version and define the data type as numeric with a value 6.</li> <li>• Add Packet Dissection Filter to Firewall Access rule that allows traffic.</li> <li>• Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.</li> </ul>

- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **ICMPv4 - type**

- Create a Packet Dissection object with the header field set to ICMP Type and define the data type as numeric with a value 9.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

#### **ICMPv6 - type**

- Create a Packet Dissection object with the header field set to ICMPv6 Type and define the data type as numeric with a value of 129.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that no response was received for ICMP packets.

#### **TCP - window**

- Create a Packet Dissection object with the header field set to Window and define the data type as numeric with a value 256.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.
- Verify through packet capture that a TCP session was not established.

#### **UDP – Packet Length**

- Create a Packet Dissection object with the header field set to Packet Length and define the data type as Numeric with a value 8.
- Add Packet Dissection Filter to Firewall Access rule that allows traffic.
- Craft a packet using the Scapy tool to match the packet dissection rule. Send the crafted packet from the WAN VM to the LAN VM.
- Verify through logs that the packet was dropped due to the created Packet Dissection object.

	<ul style="list-style-type: none"> <li>• Verify through packet capture that UDP packets were dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When TOE is triggered with traffic matching configured signatures, it should react in the expected way by dropping the traffic.</li> <li>• TOE logs and packet capture should show that the traffic is dropped due to configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE is triggered with traffic matching configured signatures and reacts in the expected way by dropping the traffic. This meets the testing requirements.

7.8.15 IPS\_SBD\_EXT.1.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 1: The evaluator shall use the instructions in the operational guidance to test that packet payload string-based detection rules can be assigned to the reactions specified in IPS_SBD_EXT.1.5 using the attributes specified in IPS_SBD_EXT.1.2. However it is not required (nor is it feasible) to test all possible strings of protocol data, the evaluator shall ensure that a selection of strings in the requirement is selected to be tested. At a minimum at least one string using each of the following attributes from IPS_SBD_EXT.1.2 should be tested for each protocol. The evaluator shall generate packets that match the string in the rule and observe the corresponding reaction is as configured.</p> <ul style="list-style-type: none"> <li>• Test at least one string of characters for ICMPv4 data: beyond the first 4 bytes of the ICMP header.</li> <li>• Test at least one string of characters for ICMPv6 data: beyond the first 4 bytes of the ICMP header.</li> <li>• TCP data (characters beyond the 20 byte TCP header): <ul style="list-style-type: none"> <li>i) Test at least one FTP (file transfer) command: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.</li> <li>ii) HTTP (web) commands and content: <ul style="list-style-type: none"> <li>(1) Test both GET and POST commands</li> <li>(2) Test at least one administrator-defined strings to match URLs/URIs, and web page content.</li> </ul> </li> <li>iii) Test at least one SMTP (email) state: start state, SMTP commands state, mail header state, mail body state, abort state.</li> <li>iv) Test at least one string in any additional attribute type defined within the “other types of TCP payload inspection” assignment, if any other types are specified.</li> </ul> </li> <li>• Test at least one string of UDP data: characters beyond the first 8 bytes of the UDP header;</li> </ul>

	<ul style="list-style-type: none"> <li>• Test at least one string for each additional attribute type defined in the “other types of packet payload inspection” assignment, if any other types are specified.</li> </ul>
<b>Test Steps</b>	<p><b>ICMPv4</b></p> <ul style="list-style-type: none"> <li>• Create a custom Object with the Match Type as Exact Match and the content as “abcdefghij123”.</li> <li>• Create an App rule to scan for ICMPv4 string and select the created object. Select the destination service as Ping 8.</li> <li>• The rule will automatically be enabled.</li> <li>• Craft a packet using the Scapy tool to match the created object string and send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <p><b>ICMPv6</b></p> <ul style="list-style-type: none"> <li>• Create a custom Object with the Match Type as Exact Match and the content as “abcdefghij123”.</li> <li>• Create an App rule to scan for ICMP6 string and select the created object. Select the destination service as Ping6 128.</li> <li>• The rule will be automatically enabled after it is created.</li> <li>• Craft a packet using the Scapy tool to match the created object string and send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <p><b>FTP</b></p> <ul style="list-style-type: none"> <li>• Create a user named 'test' on the WAN VM and install FileZilla on the LAN VM. Establish a successful FTP connection.</li> <li>• Create an Object with the Match Object Type as FTP command and the content as HELP and SIZE.</li> <li>• Create an App rule to scan for the FTP client and select the created object. Select the destination service as FTP control.</li> <li>• The rule will be automatically enabled once it is created.</li> <li>• Send HELP as the custom command.</li> <li>• Verify that the FTP connection is reset by the peer when the HELP command is sent.</li> <li>• Send SIZE as the custom command.</li> <li>• Verify that the FTP connection is reset by the peer when the SIZE command is sent.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that the connection was reset for the HELP and SIZE commands.</li> </ul>

### **HTTP (web) Commands:**

#### **HTTP(GET):**

- Create an Object to match HTTP GET requests.
- Create an App rule to scan for the FTP client and select the created object. Select the destination service as FTP control.
- Create an App rule and select the created object. Select the destination service as HTTP.
- The rule will be automatically enabled.
- Craft a packet using the Scapy tool to match the created object and send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the applied rule.
- Verify through packet capture that the connection was reset.

#### **HTTP POST:**

- Create an Object to match HTTP POST requests.
- Create an App rule and select the created object. Select the destination service as HTTP.
- The rule will be automatically enabled after it is created.
- Craft a packet using the Scapy tool to match the created object and send the crafted packet from the LAN VM to the WAN VM.
- Verify through logs that the packet was dropped due to the applied rule.
- Verify through packet capture that the connection was reset.

#### **HTTP URL:**

- From the LAN host connect to the web page with an HTTP post link and verify that the web page is displayed.
- Create an Object to Match URL value.
- Create an App rule and select the created object. Select the destination service as HTTP.
- The rule will automatically be enabled.
- Craft traffic using the Scapy tool to match the created object and send the crafted packet from the LAN VM to the WAN VM.
- From the LAN Host attempt to access the web page and verify it is blocked.
- Verify through logs that the packet was dropped due to the applied rule.
- Verify through packet capture that the connection was reset.

#### **HTTP Web Page Content:**

- From the LAN host connect to the web page with an HTTP link and the web page will be displayed. Verify that the “test.zip” can be accessed from the web page.
- Create Custom Match Object with web page content “test.zip”.

	<ul style="list-style-type: none"> <li>• Create an App rule and select the created object.</li> <li>• The rule will be automatically enabled.</li> <li>• Attempt to access the website with the blocked content 'test.zip' and verify that the request fails.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that the connection was reset.</li> </ul> <p><b>SMTP:</b></p> <ul style="list-style-type: none"> <li>• Test the SMTP start state by using the EHLO command.</li> <li>• Create a Match Object for outbound SMTP requests (HELO or EHLO).</li> <li>• Create an App rule and select the created object.</li> <li>• The rule will be automatically enabled.</li> <li>• Send the SMTP traffic from the LAN VM to the WAN VM and verify that it fails.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that the connection was reset.</li> </ul> <p><b>UDP:</b></p> <ul style="list-style-type: none"> <li>• Create a custom Object with the Match Type as Exact Match and the content as "ABCDEFGHJIJ123".</li> <li>• Create an App rule and select the created object.</li> <li>• The rule will be automatically enabled after it is created.</li> <li>• Craft a packet using the Scapy tool to match the created object and send the crafted packet from the LAN VM to the WAN VM.</li> <li>• Verify through logs that the packet was dropped due to the applied rule.</li> <li>• Verify through packet capture that no response was received for UDP packets.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When TOE is triggered with traffic matching configured signatures, it should react in the expected way by dropping the traffic.</li> <li>• TOE logs and packet capture should show that the traffic is dropped due to configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE is triggered with traffic matching configured signatures and reacts in the expected way by dropping the traffic. This meets the testing requirements.

7.8.16 IPS\_SBD\_EXT.1.2 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test 2: The evaluator shall repeat Test 1 above to ensure that signature-based IPS policies can be defined for each distinct network interface type capable of applying signatures as supported by the TOE.</p>

<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Create an Object to match the string “abcdefghij123” in the packet.</li> <li>• Create an App rule to scan for ICMPv4 string and select the created object.</li> <li>• Craft and send an ICMPv4 packet using the Scapy tool with the string 'abcdefghij123' to be matched.</li> <li>• Verify through logs that the packets are dropped due to the created APP rule.</li> <li>• Verify through the packet capture that no response was received for ICMP packets.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• When TOE is triggered with traffic matching configured object, it should react in the expected way by dropping the traffic.</li> <li>• TOE logs and packet capture should show that the traffic is dropped due to configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE is triggered with traffic matching configured signatures and reacts in the expected way by dropping the traffic. This meets the testing requirements.

7.8.17 IPS\_SBD\_EXT.1.3 TEST #1

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	The evaluator shall create and/or configure rules for each attack signature in IPS_SBD_EXT.1.3. For each attack, the TOE should apply its corresponding signature and enable it to each distinct network interface type capable of applying the signatures. The evaluator shall use packet captures to ensure that the attack traffic is detected by the TOE and a reaction specified in IPS_SBD_EXT.1.5 is triggered and stops the attack. Each attack should be performed one after another so as to ensure that its corresponding signature successfully identified and appropriately reacted to a particular attack.
<b>Test Steps</b>	<p><b>a) IP Attacks:</b></p> <p>i. IP fragment Overlap: Teardrop attack</p> <ul style="list-style-type: none"> <li>• Send the Teardrop attack packets using the Python script to the TOE.</li> <li>• Verify through the TOE logs that packets are dropped.</li> <li>• Verify through the packet capture that packets have been dropped.</li> </ul> <p>ii. IP source address equal to the IP destination address (Land attack):</p> <ul style="list-style-type: none"> <li>• Craft and send a packet using the Scapy tool with the source address equal to the destination address.</li> <li>• Verify through the TOE logs that packets are dropped due to the land attack.</li> <li>• Verify through the packet capture that no response was found for ICMP packets.</li> </ul> <p><b>b) ICMP Attacks:</b></p> <p>i. Fragmented ICMP Traffic:</p> <ul style="list-style-type: none"> <li>• Craft a packet using the Scapy tool to send fragmented ICMP packets from the LAN VM to the WAN VM.</li> </ul>



- Verify through the packet capture that packets have been dropped due to an attack.
  - Verify through the packet capture that packets have been dropped.
- ii. Large ICMP Traffic (Ping of Death attack):
- Craft packets using the Scapy tool to send large ICMP packets to the TOE.
  - Verify through the packet capture that packets have been dropped due to an attack.
  - Verify through the packet capture that packets have been dropped.
- c) **TCP Attacks:**
- i. TCP NULL flags:
- Craft a packet using the Scapy tool to send a TCP packet with the NULL flag set.
  - Verify through the packet capture that packets have been dropped due to the TCP NULL flag.
  - Verify through the packet capture that packets have been dropped.
- ii. TCP SYN+FIN flags:
- Craft a packet using the Scapy tool to send a TCP packet with the SYN and FIN flags set.
  - Verify through the TOE logs that packets are dropped due to TCP SYN/FIN.
  - Verify through the packet capture that packets have been dropped.
- iii. TCP FIN only flags:
- Craft a packet using the Scapy tool to send a TCP packet with the FIN flag set.
  - Verify through the TOE logs that packets are dropped.
  - Verify through the packet capture that packets have been dropped.
- iv. TCP SYN+RST flags:
- Craft a packet using the Scapy tool to send a TCP packet with the SYN and RST flags set.
  - Verify through the TOE logs that packets are dropped.
  - Verify through the packet capture that packets have been dropped.
- d) **UDP Attacks:**
- i. UDP Chargen DoS Attack:
- Create a service object for UDP port 19 to represent a Chargen packet.
  - Select the created service object in the access rule and set the action to 'Drop'.
  - Craft a packet using the Scapy tool to match the created object, and send the crafted packet from the LAN VM to the WAN VM.
  - Verify through the TOE logs that packets are dropped.
  - Verify through the packet capture that packets have been dropped.
- ii. UDP Bomb Attack:
- Create and send UDP packets with a UDP length that is less than the IP length.

	<ul style="list-style-type: none"> <li>• Verify through the TOE logs that packets are dropped.</li> <li>• Verify through the packet capture that packets have been dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should detect each attack based on its corresponding signature and trigger the specified reaction as outlined in IPS_SBD_EXT.1.5, successfully stopping the attack.</li> <li>• TOE logs and packet capture should show that the attack is detected and traffic is dropped.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE successfully identifies each attack using the corresponding signature and triggers the specified reaction to stop the attack. This meets the testing requirement.

7.8.18 IPS\_SBD\_EXT.1.4 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall configure individual signatures for each attack in IPS_SBD_EXT.1.4. For each attack, the TOE should apply its corresponding signature and enable it to each distinct network interface type capable of applying signatures. The evaluator shall use packet captures to ensure that the attack traffic is detected by the TOE and a reaction specified in IPS_SBD_EXT.1.5 is triggered and stops the attack. Each attack should be performed one after another so as to ensure that its corresponding signature successfully identified and appropriately reacted to a particular attack.
<b>Test Steps</b>	<p><b>a) <u>Flooding a Host (DOS attack):</u></b></p> <p><b>i. <u>ICMP flooding (Smurf attack):</u></b></p> <p>The Smurf attack is a distributed denial-of-service attack in which large numbers of Internet Control Message Protocol (ICMP) packets with the intended victim's spoofed source IP are broadcast to a computer network using an IP broadcast address.</p> <ul style="list-style-type: none"> <li>• Using the hping3 tool, send a large number of ICMP packets with a spoofed source IP address, using the broadcast IP address.</li> <li>• Verify through the logs that the TOE detects the attack and that the traffic is dropped.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <p><b>ii. <u>TCP flooding (SYN flood):</u></b></p> <ul style="list-style-type: none"> <li>• Verify that the SYN flood protection mode is enabled on the TOE.</li> <li>• Using the hping3 tool send a large number of TCP packets to port 1001 on the target IP address.</li> <li>• Verify through the logs that the TOE detects the attack and that the traffic is dropped.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify through packet capture that a large number of TCP SYN packets were sent to the TOE.</li> </ul> <p><b>b) <u>Flooding a network (DoS attack):</u></b></p> <ul style="list-style-type: none"> <li>• Verify that the ICMP flood protection is enabled on the TOE.</li> <li>• Craft and send a large number of ICMP packets to different destination addresses.</li> <li>• Verify through the logs that the TOE detects the attack and that the traffic is dropped.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul> <p><b>c) <u>Protocol and port scanning:</u></b></p> <ul style="list-style-type: none"> <li>• Verify that the NDPP port scan detection is enabled on the TOE.</li> </ul> <p><b>i. <u>IP protocol scanning:</u></b></p> <ul style="list-style-type: none"> <li>• Send traffic for IP protocol scan using NMAP.</li> <li>• Verify through the logs that the TOE detects the possible port scan.</li> <li>• Verify through the packet capture that TOE reacts correctly to scanned protocols.</li> </ul> <p><b>ii. <u>TCP port scanning:</u></b></p> <ul style="list-style-type: none"> <li>• Send traffic for TCP protocol scan using NMAP.</li> <li>• Verify through the logs that the TOE detects the port scan.</li> <li>• Verify through the packet capture that TOE reacts correctly to scanned ports.</li> </ul> <p><b>iii. <u>UDP port scanning:</u></b></p> <ul style="list-style-type: none"> <li>• Send traffic for UDP protocol scan using NMAP.</li> <li>• Verify through the logs that the TOE detects the port scan.</li> <li>• Verify through the packet capture that TOE reacts correctly to scanned ports.</li> </ul> <p><b>iv. <u>ICMP scanning:</u></b></p> <ul style="list-style-type: none"> <li>• Send traffic for ICMP protocol scan using NMAP.</li> <li>• Verify through the logs that the TOE detects the port scan.</li> <li>• Verify through packet capture that no response was received for ICMP packets.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• TOE should detect each attack based on its corresponding signature and trigger the specified reaction as outlined in IPS_SBD_EXT.1.5, successfully stopping the attack.</li> <li>• TOE logs and packet capture should show that the attack is detected and traffic is dropped.</li> </ul>

<b>Pass/Fail with Explanation</b>	Pass. The TOE detects each attack based on its corresponding signature and triggers the specified reaction, effectively stopping the attack. This meets the testing requirements.
-----------------------------------	---

#### 7.8.19 IPS\_SBD\_EXT.1.5 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The test EAs for this element are performed in conjunction with those for IPS_SBD_EXT.1.1, IPS_SBD_EXT.1.2, IPS_SBD_EXT.1.3, and IPS_SBD_EXT.1.4.
<b>Pass/Fail with Explanation</b>	Pass. The test EAs for this element are performed in conjunction with those for IPS_SBD_EXT.1.1, IPS_SBD_EXT.1.2, IPS_SBD_EXT.1.3, and IPS_SBD_EXT.1.4.

#### 7.8.20 IPS\_SBD\_EXT.1.6 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall repeat one of the tests in IPS_SBD_EXT.1.2 Test 1 but generate multiple non-fragmented packets that contain the string in the rule defined. The evaluator shall verify that the malicious traffic is still detected when split across multiple non-fragmented packets.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure a Match object with the match type set to partial match to detect the EICAR test file string.</li> <li>• Configure an App rule and select the configured object to scan for the partial string, with the policy type set to the HTTP server.</li> <li>• The rule will be automatically enabled.</li> <li>• Run the HTTP server with the EICAR test file on the WAN VM and send a request for the EICAR test file from the LAN VM to the server.</li> <li>• Verify through the logs that the packet is dropped due to the configured rule.</li> <li>• Verify through packet capture that the EICAR test file does not pass through the TOE.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should detect malicious traffic even when split across multiple non-fragmented packets.</li> <li>• Packet capture and TOE logs should confirm that the malicious traffic is dropped.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE detects and drops malicious traffic even when split across multiple non-fragmented packets. This meets the testing requirements.

## 7.9.1 FCS\_IPSEC\_EXT.1.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator uses the guidance documentation to configure the TOE to carry out the following tests:</p> <p>Test 1: The evaluator shall configure the SPD such that there is a rule for dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext. The selectors used in the construction of the rule shall be different such that the evaluator can generate a packet and send packets to the gateway with the appropriate fields (fields that are used by the rule - e.g., the IP addresses, TCP/UDP ports) in the packet header. The evaluator performs both positive and negative test cases for each type of rule (e.g. a packet that matches the rule and another that does not match the rule). The evaluator observes via the audit trail, and packet captures that the TOE exhibited the expected behavior: appropriate packets were dropped, allowed to flow without modification, encrypted by the IPsec implementation.</p>
<b>Test Steps</b>	<p>Protect (Positive Case):</p> <ul style="list-style-type: none"> <li>• Configure an access rule to protect the ICMP traffic.</li> <li>• Start the IPsec connection and generate traffic.</li> <li>• Verify via logs that the ICMP packets are allowed.</li> <li>• Verify via packet capture that the packets are sent encrypted.</li> </ul> <p><b>Note- Negative Case: This is covered by the "deny" sub-test below.</b></p> <p>Deny (Positive Case):</p> <ul style="list-style-type: none"> <li>• Configure an access list to deny the ICMP traffic.</li> <li>• Start the IPsec connection and generate traffic.</li> <li>• Verify via logs that the ICMP packets are denied.</li> <li>• Verify via packet capture that the ICMP packets are dropped.</li> </ul> <p><b>Note- Negative Case: Covered by the "protect" (above) and "bypass" sub-test (below).</b></p> <p>Bypass (Positive Case):</p> <ul style="list-style-type: none"> <li>• Configure an access list to bypass the traffic.</li> <li>• Add a static route on TOE.</li> <li>• Generate traffic to match the configured rule.</li> <li>• Verify via logs that the ICMP packets are bypassed.</li> <li>• Verify via packet capture that the traffic is bypassed.</li> </ul> <p>Deny Bypass Traffic (Negative Case):</p>

	<ul style="list-style-type: none"> <li>• Configure an access list to deny the bypass traffic.</li> <li>• Generate traffic to match the configured rule.</li> <li>• Verify via logs that the ICMP packets are denied.</li> <li>• Verify via packet capture that the ICMP packets are dropped.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should be able to implement rules for dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext.</li> <li>• The TOE logs and packet capture should show that the packets are processed according to the configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE dropped packets when configured, encrypted packets when configured, and sent packets in plaintext when configured. This meets the testing requirements.

7.9.2 FCS\_IPSEC\_EXT.1.1 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator uses the guidance documentation to configure the TOE to carry out the following tests:</p> <p>Test 2: The evaluator shall devise several tests that cover a variety of scenarios for packet processing. As with Test 1, the evaluator ensures both positive and negative test cases are constructed. These scenarios must exercise the range of possibilities for SPD entries and processing modes as outlined in the TSS and guidance documentation. Potential areas to cover include rules with overlapping ranges and conflicting entries, inbound and outbound packets, and packets that establish SAs as well as packets that belong to established SAs. The evaluator shall verify, via the audit trail and packet captures, for each scenario that the expected behavior is exhibited, and is consistent with both the TSS and the guidance documentation.</p>
<b>Test Steps</b>	<p>Add conflicting rules:</p> <p>Covered in test case FFW_RUL_EXT.1.8 TEST #1.</p> <p>Protect a large set and deny a small subset:</p> <ul style="list-style-type: none"> <li>• Configure the device to discard a small subset of traffic (ICMP) and protect a large set of traffic.</li> <li>• Ping from the PEER and verify the ICMP request fails, and all other traffic is allowed.</li> <li>• Verify via logs that only the ICMP has been dropped and all other traffic is allowed.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify via that only the ICMP has been dropped and all other traffic is allowed.</li> <li>• Change the priorities for the created rules.</li> <li>• Generate traffic to match the configured rules. Verify that ICMP packets and all other traffic are allowed.</li> <li>• Verify via log that the ICMP packets and all other traffic are allowed.</li> <li>• Verify via packet capture that the ICMP packets and all other traffic are allowed.</li> </ul> <p>Bypass a large set and deny a small subset:</p> <ul style="list-style-type: none"> <li>• Configure the device to discard a small subset of traffic (ICMP) and bypass a large set of traffic.</li> <li>• Ping from the PEER and verify the ICMP request fails, and all other traffic is bypassed.</li> <li>• Verify via logs that only the ICMP has been dropped and all other traffic is bypassed.</li> <li>• Verify via that only the ICMP has been dropped and all other traffic is bypassed.</li> <li>• Change the priorities for the created rules.</li> <li>• Generate traffic to match the configured rules. Verify that ICMP packets and all other traffic are bypassed.</li> <li>• Verify via log that the ICMP packets and all other traffic are bypassed.</li> <li>• Verify via packet capture that the ICMP packets and all other traffic are bypassed.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should be able to encrypt/drop/bypass the traffic in sequence when configured with rules with overlapping ranges and conflicting entries, inbound and outbound packets, and packets that establish SAs as well as packets that belong to established SAs.</li> <li>• The TOE logs and packet capture should show that the packets are processed according to the configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE was able to successfully process the traffic according to the configured rules, even with overlapping ranges and conflicting entries. This meets the testing requirements.

7.9.3 FCS\_IPSEC\_EXT.1.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The assurance activity for this element is performed in conjunction with the activities for FCS_IPSEC_EXT.1.1.

	<p>The evaluator uses the guidance documentation to configure the TOE to carry out the following tests:</p> <p>The evaluator shall configure the SPD such that there is a rule for dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext. The evaluator may use the SPD that was created for verification of FCS_IPSEC_EXT.1.1. The evaluator shall construct a network packet that matches the rule to allow the packet to flow in plaintext and send that packet. The evaluator should observe that the network packet is passed to the proper destination interface with no modification. The evaluator shall then modify a field in the packet header; such that it no longer matches the evaluator-created entries (there may be a "TOE created" final entry that discards packets that do not match any previous entries). The evaluator sends the packet and observes that the packet was dropped.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure an access list to allow the bypass traffic.</li> <li>• Add a static route.</li> <li>• Generate traffic to match the configured rule.</li> <li>• Verify via logs that the ICMP packets are allowed.</li> <li>• Verify via packet capture that the ICMP packets are allowed.</li> <li>• Ping the modified IP address which does not match the configured access list.</li> <li>• Verify via logs that the ICMP packets to the modified IP address are denied.</li> <li>• Verify via packet capture that the ICMP packets to the modified IP address are denied.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The network packet should flow in plaintext through the network and reach the proper destination interface without any modification.</li> <li>• The TOE should be able to drop packets with the modified header.</li> <li>• The TOE logs and packet capture should show that the packets are processed according to the configured rules.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The packet matches the rule created by the evaluator, allowing it to be transmitted without encryption or alteration, confirming that the rule functions correctly for plaintext traffic. When the modified packet is sent, the TOE rejects the connection. This meets the testing requirements.</p>

7.9.4 FCS\_IPSEC\_EXT.1.3 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following test(s) based on the selections chosen:



	<p>Test 1: If <b>tunnel mode</b> is selected, the evaluator uses the guidance documentation to configure the TOE to operate in tunnel mode and also configures a VPN peer to operate in tunnel mode. The evaluator configures the TOE and the VPN peer to use any of the allowable cryptographic algorithms, authentication methods, etc. to ensure an allowable SA can be negotiated. The evaluator shall then initiate a connection from the TOE to connect to the VPN peer. The evaluator observes (for example, in the audit trail and the captured packets) that a successful connection was established using the tunnel mode.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the TOE for an IPsec connection in Tunnel Mode.</li> <li>• Configure the peer for an IPsec connection in Tunnel Mode.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify through logs that the connection is established using tunnel mode.</li> <li>• Verify through packet capture that the successful connection is established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should perform a successful connection using tunnel mode.</li> <li>• The TOE log should show that a successful connection was established using tunnel mode.</li> <li>• Packet capture should show that a successful IPsec connection was established.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE can be configured to operate in tunnel mode and successfully establish a connection with a peer using this mode. This meets the testing requirements.</p>

7.9.5 FCS\_IPSEC\_EXT.1.3 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following test(s) based on the selections chosen:</p> <p>Test 2: If <b>transport mode</b> is selected, the evaluator uses the guidance documentation to configure the TOE to operate in transport mode and also configures a VPN peer to operate in transport mode. The evaluator configures the TOE and the VPN peer to use any of the allowed cryptographic algorithms, authentication methods, etc. to ensure an allowable SA can be negotiated. The evaluator then initiates a connection from the TOE to connect to the VPN peer. The evaluator observes (for example, in the audit trail and the captured packets) that a successful connection was established using the transport mode.</p>
<b>Pass/Fail with Explanation</b>	<p>N/A. This test is not applicable because transport mode selection is not included in the ST.</p>

7.9.6 FCS\_IPSEC\_EXT.1.4 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall configure the TOE as indicated in the guidance documentation configuring the TOE to use each of the supported algorithms, attempt to establish a connection using ESP, and verify that the attempt succeeds.
<b>Test Steps</b>	<p><b>AES-CBC-128 and HMAC-SHA-256</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 AES-CBC-128 &amp; sha-256 configuration.</li> <li>• Configure the PEER for IKEv2 AES-CBC-128 &amp; sha-256 configuration.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-CBC-128 &amp; SHA256.</li> <li>• Verify via packet capture that the connection was established using AES-CBC-128 &amp; SHA256.</li> </ul> <p><b>AES-CBC-256 and HMAC-SHA-384</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 AES-CBC-256 &amp; sha-384 configuration.</li> <li>• Configure the PEER for IKEv2 AES-CBC-256 &amp; sha-384 configuration.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-CBC-256 &amp;&amp; SHA384.</li> <li>• Verify via packet capture that the connection was established using AES-CBC-256 &amp; SHA384.</li> </ul> <p><b>AES-CBC-192 and HMAC-SHA-512</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 AES-CBC-192 &amp; sha-512 configuration.</li> <li>• Configure the PEER for IKEv2 AES-CBC-192 &amp; sha-512 configuration.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-CBC-192 &amp; SHA512.</li> <li>• Verify via packet capture that the connection was established using AES-CBC-192 &amp;&amp; SHA512.</li> </ul> <p><b>AES-GCM-128</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE with an IKEv2 policy using AES-GCM-128.</li> <li>• Configure the PEER with an IKEv2 policy using AES-GCM-128.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> </ul>

	<ul style="list-style-type: none"> <li>• Verify via logs that the connection was established using AES-GCM-128.</li> <li>• Verify via packet capture that the connection is established using AES-GCM-128.</li> </ul> <p><b>AES-GCM-256</b></p> <ul style="list-style-type: none"> <li>• Configure the TOE with an IKEv2 policy using AES-GCM-256.</li> <li>• Configure the PEER with an IKEv2 policy using AES-GCM-256.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-GCM-256.</li> <li>• Verify via packet capture that the connection is established using AES-GCM-256.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should negotiate and establish a secure connection with the peer using the supported ESP algorithms.</li> <li>• The TOE log and packet capture should show that the IPsec connection was established using the claimed encryption and hash algorithm.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE can be configured with each supported algorithm and successfully negotiates and establishes a secure connection with the peer using these ESP algorithms. This meets the testing requirements.

#### 7.9.7 FCS\_IPSEC\_EXT.1.5 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>Tests are performed in conjunction with the other IPsec evaluation activities.</p> <p>Test 1: If <b>IKEv1</b> is selected, the evaluator shall configure the TOE as indicated in the guidance documentation and attempt to establish a connection using an IKEv1 Phase 1 connection in aggressive mode. This attempt should fail. The evaluator should then show that main mode exchanges are supported.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since 'IKEv1' selection is not included in the ST.

#### 7.9.8 FCS\_IPSEC\_EXT.1.5 TEST #2

Item	Data
<b>Test Assurance Activity</b>	Tests are performed in conjunction with the other IPsec evaluation activities.

	Test 2: If <b>NAT traversal</b> is selected within the IKEv2 selection, the evaluator shall configure the TOE so that it will perform NAT traversal processing as described in the TSS and RFC 5996, section 2.23. The evaluator shall initiate an IPsec connection and determine that the NAT is successfully traversed.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>Using a docker container in the VM, configure the VPN Peer.</li> <li>Configure the TOE with the VPN configuration and ensure that the VPN supports NAT traversal.</li> <li>Start the IPsec connection and verify that it is established.</li> <li>Generate some traffic to go through the tunnel.</li> <li>Verify with the logs that the TOE negotiated the VPN tunnel with NAT traversal.</li> <li>Verify with the packet capture that the TOE negotiated the VPN tunnel with NAT traversal.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>The TOE should successfully perform NAT traversal processing.</li> <li>TOE logs and packet capture should show that TOE negotiated the VPN tunnel with NAT traversal.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The NAT traversal is performed as expected, and the IPsec connection is established, traversing the NAT device without any issues. This meets the testing requirements.

7.9.9 FCS\_IPSEC\_EXT.1.6 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall configure the TOE to use the ciphersuite under test to encrypt the IKEv1 and/or IKEv2 payload and establish a connection with a peer device, which is configured to only accept the payload encrypted using the indicated ciphersuite. The evaluator will confirm the algorithm was that used in the negotiation.
<b>Test Steps</b>	<p>AES-CBC-128:</p> <ul style="list-style-type: none"> <li>Configure the TOE with an IKEv2 policy using AES-CBC-128.</li> <li>Configure the PEER with an IKEv2 policy using AES-CBC-128.</li> <li>Start the IPsec and verify that the connection is established.</li> <li>Generate traffic from TOE to peer.</li> <li>Verify via logs that the connection was established using AES-CBC-128.</li> <li>Verify via packet capture that the connection is established using AES-CBC-128.</li> </ul> <p>AES-CBC-256:</p> <ul style="list-style-type: none"> <li>Configure the TOE with an IKEv2 policy using AES-CBC-256.</li> <li>Configure the PEER with an IKEv2 policy using AES-CBC-256.</li> </ul>

	<ul style="list-style-type: none"> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-CBC-256.</li> <li>• Verify via packet capture that the connection is established using AES-CBC-256.</li> </ul> <p>AES-CBC-192:</p> <ul style="list-style-type: none"> <li>• Configure the TOE with an IKEv2 policy using AES-CBC-192.</li> <li>• Configure the PEER with an IKEv2 policy using AES-CBC-192.</li> <li>• Start the IPsec and verify that the connection is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using AES-CBC-192.</li> <li>• Verify via packet capture that the connection is established using AES-CBC-192.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should establish a connection with the peer device, and the connection should be maintained without errors or fallback to a different ciphersuite. This indicates that the TOE is correctly using the specified ciphersuite for encrypting the IKEv1 and/or IKEv2 payload</li> <li>• The TOE log and packet capture should show that the IKE session was established using the claimed ciphersuite.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE establishes a connection with the peer device, and the connection is maintained without errors or fallback to a different ciphersuite. This indicates that the TOE is correctly using the specified ciphersuite for encrypting the IKEv1 and/or IKEv2 payload. This meets the testing requirements.</p>

7.9.10 FCS\_IPSEC\_EXT.1.7 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>When testing this functionality, the evaluator needs to ensure that both sides are configured appropriately. From the RFC “A difference between IKEv1 and IKEv2 is that in IKEv1 SA lifetimes were negotiated. In IKEv2, each end of the SA is responsible for enforcing its own lifetime policy on the SA and rekeying the SA when necessary. If the two ends have different lifetime policies, the end with the shorter lifetime will end up always being the one to request the rekeying. If the two ends have the same lifetime policies, it is possible that both will initiate a rekeying at the same time (which will result in redundant SAs). To reduce the probability of this happening, the timing of rekeying requests SHOULD be jittered.”</p>

	<p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 1: If <b>'number of bytes'</b> is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime in terms of the number of bytes allowed following the guidance documentation. The evaluator shall configure a test peer with a byte lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer, and determine that once the allowed number of bytes through this SA is exceeded, a new SA is negotiated. The evaluator shall verify that the TOE initiates a Phase 1 negotiation.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable as 'number of bytes' selection is not included in the ST.

**7.9.11 FCS\_IPSEC\_EXT.1.7 TEST #2 [TD0800]**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>When testing this functionality, the evaluator needs to ensure that both sides are configured appropriately. From the RFC "A difference between IKEv1 and IKEv2 is that in IKEv1 SA lifetimes were negotiated. In IKEv2, each end of the SA is responsible for enforcing its own lifetime policy on the SA and rekeying the SA when necessary. If the two ends have different lifetime policies, the end with the shorter lifetime will end up always being the one to request the rekeying. If the two ends have the same lifetime policies, it is possible that both will initiate a rekeying at the same time (which will result in redundant SAs). To reduce the probability of this happening, the timing of rekeying requests SHOULD be jittered."</p> <p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 2: If <b>'length of time'</b> is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime no later than 24 hours for the Phase 1 SA following the guidance documentation. The evaluator shall configure a test peer with a Phase 1 SA lifetime that exceeds the Phase 1 SA lifetime on the TOE. The evaluator shall establish a SA between the TOE and the test peer, maintain the Phase 1 SA for 24 hours, and determine that a new Phase 1 SA is negotiated on or before 24 hours has elapsed. The evaluator shall verify that the TOE initiates a Phase 1 negotiation.</p> <p><b>TD0800 has been applied.</b></p>

<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the IPsec on TOE to have a phase 1 lifetime of 23 hours (82800 sec).</li> <li>• Configure the IPsec on Peer to have a lifetime of 24 hours (86400 sec) exceeding the Phase 1 SA lifetime on the TOE.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Verify through logs that the IPsec connection is established.</li> <li>• Verify through packet capture that the IPsec connection is established.</li> <li>• Check the tunnel status and verify that it is re-established.</li> <li>• Verify the rekey took place before 23 hours via logs.</li> <li>• Verify the rekey occurred via packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should initiate a new Phase 1 SA negotiation on or before 24 hours, resulting in the establishment of a new SA.</li> <li>• The TOE logs should show the session rekey.</li> <li>• The packet capture should show the establishment of a new SA.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE correctly initiates a new Phase 1 SA negotiation and establishes it within the 24-hour timeframe. This meets the testing requirements.

7.9.12 FCS\_IPSEC\_EXT.1.8 TEST #1

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>When testing this functionality, the evaluator needs to ensure that both sides are configured appropriately. From the RFC “A difference between IKEv1 and IKEv2 is that in IKEv1 SA lifetimes were negotiated. In IKEv2, each end of the SA is responsible for enforcing its own lifetime policy on the SA and rekeying the SA when necessary. If the two ends have different lifetime policies, the end with the shorter lifetime will end up always being the one to request the rekeying. If the two ends have the same lifetime policies, it is possible that both will initiate a rekeying at the same time (which will result in redundant SAs). To reduce the probability of this happening, the timing of rekeying requests SHOULD be jittered.”</p> <p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 1: If ‘<b>number of bytes</b>’ is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime in terms of the number of bytes allowed following the guidance documentation. The evaluator shall configure a test peer with a byte lifetime that exceeds the lifetime of the TOE. The evaluator shall establish a SA between the TOE and the test peer, and determine that once the allowed number of bytes through this SA</p>

	is exceeded, a new SA is negotiated. The evaluator shall verify that the TOE initiates a Phase 2 negotiation.
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable as 'number of bytes' selection is not included in the ST.

**7.9.13 FCS\_IPSEC\_EXT.1.8 TEST #2 [TD0800]**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>When testing this functionality, the evaluator needs to ensure that both sides are configured appropriately. From the RFC "A difference between IKEv1 and IKEv2 is that in IKEv1 SA lifetimes were negotiated. In IKEv2, each end of the SA is responsible for enforcing its own lifetime policy on the SA and rekeying the SA when necessary. If the two ends have different lifetime policies, the end with the shorter lifetime will end up always being the one to request the rekeying. If the two ends have the same lifetime policies, it is possible that both will initiate a rekeying at the same time (which will result in redundant SAs). To reduce the probability of this happening, the timing of rekeying requests SHOULD be jittered."</p> <p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 2: If '<b>length of time</b>' is selected as the SA lifetime measure, the evaluator shall configure a maximum lifetime no later than 8 hours for the Phase 2 SA following the guidance documentation. The evaluator shall configure a test peer with a Phase 2 SA lifetime that exceeds the Phase 2 SA lifetime on the TOE. The evaluator shall establish a SA between the TOE and the test peer, maintain the Phase 1 SA for 8 hours, and determine that once a new Phase 2 SA is negotiated when or before 8 hours has lapsed. The evaluator shall verify that the TOE initiates a Phase 2 negotiation.</p> <p><b>TD0800 has been applied.</b></p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure IPsec on the TOE to have a phase 2 lifetime of 7.5 hours (27000 sec).</li> <li>• Configure IPsec on the Peer to have a lifetime of 8 hours (28800 sec) that exceeds the Phase 2 SA lifetime on the TOE.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Verify that the rekey took place within the last 8 hours and that the tunnel is established via logs.</li> <li>• Verify the rekey occurred via packet capture.</li> </ul>



<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE successfully initiates a Phase 2 negotiation before or at the 8-hour mark.</li> <li>• The TOE logs should show the session rekey.</li> <li>• The packet capture should show the establishment of a new SA.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE successfully initiates a Phase 2 negotiation before the 8-hour mark, confirming that it adheres to the configured maximum SA lifetime. This meets the testing requirements.

#### 7.9.14 FCS\_IPSEC\_EXT.1.10 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 1: If the first selection is chosen, the evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating each nonce. The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.</p>
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by TSS Assurance Activities in the AAR.

#### 7.9.15 FCS\_IPSEC\_EXT.1.10 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:</p> <p>Test 2: If the second selection is chosen, the evaluator shall check to ensure that, for each PRF hash supported, the TSS describes the process for generating each nonce. The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this PP is used, and that the length of the nonces meet the stipulations in the requirement.</p>
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by covered by TSS Assurance Activities in the AAR.

#### 7.9.16 FCS\_IPSEC\_EXT.1.11 TEST #1

Item	Data
<b>Test Assurance Activity</b>	For each supported DH group, the evaluator shall test to ensure that all supported IKE protocols can be successfully completed using that particular DH group.
<b>Test Steps</b>	<p>DH group 14 (2048-bit MODP):</p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 group 14 configuration.</li> <li>• Configure the peer for IKEv2 group 14 configuration.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using group 14.</li> <li>• Verify via packet capture that the connection was established using group 14.</li> </ul> <p>DH group 19 (256-bit Random ECP):</p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 group 19 configuration.</li> <li>• Configure the peer for IKEv2 group 19 configuration.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using group 19.</li> <li>• Verify via packet capture that the connection was established using group 19.</li> </ul> <p>DH group 20 (384-bit Random ECP):</p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 group 20 configuration.</li> <li>• Configure the peer for IKEv2 group 20 configuration.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using group 20.</li> <li>• Verify via packet capture that the connection was established using group 20.</li> </ul> <p>DH group 21 (521-bit Random ECP):</p> <ul style="list-style-type: none"> <li>• Configure the TOE for IKEv2 group 21 configuration.</li> <li>• Configure the peer for IKEv2 group 21 configuration.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Generate traffic from TOE to peer.</li> <li>• Verify via logs that the connection was established using group 21.</li> <li>• Verify via packet capture that the connection was established using group 21.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• IKE protocol should complete the key exchange process using each supported DH group without errors.</li> <li>• The packet capture and logs should show that the IKE session was in the claimed exchange method.</li> </ul>

<b>Pass/Fail with Explanation</b>	Pass. The successful completion of IKE protocols using each DH group confirms that the TOE correctly implements and supports all configured DH groups, ensuring secure key exchanges. This meets the testing requirements.
-----------------------------------	--

#### 7.9.17 FCS\_IPSEC\_EXT.1.12 TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator simply follows the guidance to configure the TOE to perform the following tests.  Test 1: This test shall be performed for each version of IKE supported. The evaluator shall successfully negotiate an IPsec connection using each of the supported algorithms and hash functions identified in the requirements.
<b>Pass/Fail with Explanation</b>	Pass. This testing is covered in conjunction with the FCS_IPSEC_EXT.1.4 Test#1 and FCS_IPSEC_EXT.1.6 Test#1 test cases.

#### 7.9.18 FCS\_IPSEC\_EXT.1.12 TEST #2

Item	Data
<b>Test Assurance Activity</b>	The evaluator simply follows the guidance to configure the TOE to perform the following tests.  Test 2: This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish a SA for ESP that selects an encryption algorithm with more strength than that being used for the IKE SA (i.e., symmetric algorithm with a key size larger than that being used for the IKE SA). Such attempts should fail.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Configure the TOE such that Phase 2 encryption is stronger than Phase 1 and verify that it fails.</li> <li>• Attempt to configure TOE to use AES-CBC-128 in Phase 1 and AES-CBC-256 in Phase 2 ikev2.</li> <li>• Verify that TOE denies the configuration.</li> <li>• Verify the configuration denial using logs</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• TOE should deny configuration with a SA for ESP that selects an encryption algorithm with more strength than that being used for the IKE SA.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. TOE denies configuration with a SA for ESP that selects an encryption algorithm with more strength than that being used for the IKE SA. This meets the testing requirements.

#### 7.9.19 FCS\_IPSEC\_EXT.1.12 TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator simply follows the guidance to configure the TOE to perform the following tests.</p> <p>Test 3: This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an IKE SA using an algorithm that is not one of the supported algorithms and hash functions identified in the requirements. Such an attempt should fail.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"><li>• Configure the TOE to use AES128 and SHA512.</li><li>• Configure the Peer to use 3DES and SHA512.</li><li>• Start the IPsec connection and verify that it is not established.</li><li>• Verify the IPsec connection is not established using logs.</li><li>• Verify the IPsec connection is not established using packet capture.</li></ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"><li>• The TOE should only support and propose the configured algorithm. If the peer does not have matching algorithms this session should not be established.</li><li>• The packet capture and logs should show that the connection is not established.</li></ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE correctly rejects any attempts to establish an IKE SA with an algorithm or hash function that is not among the supported ones as specified in the requirements. This meets the testing requirements.</p>

#### 7.9.20 FCS\_IPSEC\_EXT.1.12 TEST #4

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator simply follows the guidance to configure the TOE to perform the following tests.</p> <p>Test 4: This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish a SA for ESP (assumes the proper parameters were used to establish the IKE SA) that selects an encryption algorithm that is not identified in FCS_IPSEC_EXT.1.4. Such an attempt should fail.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"><li>• Configure TOE to support AES-128, SHA 512 in Phase 1 and AES-128, SHA 512 in Phase 2.</li><li>• Configure Peer to support AES-128, SHA 512 in Phase 1 and 3-DES, SHA 512 in Phase 2.</li><li>• Start the IPsec connection and verify that it is not established.</li></ul>

	<ul style="list-style-type: none"> <li>• Verify the IPsec connection is not established using logs.</li> <li>• Verify the IPsec connection is not established using packet capture.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The evaluator's attempt to establish an SA for ESP using an encryption algorithm not identified in FCS_IPSEC_EXT.1.4 should fail.</li> <li>• The packet capture and logs should show that the connection is not established.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE correctly prevents the establishment of an SA for ESP using any encryption algorithm that is not listed in FCS_IPSEC_EXT.1.4. This meets the testing requirements.

#### 7.9.21 FCS\_IPSEC\_EXT.1.13 TEST #1

Item	Data
<b>Test Assurance Activity</b>	For efficiency sake, the testing is combined with the testing for FIA_X509_EXT.1, FIA_X509_EXT.2 (for IPsec connections), and FCS_IPSEC_EXT.1.1.
<b>Pass/Fail with Explanation</b>	Pass. The testing is combined with the testing for FIA_X509_EXT.1, FIA_X509_EXT.2 (for IPsec connections), and FCS_IPSEC_EXT.1.1.

#### 7.9.22 FCS\_IPSEC\_EXT.1.14 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:</p> <p>Test 1: [conditional] For each CN/identifier type combination selected, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the field in the peer's presented certificate and shall verify that the IKE authentication succeeds. If the TOE prioritizes CN checking over SAN (through explicit configuration of the field when specifying the reference identifier or prioritization rules), the evaluator shall also configure the SAN so it contains an incorrect identifier of the correct type (e.g. the reference identifier on the TOE is example.com, the CN=example.com, and the SAN:FQDN=otherdomain.com) and verify that IKE authentication succeeds.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since the TOE does not prioritize CN checking over SAN.

7.9.23 FCS\_IPSEC\_EXT.1.14 TEST #2

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:</p> <p>Test 2: [conditional] For each SAN/identifier type combination selected, the evaluator shall configure the peer’s reference identifier on the TOE (per the administrative guidance) to match the field in the peer’s presented certificate and shall verify that the IKE authentication succeeds. If the TOE prioritizes SAN checking over CN (through explicit specification of the field when specifying the reference identifier or prioritization rules), the evaluator shall also configure the CN so it contains an incorrect identifier formatted to be the same type (e.g. the reference identifier on the TOE is DNS-ID; identify certificate has an identifier in SAN with correct DNS-ID, CN with incorrect DNS-ID (and not a different type of identifier)) and verify that IKE authentication succeeds.</p>
<p><b>Test Steps</b></p>	<p>SAN as IPv4 address</p> <ul style="list-style-type: none"> <li>• Generate TOE certificate with correct SAN configured for IPv4.</li> <li>• Generate PEER certificate with incorrect CN and correct SAN configured for IPv4.</li> <li>• Import the End Entity certificates into the respective Device.</li> <li>• Configure the VPN policy in the TOE to match with the SAN: IP address of the PEER.</li> <li>• Configure the VPN policy in the PEER to match with the SAN: IP address of the TOE.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Verify via logs that the connection is established.</li> <li>• Verify via packet capture that the connection is established using a digital certificate.</li> </ul> <p>SAN as FQDN</p> <ul style="list-style-type: none"> <li>• Generate TOE certificate with correct SAN configured for FQDN.</li> <li>• Generate PEER certificate with incorrect CN and correct SAN configured for FQDN.</li> <li>• Import the End Entity certificates into the respective Device.</li> <li>• Configure the VPN policy in the TOE to match with the SAN: FQDN of the PEER.</li> <li>• Configure the VPN policy in the PEER to match with the SAN: FQDN of the TOE.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Verify via logs that the connection is established.</li> </ul>

	<ul style="list-style-type: none"> <li>Verify via packet capture that the connection is established using a digital certificate.</li> </ul> <p>SAN as User FQDN</p> <ul style="list-style-type: none"> <li>Generate TOE certificate with correct SAN configured for User FQDN.</li> <li>Generate PEER certificate with incorrect CN and correct SAN configured for User FQDN.</li> <li>Import the End Entity certificates into the respective Device.</li> <li>Configure the VPN policy in the TOE to match with the SAN: User FQDN of the PEER.</li> <li>Configure the VPN policy in the PEER to match with the SAN: User FQDN of the TOE.</li> <li>Start the IPsec connection and verify that it is established.</li> <li>Verify via logs that the connection is established.</li> <li>Verify via packet capture that the connection is established using a digital certificate.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>IKE authentication should succeed when the peer's reference identifier on the TOE matches the identifier in the SAN of the peer's certificate, even if the CN contains an incorrect identifier of the same type.</li> <li>Packet capture and TOE logs should show successful connection establishment.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE correctly prioritizes SAN over CN when validating the peer's certificate, ensuring IKE authentication succeeds only when the SAN matches the configured reference identifier, regardless of the CN value. This meets the testing requirements.</p>

7.9.24 FCS\_IPSEC\_EXT.1.14 TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:</p> <p>Test 3: [conditional] For each CN/identifier type combination selected, the evaluator shall:</p>

	<p>a) Create a valid certificate with the CN so it contains the valid identifier followed by '\0'. If the TOE prioritizes CN checking over SAN (through explicit specification of the field when specifying the reference identifier or prioritization rules) for the same identifier type, the evaluator shall configure the SAN so it matches the reference identifier.</p> <p>b) Configure the peer's reference identifier on the TOE (per the administrative guidance) to match the CN without the '\0' and verify that IKE authentication fails.</p>
<b>Pass/Fail with Explanation</b>	N/A. This test is not applicable since the TOE does not support the CN identifier type.

#### 7.9.25 FCS\_IPSEC\_EXT.1.14 TEST #4

Item	Data
<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:</p> <p>Test 4: [conditional] For each SAN/identifier type combination selected, the evaluator shall:</p> <p>a) Create a valid certificate with an incorrect identifier in the SAN. The evaluator shall configure a string representation of the correct identifier in the DN. If the TOE prioritizes CN checking over SAN (through explicit specification of the field when specifying the reference identifier or prioritization rules) for the same identifier type, the addition/modification shall be to any non-CN field of the DN. Otherwise, the addition/modification shall be to the CN.</p> <p>b) Configure the peer's reference identifier on the TOE (per the administrative guidance) to match the correct identifier (expected in the SAN) and verify that IKE authentication fails.</p>
<b>Test Steps</b>	<p>SAN as IP address</p> <ul style="list-style-type: none"> <li>• Generate a PEER certificate which contains incorrect SAN (IP address different from the one mentioned in the VPN policy) and string representation of the correct identifier in the DN.</li> <li>• Import this certificate into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the original SAN: IP address of the PEER.</li> <li>• Configure the VPN policy in the PEER to match the SAN: IP address in the TOE.</li> <li>• Start the IPsec connection and verify that it is not established.</li> </ul>



	<ul style="list-style-type: none"> <li>• Verify via logs that the connection is not established.</li> <li>• Verify via packet capture that the connection is not established.</li> </ul> <p>SAN as FQDN</p> <ul style="list-style-type: none"> <li>• Generate a PEER certificate which contains incorrect SAN (FQDN different from the one mentioned in the VPN policy) and string representation of the correct identifier in the DN.</li> <li>• Import this certificate into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the original FQDN of the PEER.</li> <li>• Configure the VPN policy in the PEER to match the FQDN in the TOE.</li> <li>• Start the IPsec connection and verify that it is not established.</li> <li>• Verify via logs that the connection is not established.</li> <li>• Verify via packet capture that the connection is not established.</li> </ul> <p>SAN as User FQDN</p> <ul style="list-style-type: none"> <li>• Generate a PEER certificate which contains incorrect SAN (User FQDN different from the one mentioned in the VPN policy) and string representation of the correct identifier in the DN.</li> <li>• Import this certificate into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the original SAN: User FQDN of the PEER.</li> <li>• Configure the VPN policy in the PEER to match the SAN: User FQDN in the TOE.</li> <li>• Start the IPsec connection and verify that it is not established.</li> <li>• Verify via logs that the connection is not established.</li> <li>• Verify via packet capture that the connection is not established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• IKE authentication should fail when the peer's reference identifier on the TOE matches the correct identifier expected in the SAN, but the certificate presents an incorrect identifier in the SAN.</li> <li>• The packet capture and logs should show that the connection is not established.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE correctly rejects the IKE authentication attempt when the SAN contains an incorrect identifier, even if the DN (including the CN) contains a correct or matching identifier. This demonstrates that the TOE does not prioritize incorrect or lower-priority fields over the SAN when performing authentication. This meets the testing requirements.</p>

7.9.26 FCS\_IPSEC\_EXT.1.14 TEST #5

Item	Data
------	------

<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:  Test 5: [conditional] If the TOE supports DN identifier types, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the subject DN in the peer's presented certificate and shall verify that the IKE authentication succeeds.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Import the certificates into the TOE.</li> <li>• Import the certificates into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the DN of the PEER.</li> <li>• Configure the VPN policy in the PEER to match the DN of the TOE.</li> <li>• Start the IPsec connection and verify that it is established.</li> <li>• Verify via logs that the IPsec connection is established.</li> <li>• Verify via packet capture that the connection is established using a digital certificate.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• IKE authentication should succeed when the peer's reference identifier on the TOE is configured to match the subject DN in the peer's presented certificate.</li> <li>• The packet capture and logs should show that the connection is established.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The TOE successfully establishes IKE authentication when the subject DN in the peer's certificate matches the configured reference identifier on the TOE, demonstrating that the TOE correctly handles DN identifier types during authentication.</p>

7.9.27 FCS\_IPSEC\_EXT.1.14 TEST #6A

Item	Data
<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:  Test 6: [conditional] If the TOE supports DN identifier types, to demonstrate a bit-wise comparison of the DN, the evaluator shall create the following valid certificates and verify that the IKE authentication fails when each certificate is presented to the TOE:</p>

	a) Duplicate the CN field, so the otherwise authorized DN contains two identical CNs.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Generate a PEER certificate with duplicate CN.</li> <li>• Import the certificate into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the DN of the PEER without the duplicate CN.</li> <li>• Configure the VPN policy in the PEER to match the DN of the TOE.</li> <li>• Start the IPsec connection and verify that the tunnel has not been established.</li> <li>• Verify via logs that the connection is not established.</li> <li>• Verify via packet capture that the connection is not established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• IKE authentication should fail when a certificate with two identical CN fields in the DN is presented to the TOE.</li> <li>• The packet capture and logs should show that the connection is not established.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE correctly rejects the IKE authentication attempt when the DN contains duplicate CN fields, demonstrating that the TOE performs a strict bit-wise comparison of the DN and does not accept certificates with duplicated or non-compliant DN structures. This meets the testing requirements.

7.9.28 FCS\_IPSEC\_EXT.1.14 TEST #6B

Item	Data
<b>Test Assurance Activity</b>	<p>In the context of the tests below, a valid certificate is a certificate that passes FIA_X509_EXT.1 validation checks but does not necessarily contain an authorized subject.</p> <p>The evaluator shall perform the following tests:</p> <p>Test 6: If the TOE supports DN identifier types, to demonstrate a bit-wise comparison of the DN, the evaluator shall create the following valid certificates and verify that the IKE authentication fails when each certificate is presented to the TOE:</p> <p>b) Append '\0' to a non-CN field of an otherwise authorized DN.</p>
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Generate a PEER certificate with \0 appended in the 'Organization' field using the x509-mod tool.</li> <li>• Import the certificate into the PEER.</li> <li>• Configure the VPN policy in the TOE to match the DN of the PEER without the \0 appended.</li> <li>• Configure the VPN policy in the PEER to match the DN of the TOE.</li> </ul>

	<ul style="list-style-type: none"> <li>Start the IPsec connection and verify that it is not established.</li> <li>Verify via logs that the connection is not established.</li> <li>Verify via packet capture that the connection is not established.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>IKE authentication should fail when a certificate with a '\0' (null character) appended to a non-CN field in the DN is presented to the TOE.</li> <li>The packet capture and logs should show that the connection is not established.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The TOE correctly rejects the IKE authentication attempt when the DN includes a null character appended to a non-CN field. This demonstrates that the TOE performs a strict bit-wise comparison of the DN and does not accept certificates with non-standard or improperly formatted DN fields. This meets the testing requirements.

## 7.10 VPNGW

### 7.10.1 FCS\_COP.1/DATAENCRYPTION TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to require the ST author to make certain selections, but these selections are all part of the original definition of the SFR so no new behavior is defined by the PP-Module.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FCS_COP.1/DataEncryption from the Crypto module.

### 7.10.2 FCS\_IPSEC\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There are no additional testing activities.
<b>Pass/Fail with Explanation</b>	There are no additional testing activities.

### 7.10.3 FIA\_X509\_EXT.1/REV TEST #1

Item	Data
------	------

<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FIA_X509_EXT.1/Rev Test#1 from the X509 module.

#### 7.10.4 FIA\_X509\_EXT.2 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to support its use for IPsec at a minimum. The evaluator shall ensure that all evaluation of this SFR is performed against its use in IPsec communications as well as any other supported usage.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FIA_X509_EXT.2 Test#1 from the X509 module.

#### 7.10.5 FIA\_X509\_EXT.3 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FIA_X509_EXT.3 Test#1 from the X509 module.

#### 7.10.6 FMT\_MTD.1/CRYPTOKEYS TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to make it mandatory because of the TOE's required support for IPsec.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FMT_MTD.1/Cryptokeys Test#1 from the Auth module.

#### 7.10.7 FPT\_TST\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module requires a particular self-test to be performed, but this self-test is still evaluated using the same methods specified in the Supporting Document.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FPT_TST_EXT.1/Test#1 from the update module.

#### 7.10.8 FPT\_TUD\_EXT.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	There is no change to the Evaluation Activities specified for this SFR in the NDcPP Supporting Document. The PP-Module modifies this SFR to mandate that a particular selection be chosen, but this selection is part of the original definition of the SFR so no new behavior is defined by the PP-Module.
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FPT_TUD_EXT.1/Test#1 from the update module.

#### 7.10.9 FAU\_GEN.1/VPN TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall test the audit functionality by performing actions that trigger each of the claimed audit events and verifying that the audit records are accurate and that their format is consistent with what is specified in the operational guidance. The evaluator may generate these audit events as a consequence of performing other tests that would cause these events to be generated.
<b>Test Steps</b>	<ul style="list-style-type: none"> <li>• Trigger each auditable event on the TOE. Verify that each audit record is generated and contains the required information.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should accurately generate audit records for all the required auditable events described in the ST.</li> <li>• The audit records generated should match the format specified in the guidance documentation.</li> <li>• Evidence- Audit logs generated for each SFR.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FAU_GEN.1 Test#1 from the audit module.

7.10.10 FCS\_CKM.1/IKE TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p><b>For FFC Schemes using “safe-prime” groups:</b></p> <p>Testing for FFC Schemes using safe-prime groups is done as part of testing in FCS_CKM.2.</p> <p><b>For all other selections:</b></p> <p>The evaluator shall perform the corresponding tests for FCS_CKM.1 specified in the NDcPP SD, based on the selections chosen for this SFR. If IKE key generation is implemented by a different algorithm than the NDcPP key generation function, the evaluator shall ensure this testing is performed using the correct implementation.</p>
<b>Pass/Fail with Explanation</b>	Pass. This test is covered by FCS_CKM.1 FFC – “SAFE-PRIME” GROUPS from the crypto module.

7.10.11 FMT\_SMF.1/VPN TEST #1

Item	Data
<b>Test Assurance Activity</b>	The evaluator tests management functions as part of performing other test EAs. No separate testing for FMT_SMF.1/VPN is required unless one of the management functions in FMT_SMF.1.1/VPN has not already been exercised under any other SFR.
<b>Pass/Fail with Explanation</b>	Pass. No separate testing for FMT_SMF.1/VPN is required as all of the management functions in FMT_SMF.1.1/VPN have already been exercised.

7.10.12 FPF\_RUL\_EXT.1.1 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.1:1: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would otherwise be denied by the ruleset should be sourced and directed to a host. The evaluator shall use a packet sniffer to verify none of the generated network traffic is permitted through the TOE during initialization.</p> <p>Note: The remaining testing associated with application of the ruleset is addressed in the subsequent test EAs.</p>
<b>Pass/Fail with Explanation</b>	Pass. This test is performed in conjunction with FFW_RUL_EXT.1 Test #1.

---

**7.10.13 FPF\_RUL\_EXT.1.1 TEST #2**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.1:2: The evaluator shall attempt to get network traffic to flow through the TOE while the TOE is being initialized. A steady flow of network packets that would be permitted by the ruleset should be sourced and directed to a host. The evaluator shall use a packet sniffer to verify none of the generated network traffic is permitted through the TOE during initialization and is only permitted once initialization is complete.</p> <p>Note: The remaining testing associated with application of the ruleset is addressed in the subsequent test EAs.</p>
<b>Pass/Fail with Explanation</b>	Pass. This test is performed in conjunction with FFW_RUL_EXT.1 Test #2.

---

**7.10.14 FPF\_RUL\_EXT.1.2 TEST #1**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	There are no EAs specified for this element. Definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces is described collectively under FPF_RUL_EXT.1.4.
<b>Pass/Fail with Explanation</b>	Pass. The definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces are described collectively under FPF_RUL_EXT.1.4.

---

**7.10.15 FPF\_RUL\_EXT.1.3 TEST #1**

<b>Item</b>	<b>Data</b>
<b>Test Assurance Activity</b>	There are no EAs specified for this element. Definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces is described collectively under FPF_RUL_EXT.1.4.
<b>Pass/Fail with Explanation</b>	Pass. The definition of packet filtering policy, association of operations with packet filtering rules, and association of these rules to network interfaces is described collectively under FPF_RUL_EXT.1.4.



7.10.16 FPF\_RUL\_EXT.1.4 TEST #1

Item	Data
<p><b>Test Assurance Activity</b></p>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.4:1: The evaluator shall use the instructions in the operational guidance to test that packet filter rules can be created that permit, discard, and log packets for each of the following attributes:</p> <ul style="list-style-type: none"> <li>• IPv4 <ul style="list-style-type: none"> <li>○ Destination Address</li> <li>○ Protocol</li> </ul> </li> <li>• IPv6 <ul style="list-style-type: none"> <li>○ Source address</li> <li>○ Destination Address</li> <li>○ Next Header (Protocol)</li> </ul> </li> <li>• TCP <ul style="list-style-type: none"> <li>○ Source Port</li> <li>○ Destination Port</li> </ul> </li> <li>• UDP <ul style="list-style-type: none"> <li>○ Source Port</li> <li>○ Destination Port</li> </ul> </li> </ul> <p>Note that these test activities should be performed in conjunction with those of FPF_RUL_EXT.1.6 where the effectiveness of the rules is tested; here the evaluator is just ensuring the guidance is sufficient and the TOE supports the administrator creating a ruleset based on the above attributes. The test activities for FPF_RUL_EXT.1.6 define the combinations of protocols and attributes required to be tested. If those combinations are configured manually, that will fulfill the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.</p>
<p><b>Pass/Fail with Explanation</b></p>	<p>Pass. This test is performed in conjunction with FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #1.</p>

7.10.17 FPF\_RUL\_EXT.1.4 TEST #2

Item	Data
------	------

<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.4:2: The evaluator shall repeat Test 1 above for each distinct network interface type supported by the TOE to ensure that packet filtering rules can be defined for all supported types.</p> <p>Note that these test activities should be performed in conjunction with those of FPF_RUL_EXT.1.6 where the effectiveness of the rules is tested; here the evaluator is just ensuring the guidance is sufficient and the TOE supports the administrator creating a ruleset based on the above attributes. The test activities for FPF_RUL_EXT.1.6 define the combinations of protocols and attributes required to be tested. If those combinations are configured manually, that will fulfill the objective of these test activities, but if those combinations are configured otherwise (e.g., using automation), these test activities may be necessary in order to ensure the guidance is correct and the full range of configurations can be achieved by a TOE administrator.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. This test is performed in conjunction with FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #1 and FFW_RUL_EXT.1.2/FFW_RUL_EXT.1.3/FFW_RUL_EXT.1.4 Test #2.</p>

7.10.18 FPF\_RUL\_EXT.1.5 TEST #1


Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.5:1: The evaluator shall devise two equal packet filtering rules with alternate operations – permit and discard. The rules should then be deployed in two distinct orders and in each case the evaluator shall ensure that the first rule is enforced in both cases by generating applicable packets and using packet capture and logs for confirmation.</p>
<b>Pass/Fail with Explanation</b>	<p>Pass. This test is performed in conjunction with FFW_RUL_EXT.1.8 Test #1.</p>

7.10.19 FPF\_RUL\_EXT.1.5 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p>

	Test FPF_RUL_EXT.1.5:2: The evaluator shall repeat the procedure above, except that the two rules should be devised where one is a subset of the other (e.g., a specific address vs. a network segment). Again, the evaluator shall test both orders to ensure that the first is enforced regardless of the specificity of the rule.
<b>Pass/Fail with Explanation</b>	Pass. This test is performed in conjunction with FFW_RUL_EXT.1.8 Test #2.


#### 7.10.20 FPF\_RUL\_EXT.1.6 TEST #1

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:1: The evaluator shall configure the TOE to permit and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each supported IPv4 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Any protocols not supported by the TOE must be denied.</p> <p>IP Transport Layer Protocols:</p>  <p>IP Transport Layer Protocols.xlsx</p>
<b>Test Steps</b>	<p><b><u>Specific source and Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for specific source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul> <p><b><u>Specific source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for specific source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul>

	<p><b><u>Wildcard source Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for wildcard source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul> <p><b><u>Wildcard source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for wildcard source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for permitting each IPv4 traffic flow and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Logs should show that the correct traffic is permitted.</li> <li>• Packet capture should show that the supported protocols are permitted.</li> </ul>
<b>Pass/Fail with Explanation</b>	<p>Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was permitted and logged. This meets the requirement.</p>


7.10.21 FPF\_RUL\_EXT.1.6 TEST #2

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:2: The evaluator shall configure the TOE to permit all traffic except to discard and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv4 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must also be denied but are not required to be logged.</p>

	<p>IP Transport Layer Protocols:</p>  <p>IP Transport Layer Protocols.xlsx</p>
<p><b>Test Steps</b></p>	<p><b><u>Specific source and Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create a deny rule for specific source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was denied through the interface.</li> <li>• Verify through packet capture that packets were dropped due to the configured rule.</li> </ul> <p><b><u>Specific source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create a deny rule for the specific source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was denied through the interface.</li> <li>• Verify through packet capture that packets were dropped due to the configured rule.</li> </ul> <p><b><u>Wildcard source Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create a deny rule for the wildcard source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was denied through the interface.</li> <li>• Verify through packet capture that packets were dropped due to the configured rule.</li> </ul> <p><b><u>Wildcard source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create a deny rule for wildcard source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was denied through the interface.</li> <li>• Verify through packet capture that packets were dropped due to the configured rule.</li> </ul>
<p><b>Expected Test Results</b></p>	<ul style="list-style-type: none"> <li>• The TOE should create rules for denying each IPV4 traffic flow and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Logs should show that the correct traffic is denied.</li> <li>• Packet capture should show that the packets were dropped due to configured rule..</li> </ul>

<b>Pass/Fail with Explanation</b>	Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was denied and logged. This meets the testing requirement.
-----------------------------------	---

7.10.22 FPF\_RUL\_EXT.1.6 TEST #3

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:3: The evaluator shall configure the TOE to permit and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. Additionally, the evaluator shall configure the TOE to discard and log each supported IPv4 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with different (than those permitted above) combinations of a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each supported IPv4 Transport Layer Protocol and outside the scope of all source and destination addresses configured above in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must be denied.</p> <p>IP Transport Layer Protocols:</p>  <p>IP Transport Layer Protocols.xlsx</p>
<b>Test Steps</b>	<p><b>SrcA: IP 10.1.3.107, DstA: IP 10.1.4.116</b></p> <p><b>SrcB: IP 10.1.3.109 DstB: IP 10.1.9.117</b></p> <p><b>SrcC IP 10.1.2.7, DstC: IP 10.1.7.116</b></p> <p><b><u>Specific source and Specific destination:</u></b></p>

- Create a filter to permit traffic that contains a specific source address SrcA and specific destination address DstA but deny traffic that contains a specific source address SrcB and specific destination address DstB. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.
- Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.
- Verify through packet capture that traffic with a specific source address (SrcA) and a specific destination address (DstA) was permitted.
- Verify through packet capture that traffic with a specific source address (SrcB) and a specific destination address (DstB) was denied.
- Verify through packet capture that all other traffic was discarded.

**Specific source Wildcard destination:**

- Create a filter to permit traffic that contains a specific source address SrcA and wildcard destination address but denies traffic that contains a specific source address SrcB and wildcard destination address. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.
- Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.
- Verify through packet capture that traffic with a specific source address (SrcA) and wildcard destination address was permitted.
- Verify through packet capture that traffic with a specific source address (SrcB) and wildcard destination address was denied.
- Verify through packet capture that all other traffic was discarded.

**Wildcard source Specific destination:**


- Create a filter to permit traffic that contains a wildcard source address and specific destination address DstA but deny traffic that contains a wildcard source address and specific destination address DstB. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.
- Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.
- Verify through packet capture that traffic with a wildcard source address and a specific destination address (DstA) was permitted.
- Verify through packet capture that traffic with a wildcard source address and a specific destination address (DstB) was denied.
- Verify through packet capture that all other traffic was discarded.

**Wildcard source Wildcard destination:**

- Create a filter to permit traffic that contains a wildcard source address and wildcard destination address but deny traffic that contains a wildcard source address and wildcard destination address. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.

	<ul style="list-style-type: none"> <li>• Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.</li> <li>• Verify through packet capture that traffic with a wildcard source address and wildcard destination address was permitted.</li> <li>• Verify through packet capture that traffic with a wildcard source address and wildcard destination address was denied.</li> <li>• Verify through packet capture that all other traffic was discarded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should filter traffic based on the order of the ACL configured. When configured with the permit rule first, the traffic is allowed to pass. When configured with the deny rule first, the traffic is not allowed to pass.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL and assigning it on an interface.</li> <li>• Log showing the behavior of traffic.</li> <li>• Packet capture showing the behavior of traffic.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was logged and not permitted. This meets the testing requirement.

7.10.23 FPF\_RUL\_EXT.1.6 TEST #4

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:4: The evaluator shall configure the TOE to permit and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Any protocols not supported by the TOE must be denied.</p> <p>IP Transport Layer Protocols:</p>  <p>IP Transport Layer Protocols.xlsx</p>



<b>Test Steps</b>	<p><b><u>Specific source and Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for specific source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul> <p><b><u>Specific source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for specific source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul> <p><b><u>Wildcard source Specific destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for wildcard source address and specific destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul> <p><b><u>Wildcard source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create permit rule for wildcard source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was permitted through the interface.</li> <li>• Verify through packet capture that the supported protocols are permitted.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for permitting each IPv6 traffic flow and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Logs should show that the correct traffic is permitted. Packet capture should show that the supported</li> <li>• protocols are permitted.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was permitted and logged. This meets the requirement.

7.10.24 FPF\_RUL\_EXT.1.6 TEST #5

Item	Data
<b>Test Assurance Activity</b>	The evaluator shall perform the following tests:

Test FPF\_RUL\_EXT.1.6:5: The evaluator shall configure the TOE to permit all traffic except to discard and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and within the configured source and destination addresses in order to ensure that the supported protocols are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must also be denied but are not required to be logged.

IP Transport Layer Protocols:



IP Transport Layer  
Protocols.xlsx

## Test Steps

### **Specific source and Specific destination:**

- Create a deny rule for specific source address and specific destination address.
- Generate traffic to hit the access rule.
- Verify through logs that the correct traffic was denied through the interface.
- Verify through packet capture that packets were dropped due to the configured rule.

### **Specific source Wildcard destination:**

- Create a deny rule for the specific source address and wildcard destination address.
- Generate traffic to hit the access rule.
- Verify through logs that the correct traffic was denied through the interface.
- Verify through packet capture that packets were dropped due to the configured rule.

### **Wildcard source Specific destination:**

- Create a deny rule for the wildcard source address and specific destination address.
- Generate traffic to hit the access rule.
- Verify through logs that the correct traffic was denied through the interface.
- Verify through packet capture that packets were dropped due to the configured rule.

### **Wildcard source Wildcard destination:**

	<ul style="list-style-type: none"> <li>• Create a deny rule for the wildcard source address and wildcard destination address.</li> <li>• Generate traffic to hit the access rule.</li> <li>• Verify through logs that the correct traffic was denied through the interface.</li> <li>• Verify through packet capture that packets were dropped due to the configured rule.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for denying each IPV6 traffic flow and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Logs should show that the correct traffic is denied.</li> <li>• Packet capture should show that the packets were dropped due to the configured rule.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was denied and logged. This meets the testing requirement.

7.10.25 FPF\_RUL\_EXT.1.6 TEST #6

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:6: The evaluator shall configure the TOE to permit and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. Additionally, the evaluator shall configure the TOE to discard and log each supported IPv6 Transport Layer Protocol (see RFC Values for IPv4 and IPv6 table for full possible list) in conjunction with different (than those permitted above) combinations of a specific source address and specific destination address, specific source address and wildcard destination address, wildcard source address and specific destination address, and wildcard source address and wildcard destination address. The evaluator shall generate packets matching each defined IPv6 Transport Layer Protocol and outside the scope of all source and destination addresses configured above in order to ensure that the supported protocols are dropped (i.e., by capturing no applicable packets passing through the TOE) and logged. Any protocols not supported by the TOE must be denied.</p> <p>IP Transport Layer Protocols:</p>



IP Transport Layer  
Protocols.xlsx

## Test Steps

SrcA: IP 2001:10:1:3:107::202 , DstA: IP 2001:10:1:4:116::102

SrcB: IP 2001:10:1:3:109::202, DstB: IP 2001:10:1:7:116::102

Other: SrcC IP 2001:10:1:5:108::202 , DstC: IP 2001:10:1:10:108::202

Subnet: 2001:10:1::/48

### **Specific source and Specific destination:**

- Create a filter to permit traffic that contains a specific source address SrcA and specific destination address DstA but deny traffic that contains a specific source address SrcB and specific destination address DstB. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.
- Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.
- Verify through packet capture that traffic with a specific source address (SrcA) and a specific destination address (DstA) was permitted.
- Verify through packet capture that traffic with a specific source address (SrcB) and a specific destination address (DstB) was denied.
- Verify through packet capture that all other traffic was discarded.

### **Specific source Wildcard destination:**

- Create a filter to permit traffic that contains a specific source address SrcA and wildcard destination address but denies traffic that contains a specific source address SrcB and wildcard destination address. All other traffic is discarded.
- Generate traffic to match the filters applied to the TOE's interface.
- Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.
- Verify through packet capture that traffic with a specific source address (SrcA) and wildcard destination address was permitted.
- Verify through packet capture that traffic with a specific source address (SrcB) and wildcard destination address was denied.
- Verify through packet capture that all other traffic was discarded.

### **Wildcard source Specific destination:**

- Create a filter to permit traffic that contains a wildcard source address and specific destination address DstA but deny traffic that contains a wildcard source address and specific destination address DstB. All other traffic is discarded.

	<ul style="list-style-type: none"> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.</li> <li>• Verify through packet capture that traffic with a wildcard source address and a specific destination address (DstA) was permitted.</li> <li>• Verify through packet capture that traffic with a wildcard source address and a specific destination address (DstB) was denied.</li> <li>• Verify through packet capture that all other traffic was discarded.</li> </ul> <p><b><u>Wildcard source Wildcard destination:</u></b></p> <ul style="list-style-type: none"> <li>• Create a filter to permit traffic that contains a wildcard source address and wildcard destination address but deny traffic that contains a wildcard source address and wildcard destination address. All other traffic is discarded.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the correct traffic was permitted through the interface and remaining traffic is discarded.</li> <li>• Verify through packet capture that traffic with a wildcard source address and wildcard destination address was permitted.</li> <li>• Verify through packet capture that traffic with a wildcard source address and wildcard destination address was denied.</li> <li>• Verify through packet capture that all other traffic was discarded.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should filter traffic based on the order of the ACL configured. When configured with the permit rule first, the traffic is allowed to pass. When configured with the deny rule first, the traffic is not allowed to pass.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL and assigning it on an interface.</li> <li>• Log showing the behavior of traffic.</li> <li>• Packet capture showing the behavior of traffic.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that a rule can be configured for each of the traffic flow. Each traffic flow was logged and not permitted. This meets the testing requirement.

7.10.26 FPF\_RUL\_EXT.1.6 TEST #7

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:7: The evaluator shall configure the TOE to permit and log protocol 6 (TCP) using a selected source port, a selected destination port, and a selected source and destination port combination. The evaluator shall generate packets matching</p>

	the configured source and destination TCP ports in order to ensure that they are permitted (i.e., by capturing the packets after passing through the TOE) and logged.
<b>Test Steps</b>	<p>For specific source port:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit Transport Layer Protocol 6 using a specific source port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the correct traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul> <p>For specific destination port:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit Transport Layer Protocol 6 using a specific destination port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the correct traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul> <p>For specific source and destination port combination:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit Transport Layer Protocol 6 using a specific source and destination port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the correct traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for permitting each protocol 6 source and destination ports and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Log showing each traffic match.</li> <li>• Packet capture showing each traffic flow.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that when configured TCP traffic can flow through the TOE. This meets the testing requirement.

7.10.27 FPF\_RUL\_EXT.1.6 TEST #8

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:8: The evaluator shall configure the TOE to discard and log protocol 6 (TCP) using a selected source port, a selected destination port, and a selected</p>

	source and destination port combination. The evaluator shall generate packets matching the configured source and destination TCP ports in order to ensure that they are denied (i.e., by capturing no applicable packets passing through the TOE) and logged.
<b>Test Steps</b>	<p>For specific source port:</p> <ul style="list-style-type: none"> <li>• Create a filter to deny Transport Layer Protocol 6 using a specific source port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul> <p>For specific destination port:</p> <ul style="list-style-type: none"> <li>• Create a filter to deny Transport Layer Protocol 6 using a specific destination port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul> <p>For specific source and destination port combination:</p> <ul style="list-style-type: none"> <li>• Create a filter to deny Transport Layer Protocol 6 using a specific source and destination port.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify the logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for denying each protocol 6 source and destination ports and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Log showing each traffic match.</li> <li>• Packet capture showing each traffic flow.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test showed that when configured with deny rules TCP traffic will not be permitted. This meets the testing requirement.

7.10.28 FPF\_RUL\_EXT.1.6 TEST #9

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:9: The evaluator shall configure the TOE to permit and log protocol 17 (UDP) using a selected source port, a selected destination port, and a</p>

	selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination UDP ports in order to ensure that they are permitted (i.e., by capturing the packets after passing through the TOE) and logged. Here the evaluator shall ensure that the UDP port 500 (IKE) is included in the set of tests.
<b>Test Steps</b>	<p>For specific source port:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit UDP protocol 17 using a specific source port 500.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul> <p>For specific destination port:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit UDP protocol 17 using a specific destination port 500.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul> <p>For specific source and destination port combination:</p> <ul style="list-style-type: none"> <li>• Create a filter to permit UDP protocol 17 using a specific source and destination port 500 &amp; 500 respectively.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was permitted through the interface.</li> <li>• Verify with packet capture that the traffic was permitted.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for permitting each protocol 17 source and destination ports and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Log showing each traffic match.</li> <li>• Packet capture showing each traffic flow.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test case showed that when configured UDP traffic can flow through the TOE. This meets the testing requirement.

7.10.29 FPF\_RUL\_EXT.1.6 TEST #10

Item	Data
<b>Test Assurance Activity</b>	<p>The evaluator shall perform the following tests:</p> <p>Test FPF_RUL_EXT.1.6:10: The evaluator shall configure the TOE to discard and log protocol 17 (UDP) using a selected source port, a selected destination port, and a</p>



	selected source and destination port combination. The evaluator shall generate packets matching the configured source and destination UDP ports in order to ensure that they are denied (i.e., by capturing no applicable packets passing through the TOE) and logged. Again, the evaluator shall ensure that UDP port 500 is included in the set of tests.
<b>Test Steps</b>	<p>For specific source port:</p> <ul style="list-style-type: none"> <li>• Create a filter to deny UDP protocol 17 using a specific source port 500.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul> <p>For specific destination port:</p> <ul style="list-style-type: none"> <li>• Create a filter to deny UDP protocol 17 using a specific destination port 500.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul> <p>For specific source and destination port combination:</p> <ul style="list-style-type: none"> <li>• Create a filter to drop UDP protocol 17 using a specific source and destination port 500 &amp; 500.</li> <li>• Generate traffic to match the filters applied to the TOE’s interface.</li> <li>• Verify through logs that the traffic was denied through the interface.</li> <li>• Verify with packet capture that the traffic was denied.</li> </ul>
<b>Expected Test Results</b>	<ul style="list-style-type: none"> <li>• The TOE should create rules for denying each protocol 17 source and destination ports and log each traffic match.</li> <li>• Evidence (screenshot or CLI output) showing a configuration of ACL.</li> <li>• Log showing each traffic match.</li> <li>• Packet capture showing each traffic flow.</li> </ul>
<b>Pass/Fail with Explanation</b>	Pass. The test showed that when configured with deny rules UDP traffic will not be permitted. This meets the testing requirement.

7.10.30 FPT\_FLS.1/SELFTEST TEST #1

Item	Data
<b>Test Assurance Activity</b>	There are no test EAs for this component.
<b>Pass/Fail with Explanation</b>	There are no test EAs for this component.

---

7.10.31 FPT\_TST\_EXT.3 TEST #1

Item	Data
Test Assurance Activity	There are no test EAs for this component.
Pass/Fail with Explanation	There are no test EAs for this component.

---

7.10.32 FTP\_ITC.1/VPN TEST #1

Item	Data
Test Assurance Activity	The EAs specified for FTP_ITC.1 in the Supporting Document for the Base-PP shall be applied for IPsec VPN communications. Additional testing for IPsec is covered in FCS_IPSEC_EXT.1.
Pass/Fail with Explanation	Pass. This test is covered by FTP_ITC.1 Test #4 and FCS_IPSEC_EXT.1.

## 8 CAVP MAPPING

### 8.1 TOE MODELS AND CRYPTOGRAPHIC OPERATIONAL ENVIRONMENT

This section presents a detailed listing of each card supplying cryptographic functionality and its associate cryptographic operational environment (OE).

#### Cryptographic Operational Environment (OE) for TOE hardware models

Appliance Series	Appliance Model	Operational Environment	Microarchitecture
TZ	TZ 670	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570W	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 570P	Marvell CN9130	Quad Core Armv8 Cortex-A72
	TZ 470	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 470W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 370W	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270	Marvell 88F7040	Quad core Armv8 Cortex-A72
	TZ 270W	Marvell 88F7040	Quad core Armv8 Cortex-A72
NSa	NSa 2700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 3700	Marvell CN9130	Quad Core Armv8 Cortex-A72
	NSa 4700	Intel Xeon D-2123IT	Skylake
	NSa 5700	Intel Xeon D-2123IT	Skylake
	NSa 6700	Intel Xeon D-2123IT	Skylake
NSsp	NSsp 10700	Intel Xeon D-2166NT	Skylake
	NSsp 11700	Intel Xeon D-2166NT	Skylake
	NSsp 13700	Intel Xeon D-2187NT	Skylake

#### Cryptographic Operational Environment (OE) for TOE Virtual Appliance

Appliance Series	Appliance Model	Operational Environment
NSv	NSv 270	ESXi 7.0 and 8.0 on Dell PowerEdge R640 (Running on Intel Xeon Silver 4208 (Cascade Lake))
	NSv 470	
	NSv 870	

### 8.2 OPERATIONAL ENVIRONMENT OF THE ALGORITHM IMPLEMENTATION

This section presents a detailed listing of each algorithm listing to include the name and the OE.

Algorithm	Cert #	Name	Operating Environment
AES	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)			
HMAC	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)			
DRBG	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)			
RSA	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583		Intel Xeon D-2123IT

Algorithm	Cert #	Name	Operating Environment
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
			SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
ECDSA	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)			
SHS	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)
SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)			
KAS-FFC-SSC	A5110	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Marvell 88F7040
			Marvell CN9130
	A2583	<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Intel Xeon D-2123IT
			Intel Xeon D-2166NT
			Intel Xeon D-2187NT

Algorithm	Cert #	Name	Operating Environment
	A4982	<a href="#">SonicOS/X 7.0 for NSv Series</a>	<p>SonicOS/X 7.0.1 running on ESXi 7.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)</p> <p>SonicOS/X 7.0.1 running on ESXi 8.0 on Dell PowerEdge R640 on Intel Xeon Silver 4208 (Cascade Lake)</p>

### 8.3 CERTIFICATE(S) TABLE

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

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
FCS_CKM.1	RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	RSA KeyGen (FIPS186-4) Moduli: 2048, 3072, 4096	A4982
	ECC schemes using "NIST curves" [selection: P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
	FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526]	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A2583
<a href="#">SonicOS/X 7.0 for NSv Series</a>		Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A4982	

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
FCS_CKM.1.1.1/IKE	FIPS PUB 186-4, "Digital Signature Standard (DSS)," Appendix B.3 for RSA schemes	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	RSA FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	RSA FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	RSA FIPS PUB 186-4 Key Generation (2048-bit, 3072-bit, 4096-bit)	A4982
	FIPS PUB 186-4, "Digital Signature Standard (DSS)," Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-384 and [P-256, P-521]	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	ECDSA KeyGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA KeyVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
	FFC Schemes using "safe-prime" groups that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526]	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	Safe Primes Key Generation Safe Prime Groups: modp2048  Safe Primes Key Verification Safe Prime Groups: modp2048	A4982



SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
FCS_CKM.2	RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1"	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	No NIST CAVP, CCTL has performed all assurance/evaluation activities and documented in the ETR and AAR accordingly.	N/A. This testing was performed in conjunction with FTP_TRP.1/Admin Test #1 and FTP_ITC.1 Test #1 to demonstrate correct operation.
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>		
		<a href="#">SonicOS/X 7.0 for NSv Series</a>		
	Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	KAS-ECC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: P-256, P-384, P-521	A4982
	FFC Schemes using "safe-prime" groups that meet the following: "NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment collaborative Protection Profile for Network Devices v2.2e, 23-March-2020 Page 57 of 174 Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 3526].	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	KAS-FFC-SSC Sp800-56Ar3 Domain Parameter Generation Methods: modp-2048	A4982
FCS_COP.1/ DataEncryption	AES used in [CBC, GCM] mode and cryptographic key sizes [128 bits, 192 bits, 256 bits]	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256  AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256  AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	AES-CBC Direction: Decrypt, Encrypt Key Length: 128, 192, 256	A4982

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
			AES-GCM Direction: Decrypt, Encrypt Key Length: 128, 256	
FCS_COP.1/ SigGen	For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096  RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096  RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	RSA SigGen (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096  RSA SigVer (FIPS186-4) Signature Type: PKCS 1.5 Moduli: 2048, 3072, 4096	A4982
FCS_COP.1/ Hash	For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384]; ISO/IEC 14888-3, Section 6.4	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	ECDSA SigGen (FIPS186-4) Curve: P-256, P-384, P-521  ECDSA SigVer (FIPS186-4) Curve: P-256, P-384, P-521	A4982
FCS_COP.1/ Hash	[SHA-1, SHA2-256, SHA2-384, SHA2-512] and message digest sizes [160, 256, 384, 512] bits	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	SHA-1 SHA2-256 SHA2-384 SHA2-512	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	SHA-1 SHA2-256 SHA2-384 SHA2-512	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	SHA-1 SHA2-256	A4982

SFR	Algorithm in ST	Implementation name	CAVP Alg.	CAVP Cert #
			SHA2-384 SHA2-512	
FCS_COP.1/ KeyedHash	[HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512] and cryptographic key sizes [key size (in bits) used in HMAC] and message digest sizes [160, 256, 384, 512] bits	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	HMAC-SHA-1, HMAC-SHA2- 256, HMAC-SHA2-384, HMAC-SHA2-512	A4982
FCS_RBG_EXT.1	Hash_DRBG	<a href="#">SonicOS/X 7.0.1 for TZ, NSA Series</a>	Hash DRBG SHA2-256	A5110
		<a href="#">SonicOS/X 7.0.1 for NSa, NSsp Series</a>	Hash DRBG SHA2-256	A2583
		<a href="#">SonicOS/X 7.0 for NSv Series</a>	Hash DRBG SHA2-256	A4982

## 9 CONCLUSION

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