

Northrop Grumman Corporation

Ad Noctem Connect 2.3

FIPS 140-3 Non-Proprietary Security Policy

Version: 2.3

Prepared for:

Northrop Grumman Corporation
2980 Fairview Park Drive
Falls Church, VA 22042
USA
Alex Barron
alex.barron@ngc.com

Prepared by:

TERON LABS

www.teronlabs.com

Table of Contents

1 General	5
1.1 Overview	5
1.2 Security Levels.....	5
2 Cryptographic Module Specification	6
2.1 Description	6
2.2 Tested and Vendor Affirmed Module Version and Identification.....	7
2.3 Excluded Components	8
2.4 Modes of Operation	9
2.5 Algorithms	9
2.6 Security Function Implementations	13
2.7 Algorithm Specific Information	15
2.8 RBG and Entropy	16
2.9 Key Generation	16
2.10 Key Establishment.....	16
2.11 Industry Protocols	16
3 Cryptographic Module Interfaces	17
3.1 Ports and Interfaces	17
4 Roles, Services, and Authentication.....	18
4.1 Authentication Methods.....	18
4.2 Roles	18
4.3 Approved Services.....	18
4.4 Non-Approved Services.....	22
4.5 External Software/Firmware Loaded.....	23
5 Software/Firmware Security.....	24
5.1 Integrity Techniques.....	24
5.2 Initiate on Demand	24
6 Operational Environment	25
6.1 Operational Environment Type and Requirements	25
6.2 Configuration Settings and Restrictions.....	25
7 Physical Security	26
8 Non-Invasive Security	27
9 Sensitive Security Parameters Management	28
9.1 Storage Areas.....	28
9.2 SSP Input-Output Methods	28

9.3 SSP Zeroization Methods	28
9.4 SSPs	29
10 Self-Tests	35
10.1 Pre-Operational Self-Tests	35
10.2 Conditional Self-Tests	35
10.3 Periodic Self-Test Information	36
10.4 Error States	37
10.5 Operator Initiation of Self-Tests	38
11 Life-Cycle Assurance	39
11.1 Installation, Initialization, and Startup Procedures	39
11.2 Administrator Guidance	39
11.3 Non-Administrator Guidance	39
11.4 End of Life	42
12 Mitigation of Other Attacks	43

List of Tables

Table 1: Security Levels	5
Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets).....	8
Table 3: Tested Operational Environments - Software, Firmware, Hybrid.....	8
Table 4: Modes List and Description	9
Table 5: Approved Algorithms.....	11
Table 6: Vendor-Affirmed Algorithms	12
Table 7: Non-Approved, Not Allowed Algorithms	13
Table 8: Security Function Implementations.....	15
Table 9: Entropy Certificates	16
Table 10: Entropy Sources	16
Table 11: Ports and Interfaces	17
Table 12: Roles	18
Table 13: Approved Services	22
Table 14: Non-Approved Services.....	22
Table 15: Storage Areas.....	28
Table 16: SSP Input-Output Methods	28
Table 17: SSP Zeroization Methods.....	29
Table 18: SSP Table 1.....	30
Table 19: SSP Table 2.....	34
Table 20: Pre-Operational Self-Tests	35
Table 21: Conditional Self-Tests.....	36
Table 22: Pre-Operational Periodic Information	36
Table 23: Conditional Periodic Information.....	37
Table 24: Error States	38

List of Figures

No table of figures entries found.

1 General

1.1 Overview

The Northrup Grumman Ad Noctem Connect (the module) provides secure VPN connectivity for its users. The module enables users to communicate securely over untrusted networks. The module is a VPN Client application that uses a web-based interface for its configuration. The VPN client allows users to connect to VPN Gateway endpoints using IKEv2/IPsec.

The module will be delivered pre-installed on Ad Noctem GB-100 Hardware platform. The module is unsupported on any other hardware platform. The certificate authority (CA) provides the module with x509 certificates. The module can connect to a VPN gateway that provides IKEv2/IPsec support. The gateway will need to be configured to support the parameters required by the module.

1.2 Security Levels

The Ad Noctem Connect Module is validated at the FIPS 140-3 section levels shown below.

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	N/A
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	N/A
	Overall Level	1

Table 1: Security Levels

The module has an overall security level of 1.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The Ad Noctem Connect 2.3 (the module) is a VPN Client application. It provides secure communication over untrusted networks. To provide this security, the module implements IKEv2/IPsec. It can be configured over a local, web-based interface.

Module Type: Software

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary:

The cryptographic boundary delimits the following software packages:

- IKE Daemon (strongSwan 5.9.11)
- IKE cryptographic plug-in/library (OpenSSL 1.1.1l)
- IPsec support (Linux kernel 5.14.21)

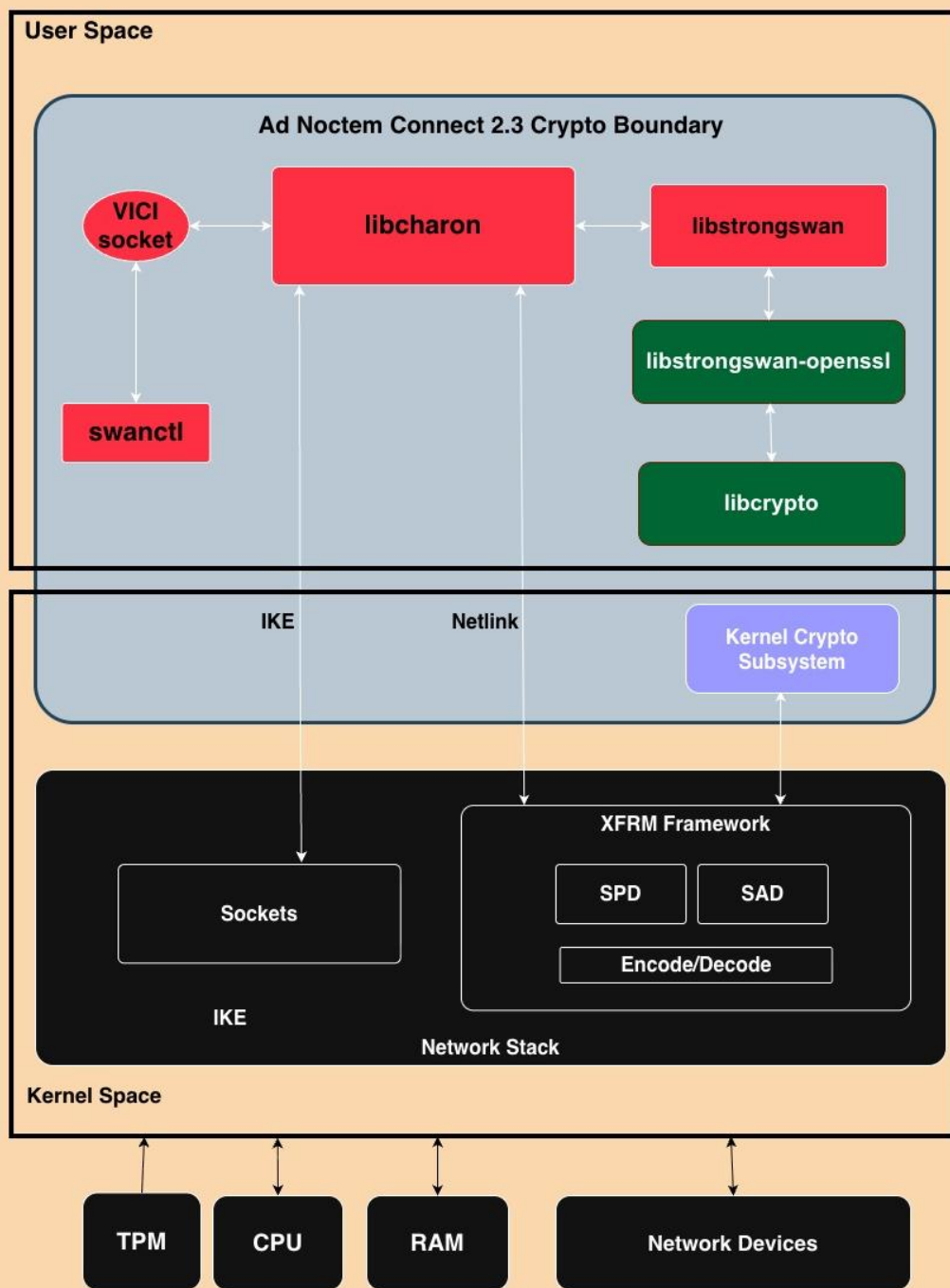
The above packages are bundled as a single package that comes preinstalled on the Ad Noctem GB-100 Hardware.

The cryptographic boundary is depicted in the block diagram. The IKE Daemon components are highlighted in red, the IKE cryptographic plug-in/library component is highlighted in green, and the IPsec support component is highlighted in blue.

Tested Operational Environment's Physical Perimeter (TOEPP):

The TOEPP of the module is defined as the Ad Noctem GB-100 Hardware platform running Ad Noctem Connect OS 2.3.

TOEPP: Ad Noctem GB-100 Hardware



Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):

Package or File Name	Software/ Firmware Version	Features	Integrity Test
/etc/adnoctem-release /usr/lib/adnoctem-fipscheck.service.sh /usr/lib/adnoctem/crypto-boundary.txt StrongSwan: /usr/sbin/ipsec /usr/lib/ipsec/_updown /usr/lib/ipsec/charon /usr/lib/ipsec/pool /usr/lib/ipsec/starter /usr/lib/ipsec/stroke /usr/lib/ipsec/_fipscheck /usr/lib64/ipsec/libstrongswan.so.0.0.0 /usr/lib64/ipsec/libcharon.so.0.0.0 /usr/sbin/swanctl OpenSSL: /usr/lib64/ipsec/plugins/libstrongswan-openssl.so /usr/lib64/libcrypto.so.1.1 Linux Kernel: /boot/vmlinux-5.14.21-150400.24.173-default *.ko files in /lib/modules/5.14.21-150400.24.173- default/kernel/arch/arm64/crypto/ *.ko files in /lib/modules/5.14.21-150400.24.173- default/kernel/crypto/	StrongSwan 5.9.11 Kernel 5.14.21 OpenSSL 1.1.1l	N/A	HMAC- SHA2-256

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

Tested Operational Environments – Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
SUSE Enterprise Linux Micro OS 5.3	GB-100	Arm Cortex- A53	No		2.3

Table 3: Tested Operational Environments - Software, Firmware, Hybrid

2.3 Excluded Components

The module does not have excluded components.

2.4 Modes of Operation

Modes List and Description:

The module supports both Approved and Non-Approved modes of operation, as described in the table below. Once operational, the module enters one of these modes depending on the service that has been executed. Upon execution of a service, the Approved Service Indicator can be queried to determine whether the module has executed an Approved Service and is therefore operating in the Approved mode.

Details of the Approved Service Indicator can be found in section 11.2.

Mode Name	Description	Types of Services	Status Indicator
Approved	When the module is operating using approved services.	Approved	Approved Service Indicators
Non-Approved	When the module is operating using non-approved services.	Non-Approved	Approved Service Indicators

Table 4: Modes List and Description

Mode Change Instructions and Status:

The module exits the approved mode when an operator configures the module to use non-approved algorithms.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A7623, A7624	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A7624	Key Length - 128, 256 Tag Length - 128, 64, 96 IV Length - IV Length: 64, 96, 104 Payload Length - Payload Length: 256 AAD Length - AAD Length: 64, 96	SP 800-38C
AES-CTR	A7623	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - No	SP 800-38A

Algorithm	CAVP Cert	Properties	Reference
		Incremental Counter - Yes Counter Tests Performed - Yes	
AES-ECB	A7623	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
AES-GCM	A7623	Direction - Decrypt, Encrypt IV Generation - External IV Generation Mode - 8.2.1, 8.2.2 Key Length - 128, 192, 256 Tag Length - 128, 64, 96 IV Length - IV Length: 96 Payload Length - Payload Length: 120, 128, 248, 256, Payload Length: 160-65536 Increment 8 AAD Length - AAD Length: 64, 96	SP 800-38D
AES-GCM	A7624	Direction - Decrypt, Encrypt IV Generation - External IV Generation Mode - 8.2.1 Key Length - 128, 256 Tag Length - 128, 64, 96 IV Length - IV Length: 96 Payload Length - Payload Length: 160-65536 Increment 8 AAD Length - AAD Length: 64, 96	SP 800-38D
Counter DRBG	A7623	Prediction Resistance - No Supports Reseed - No Mode - AES-256 Derivation Function Enabled - No Additional Input - Additional Input: 0 Entropy Input - Entropy Input: 384 Nonce - Nonce: 0 Personalization String Length - Personalization String Length: 0 Returned Bits - 256	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A7623	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A7623	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A7623	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A7623	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA2-256	A7623, A7624	MAC - MAC: 256 Key Length - Key Length: 256-512 Increment 8	FIPS 198-1
HMAC-SHA2-384	A7623, A7624	MAC - MAC: 256-384 Increment 8 Key Length - Key Length: 256-1024 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-512	A7623, A7624	MAC - MAC: 256-512 Increment 8 Key Length - Key Length: 256-1024 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A7623	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Hash Function Z - SHA2-512 Scheme - onePassDh - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A7623	Domain Parameter Generation Methods - MODP- 2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF IKEv2 (CVL)	A7623	Initiator Nonce Length - Initiator Nonce Length: 128-2048 Increment 8 Responder Nonce Length - Responder Nonce Length: 128-2048 Increment 8 Diffie-Hellman Shared Secret Length - Diffie- Hellman Shared Secret Length: 224-8192 Increment 8 Derived Keying Material Length - Derived Keying Material Length: 256-16384 Increment 8 Derived Keying Material Child Length - Derived Keying Material Child Length: 160-16384 Increment 8 Hash Algorithm - SHA2-256, SHA2-384, SHA2- 512	SP 800-135 Rev. 1
RSA SigGen (FIPS186-5)	A7623	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5	FIPS 186-5
RSA SigVer (FIPS186-5)	A7623	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5 Fixed Public Exponent - 010001 Public Exponent Mode - fixed	FIPS 186-5
Safe Primes Key Generation, Safe Primes Key Verification	A7623	Safe Prime Groups - MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192	SP 800-56A Rev. 3
SHA-1, SHA2-256, SHA2- 384, SHA2-512	A7624, A7623	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 5: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	SP800-133r2 Section 4, Example 1

Table 6: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

The module does not implement non-approved algorithms allowed in the approved mode of operation.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

The module does not implement non-approved algorithms allowed in the approved mode of operation with no security claimed.

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
Triple DES	Encryption Algorithms
CAST-128	Encryption Algorithms
Blowfish	Encryption Algorithms
Camellia	Encryption Algorithms
AES-GMAC with null encryption	Authenticated Encryption (AEAD) Algorithms
Camellia with CCM	Authenticated Encryption (AEAD) Algorithms
chacha20poly1305	Authenticated Encryption (AEAD) Algorithms
HMAC-MD5, SHA1	Integrity Algorithms
AES XCBC	Integrity Algorithms
AES CMAC	Integrity Algorithms
AES-GMAC for authentication	Integrity Algorithms
Diffie Hellman with modp768, modp1024, modp1536	Diffie Hellman
Diffie Hellman with ecp224, ecp224bp, ecp256bp, ecp384bp, ecp512bp, curve25519, x25519, curve448, x448	EC
RSA Signatures with MD5, SHA-1, SHA-224, SHA-3	Signature Algorithms
ECDSA Signatures with non-NIST curves	Signature Algorithms
EdDSA	Signature Algorithms

Table 7: Non-Approved, Not Allowed Algorithms

Configuring any of the above algorithms results in the module entering the non-approved mode of operation.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
IKEv2 KAS-ECC	KAS-Full	Key agreement scheme implemented for IKEv2 using KAS-ECC and KDF IKEv2	IG:IG D.F Scenario 2 path (2), split Key Confirmation:No Key derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology provides between 128 and 256 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A7623) KDF IKEv2: (A7623)
IKEv2 KAS-FFC	KAS-Full	Key agreement scheme implemented for IKEv2 using KAS-FFC and KDF IKEv2	IG:IG D.F Scenario 2 path (2), split Key Confirmation:No Key derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology provides between 128 and 256 bits of security strength	KAS-FFC-SSC Sp800-56Ar3: (A7623) KDF IKEv2: (A7623)
Block Cipher IKEv2	BC-UnAuth BC-Auth	Data encryption in IKEv2 protocol		AES-CBC: (A7623) AES-CCM: (A7623) AES-CTR: (A7623) AES-ECB: (A7623) AES-GCM: (A7623) HMAC-SHA2-256: (A7623) HMAC-SHA2-384: (A7623) HMAC-SHA2-512: (A7623) SHA2-256: (A7623) SHA2-384: (A7623) SHA2-512: (A7623)

Name	Type	Description	Properties	Algorithms
Block Cipher IPsec	BC-UnAuth BC-Auth	Data encryption for IPSec tunneling		AES-CBC: (A7624) AES-CCM: (A7624) AES-GCM: (A7624) HMAC-SHA2- 256: (A7624) HMAC-SHA2- 384: (A7624) HMAC-SHA2- 512: (A7624) SHA2-256: (A7624) SHA2-384: (A7624) SHA2-512: (A7624)
IKEv2 KDF	KAS-135KDF	IKEv2 key derivation		KDF IKEv2: (A7623)
IKEv2 KTS		transport via IKEv2 Encryption	Standard:SP800-38F IG D.G:Approved key wrapping Caveat:Key establishment methodology provides between 128 and 256 bits of security strength	AES-CBC: (A7623) AES-CCM: (A7623) AES-CTR: (A7623) AES-ECB: (A7623) AES-GCM: (A7623) HMAC-SHA2- 256: (A7623) HMAC-SHA2- 384: (A7623) HMAC-SHA2- 512: (A7623) SHA2-256: (A7623) SHA2-384: (A7623) SHA2-512: (A7623)
Generate random	DRBG	Generate random bits using the DRBG.	Standard:SP800-90Ar1	Counter DRBG: (A7623)
KAS Key Pair Generation	AsymKeyPair- KeyGen CKG	Key pair generation for KAS ephemeral key pairs		Counter DRBG: (A7623) ECDSA KeyGen (FIPS186-5): (A7623) ECDSA KeyVer (FIPS186-5):

Name	Type	Description	Properties	Algorithms
				(A7623) Safe Primes Key Generation: (A7623) Safe Primes Key Verification: (A7623) CKG: ()
Message Authentication	MAC	Message authentication for in IKEv2		HMAC-SHA2- 256: (A7623) HMAC-SHA2- 384: (A7623) HMAC-SHA2- 512: (A7623) SHA2-256: (A7623) SHA2-384: (A7623) SHA2-512: (A7623)
ECDSA Signature Generation	DigSig-SigGen	Digital signature generation		ECDSA SigGen (FIPS186-5): (A7623)
ECDSA Signature Verification	DigSig-SigVer	Digital signature verification		ECDSA SigVer (FIPS186-5): (A7623)
RSA Signature Generation	DigSig-SigGen	Digital signature generation		RSA SigGen (FIPS186-5): (A7623)
RSA Signature Verification	DigSig-SigVer	Digital signature verification		RSA SigVer (FIPS186-5): (A7623)

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

AES-GCM IV uniqueness:

The module employs AES-GCM for IKEv2 as per RFC 5282, and AES-GCM for IPsec as per RFC4106. The module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED from which the AES-GCM encryption keys are derived. The construction of AES-GCM IVs is identical for both IKEv2 encryption and IPsec encryption.

To ensure IV uniqueness as per IG C.H Scenario 1, the module employs proactive re-keying of Security Associations (SAs) triggered by packet, byte or time limits to establish new keys and reset IV counters before overflow can occur.

2.8 RBG and Entropy

The module relies on entropy provided from the Infineon Trusted Platform Module 2.0 Entropy Source (E142), which is an SP 800-90B entropy source located within the TOEPP, but outside the cryptographic boundary.

The entropy source contains 7.51729 per 8-bit output and is used to seed the DRBG with 256 bits of entropy input and 128 bits for the nonce for a total security strength of 192-bits.

The module claims I.G. 9.3.A scenario 1. (a) and includes the caveat: *The module generates SSPs (e.g., keys) whose strengths are modified by available entropy.*

Cert Number	Vendor Name
Infineon	E142

Table 9: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Infineon Trusted Platform Module 2.0 Entropy Source	Physical	SLB 9672	8 bits	7.51729	N/A

Table 10: Entropy Sources

2.9 Key Generation

The module generates KAS-ECC and KAS-FFC key pairs compliant with SP800-56Arev3. Random values are generated in accordance with SP 800-133 Rev 2. The DRBG output is provided directly as input to the key generation algorithm implementations.

2.10 Key Establishment

The module implements key agreement methods in compliance with IG D.F Scenario 2 path (2) by separate testing of the computation of the shared secrets and the key derivation. The agreement methods implemented are:

- SP 800-56Arev3: (Cofactor) Ephemeral Unified Model scheme with IKEv2 KDF.
- SP 800-56Arev3: dhEphem scheme with IKEv2 KDF.

2.11 Industry Protocols

The module supports IPsec per RFC 4303 and IKEv2 per RFC 7296 industrial protocols. No parts of IKE protocols, other than the KDFs, have been tested by the CAVP and CMVP.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The module supports Data Input, Data Output, Control Input and Status Output interfaces. The module does not support a Control Output or Power Interface.

The IKE Daemon (/usr/lib/ipsec/charon) loads control input from configuration files stored in /etc/swanctl/swanctl.conf. The IKE Daemon also handles Data input/output for IKEv2/IPsec protocol data. The swanctl component provides control input and status output interfaces via CLI commands.

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	Keys via configuration file. IKEv2/IPsec protocol data.
N/A	Data Output	IKEv2/IPsec protocol data
N/A	Control Input	Configuration data via configuration file. CLI command invocation.
N/A	Status Output	Status messages of the IKEv2 daemon and commands. Status of the protocol message processing.

Table 11: Ports and Interfaces

4 Roles, Services, and Authentication

4.1 Authentication Methods

The module does not implement operator authentication. The Crypto-Officer role is implicitly assumed when the module is initialized.

N/A for this module.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	

Table 12: Roles

The module does not support multiple concurrent operators.

4.3 Approved Services

The Approved Services are listed in the table below. Further details of the Approved Service Indicators can be found in section 11.2

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Start IKEv2 daemon	Execution of the IKEv2 daemon.	N/A	N/A	Success/Failure	None	Crypto Officer
Connect IKEv2	Load the VPN configuration and initiate the IKEv2 connection.	LIST-SAS-IKE	Pre-shared keys or asymmetric key pairs Configuration data	Success/failure	IKEv2 KAS-ECC IKEv2 KAS-FFC Block Cipher IKEv2 IKEv2 KDF Generate random KAS Key Pair Generation Message Authentication ECDSA Signature Generation ECDSA Signature Verification RSA Signature Generation RSA Signature	Crypto Officer - DRBG Seed: R - DRBG Entropy Input: R - DRBG V Value: R,W - DRBG Key Value: R,W - ECDH KAS Private Key: G,W,R - ECDH KAS Public Key: G,W,R - ECDH KAS Peer Key: R - DH KAS Private Key: G,W,R - DH KAS Public Key: G,W,R - DH KAS Peer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					Verification IKEv2 KTS	Key: R - SKEYSEED: G,W,R - IKEv2 Derivation Key: G,W,R - IKEv2 Encryption keys: G,W,R - IKEv2 Authentication key: G,W,R - RSA Public Key: R - RSA Private Key: R - RSA Peer Public Key: R - ECDSA Public Key: R - ECDSA Private Key: R - ECDSA Peer Public Key: R - PSK: R - IKEv2 AES- GCM IV: G,W,R
IPsec Tunnelling	Encryption and decryption of user data over the IPsec tunnel	LIST- SAS-ESP	Tunnel encryption keys, Plaintext traffic, Encrypted traffic	Encrypted traffic, decrypted traffic	Block Cipher IPsec	Crypto Officer - IPsec Encryption Key: G,E - IPsec Authentication Key: G - IPsec AES- GCM IV: G,W,R
Teardown connection	Teardown of Security Associations used for communication.	N/A	Connection identifier	Success/Failure	None	Crypto Officer - ECDH KAS Private Key: Z - ECDH KAS Public Key: Z - ECDH KAS Peer Key: Z - DH KAS Private Key: Z - DH KAS Public Key: Z - DH KAS Peer Key: Z - SKEYSEED: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- IKEv2 Derivation Key: Z - IKEv2 Encryption keys: Z - IKEv2 Authentication key: Z
Terminate IKEv2 daemon	Termination of IKEv2 daemon resulting in the clearing of all CSPs.	N/A	None	Success/Failure	None	Crypto Officer - DRBG Seed: Z - DRBG Entropy Input: Z - DRBG V Value: Z - DRBG Key Value: Z - ECDH KAS Private Key: Z - ECDH KAS Public Key: Z - ECDH KAS Peer Key: Z - DH KAS Private Key: Z - DH KAS Public Key: Z - DH KAS Peer Key: Z - SKEYSEED: Z - IKEv2 Derivation Key: Z - IKEv2 Encryption keys: Z - IKEv2 Authentication key: Z - RSA Public Key: Z - RSA Private Key: Z - RSA Peer Public Key: Z - ECDSA Public Key: Z - ECDSA Private Key: Z - ECDSA Peer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Public Key: Z - PSK: Z
Perform Self-tests	On-demand execution of self-tests by power cycling				None	Crypto Officer
Zeroization	Zeroization of SSPs via power cycling	N/A			None	Crypto Officer - DRBG Seed: Z - DRBG Entropy Input: Z - DRBG V Value: Z - DRBG Key Value: Z - ECDH KAS Private Key: Z - ECDH KAS Public Key: Z - ECDH KAS Peer Key: Z - DH KAS Private Key: Z - DH KAS Public Key: Z - DH KAS Peer Key: Z - SKEYSEED: Z - IKEv2 Derivation Key: Z - IKEv2 Encryption keys: Z - IKEv2 Authentication key: Z - RSA Public Key: Z - RSA Private Key: Z - RSA Peer Public Key: Z - ECDSA Public Key: Z - ECDSA Private Key: Z - ECDSA Peer Public Key: Z - PSK: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Show Status	Query the status of the current connections.	N/A	swanctl --list-sas		None	Unauthenticated
Show Version	Output the module name and version	N/A	cat /etc/adnoctem-release		None	Unauthenticated

Table 13: Approved Services

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Connect IKEv2 (non-approved)	Load the VPN configuration and initiate the IKEv2 connection (using non-approved algorithms)	Diffie Hellman with modp768, modp1024, modp1536 Diffie Hellman with ecp224, ecp224bp, ecp256bp, ecp384bp, ecp512bp, curve25519, x25519, curve448, x448 RSA Signatures with MD5, SHA-1, SHA-224, SHA-3 ECDSA Signatures with non-NIST curves EdDSA Triple DES CAST-128 Blowfish Camellia AES-GMAC with null encryption Camellia with CCM chacha20poly1305 HMAC-MD5, SHA1 AES XCBC AES CMAC AES-GMAC for authentication	Crypto Officer
IPsec Tunnelling (non-approved)	Encryption and decryption of user data over the IPsec tunnel (using non-approved algorithm)	Triple DES CAST-128 Blowfish Camellia AES-GMAC with null encryption Camellia with CCM chacha20poly1305 HMAC-MD5, SHA1 AES XCBC AES CMAC AES-GMAC for authentication	Crypto Officer

Table 14: Non-Approved Services

4.5 External Software/Firmware Loaded

The module does not support external Software/Firmware Loading.

5 Software/Firmware Security

5.1 Integrity Techniques

All software components within the cryptographic boundary are verified using an Approved integrity technique implemented within the cryptographic module itself. The module calculates a HMAC-SHA2-256 value calculated at run time with the value that was computed at build time, for each of the files within the cryptographic. If the MAC values do not match, the test fails, and the module enters the error state. The module's integrity check is performed automatically at module instantiation, before the IKE daemon is initiated and without action from the module operator.

5.2 Initiate on Demand

The crypto-officer can initiate the pre-operational self-tests on demand by re-instantiating the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

The module is comprised of software which executes in a modifiable operational environment.

The module has control over its own SSPs. Any additional SSPs loaded into the module are not available to other processes.

The supported operating environments provide process isolation, with resource and memory protection that ensures that SSPs can be accessed and modified the module and trusted parts of the operational environment.

6.2 Configuration Settings and Restrictions

The module is delivered pre-installed and configured with the TOEPP as described in Section 2.2 and Section 11.1. The OS is configured with a read-only file system and by default only allows for one instance of the module to execute at any given time.

All operation and configuration of the module must adhere to the guidance detailed in Section 11.

7 Physical Security

The module is comprised of Software only. Physical security is not applicable.

8 Non-Invasive Security

The module does not implement any non-invasive security mechanisms.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	SSPs are stored temporarily for use by the module services while operational.	Dynamic

Table 15: Storage Areas

The module does not perform persistent storage of SSPs. The SSPs are temporarily stored in the RAM in plaintext form.

Public and private keys for IKEv2 authentication are stored in the /etc/swanctl/x509 and /etc/swanctl/private directories, which are within the module's physical perimeter, but outside its cryptographic boundary.

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Input: Configuration	File storage within TOEPP	RAM	Plaintext	Manual	Electronic	
Input: IKEv2 protocol	Network interface	RAM	Plaintext	Automated	Electronic	
Output: IKEv2 protocol	RAM	Network interface	Plaintext	Automated	Electronic	
Input: IKEv2 encrypted protocol	Network interface	RAM	Encrypted	Automated	Electronic	Block Cipher IKEv2
Output: IKEv2 encrypted protocol	RAM	Network interface	Encrypted	Automated	Electronic	Block Cipher IKEv2
Input: IPsec encrypted protocol	Network interface	RAM	Encrypted	Automated	Electronic	Block Cipher IPsec
Output: IPsec encrypted protocol	RAM	Network interface	Encrypted	Automated	Electronic	Block Cipher IPsec

Table 16: SSP Input-Output Methods

9.3 SSP Zeroization Methods

When zeroizing SSPs by removing power from the module, the operator must maintain control of the module.

Zeroization Method	Description	Rationale	Operator Initiation
Remove power from the module	The module is powered down.	All SSPs are stored in memory only, and are cleared when power is removed.	Operator removes power from the module.
Close IKEv2 connection	IKEv2 connection is closed	SSPs in memory are overwritten with zeroes, making them irretrievable.	IKEv2 connection is closed.
Terminate IKEv2 daemon	The IKEv2 daemon is terminated.	SSPs in memory are overwritten with zeroes, making them irretrievable.	IKEv2 Daemon is terminated.

Table 17: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
DRBG Seed	Seeding material for DRBG	-	Seed - CSP			Generate random
DRBG Entropy Input	Entropy input for DRBG	256 bits - 192 bits	Entropy Input - CSP			Generate random
DRBG V Value	Internal state values for DRBG	-	DRBG State - CSP	Generate random		Generate random
DRBG Key Value	Internal state value for DRBG	-	DRBG State - CSP	Generate random		Generate random
ECDH KAS Private Key	IKEv2 KAS Ephemeral ECDH Private Key	256-521 bits - 128 to 256 bits	Private Key - CSP	KAS Key Pair Generation		IKEv2 KAS-ECC
ECDH KAS Public Key	IKEv2 KAS Ephemeral ECDH Public Key	256-521 bits - 128 to 256 bits	Public Key - PSP	KAS Key Pair Generation		
ECDH KAS Peer Key	IKEv2 KAS Ephemeral Peer ECDH Public Key	256-521 bits - 128 to 256 bits	Public Key - PSP			IKEv2 KAS-ECC
DH KAS Private Key	IKEv2 KAS Ephemeral DH Private Key	2048-4096 bits - 112 to 152 bits	Private Key - CSP	KAS Key Pair Generation		IKEv2 KAS-FFC
DH KAS Public Key	IKEv2 KAS Ephemeral DH Public Key	2048-4096 bits - 112 to 152 bits	Public Key - PSP	KAS Key Pair Generation		
DH KAS Peer Key	IKEv2 KAS Ephemeral Peer DH Public Key	2048-4096 bits - 112 to 152 bits	Public Key - PSP			IKEv2 KAS-FFC

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
KEYSEED	IKEv2 KAS derived key material	-	Symmetric Key Material - CSP		IKEv2 KAS-ECC IKEv2 KAS-FFC	IKEv2 KDF
IKEv2 Derivation Key	IKEv2 SK_d for derivation of Encryption keys	-	Symmetric Key - CSP		IKEv2 KDF	
IKEv2 Encryption keys	SK_e Key for encryption of IKEv2 messages	-	Symmetric Key - CSP		IKEv2 KDF	Block Cipher IKEv2
IKEv2 Authentication key	SK_e Key for authentication of IKEv2 messages	IKEv2 SK_a Key -	Symmetric Key - CSP		IKEv2 KDF	Block Cipher IKEv2
IPsec Encryption Key	Encryption key for encrypting IPsec packets	-	Symmetric Key - CSP		IKEv2 KDF	Block Cipher IPsec
IPsec Authentication Key	Authentication key for encrypting IPsec packets	-	Symmetric Key - CSP		IKEv2 KDF	Block Cipher IPsec
RSA Public Key	RSA Authentication Public key	- 112 to 152 bits	Public Key - PSP			
RSA Private Key	RSA Authentication Private key	- 112 to 152 bits	Private Key - CSP			RSA Signature Generation
RSA Peer Public Key	RSA Peer Authentication Key	- 112 to 152 bits	Public Key - PSP			RSA Signature Verification
ECDSA Public Key	ECDSA Authentication Public key	256-521 bits - 128 to 256 bits	Public Key - PSP			
ECDSA Private Key	ECDSA Authentication Private key	256-521 bits - 128 to 256 bits	Private Key - CSP			ECDSA Signature Generation
ECDSA Peer Public Key	ECDSA Peer Authentication Public Key	256-521 bits - 128 to 256 bits	Public Key - PSP			ECDSA Signature Verification
PSK	Pre-shared Authentication key	-	Symmetric Key - CSP			Message Authentication
IKEv2 AES-GCM IV	IV	96 bits -	Public Key - PSP			Block Cipher IKEv2
IPsec AES-GCM IV	IV	96 bits -	Public Key - PSP			Block Cipher IPsec

Table 18: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
DRBG Seed		RAM:Plaintext	Ex	Remove power from the module Terminate IKEv2 daemon	Derived From:DRBG Entropy Input
DRBG Entropy Input		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	
DRBG V Value		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	DRBG Seed:Derived From DRBG Key Value:Paired With
DRBG Key Value		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	DRBG Seed:Derived From DRBG V Value:Paired With
ECDH KAS Private Key		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Paired With:ECDH KAS Public Key
ECDH KAS Public Key	Output: IKEv2 protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Paired With:ECDH KAS Private Key
ECDH KAS Peer Key	Input: IKEv2 protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	
DH KAS Private Key		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	Paired With:DH KAS Public Key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Close IKEv2 connection	
DH KAS Public Key	Output: IKEv2 protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Paired With:DH KAS Private Key
DH KAS Peer Key	Input: IKEv2 protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	
SKEYSEED		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Derived From:ECDH KAS Private Key, ECDH KAS Peer Key, DH KAS Private Key, DH KAS Peer Key
IKEv2 Derivation Key		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Derived From:SKEYSEED
IKEv2 Encryption keys		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Derived From:IKEv2 Derivation Key
IKEv2 Authentication key		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Derived From:IKEv2 Derivation Key
IPsec Encryption Key		RAM:Plaintext		Remove power from the module Terminate	Derived From:IKEv2 Derivation Key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				IKEv2 daemon Close IKEv2 connection	
IPsec Authentication Key		RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	Derived From:IKEv2 Derivation Key
RSA Public Key	Input: Configuration Output: IKEv2 encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	Paired With:RSA Private Key
RSA Private Key	Input: Configuration	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	Paired With:RSA Public Key
RSA Peer Public Key	Input: IKEv2 encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	
ECDSA Public Key	Input: Configuration Output: IKEv2 encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	Paired With:ECDSA Private Key
ECDSA Private Key	Input: Configuration	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon	Paired With:ECDSA Public Key
ECDSA Peer Public Key	Input: IKEv2 encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	
PSK	Input: Configuration	RAM:Plaintext		Remove power from the module	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Terminate IKEv2 daemon	
IKEv2 AES-GCM IV	Input: IKEv2 encrypted protocol Output: IKEv2 encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	
IPsec AES-GCM IV	Input: IPsec encrypted protocol Output: IPsec encrypted protocol	RAM:Plaintext		Remove power from the module Terminate IKEv2 daemon Close IKEv2 connection	

Table 19: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256	HMAC-SHA2-256	KAT	SW/FW Integrity	Module becomes operational	HMAC-SHA2-256 performed on all files within the crypto boundary and compared to known digests.

Table 20: Pre-Operational Self-Tests

The pre-operation self-tests are performed immediately upon the module powering on. The module is not operational until all pre-operational self-tests have passed.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-256 (A7623)	-	KAT	CAST	Module started	Hash	Module load
SHA2-512 (A7623)	-	KAT	CAST	Module started	Hash	Module load
IKEv2 KDF (A7623)	SHA2-256	KAT	CAST	Module started	Derive	Module load
HMAC-SHA2-256 (A7623)	160-bit key	KAT	CAST	Module started	Compute MAC	Module load
AES-ECB (A7623) Encrypt	128-bit key	KAT	CAST	Module started	Encrypt	Module load
AES-ECB (A7623) Decrypt	128-bit key	KAT	CAST	Module started	Decrypt	Module load
AES-GCM (A7623) Encrypt	256-bit key	KAT	CAST	Module started	Encrypt	Module load
AES-GCM (A7623) Decrypt	256-bit key	KAT	CAST	Module started	Decrypt	Module load
Counter DRBG (A7623)	AES256	KAT	CAST	Module started	Instantiate, Reseed, Generate	Module load
ECDSA SigGen (FIPS186-5) (A7623)	SHA2-256, secp256k1	KAT	CAST	Module started	Sign	Module load
ECDSA SigVer (FIPS186-5) (A7623)	SHA2-256, secp256k1	KAT	CAST	Module started	Verify	Module load
RSA SigGen (FIPS186-5) (A7623)	SHA2-256, 2048	KAT	CAST	Module started	Sign	Module load
RSA SigVer (FIPS186-5) (A7623)	SHA2-256, 2048	KAT	CAST	Module started	Verify	Module load

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KAS-ECC-SSC Sp800-56Ar3 (A7623)	P-256	KAT	CAST	Module started	Compute Shared Secret	Module load
KAS-FFC-SSC Sp800-56Ar3 (A7623)	2048-bit	KAT	CAST	Module started	Compute Shared secret	Module load
ECDH PCT		PCT	PCT	Connection successful	Sign/Verify	Key pair generated
DH PCT		PCT	PCT	Connection successful	Compute public key	Key pair generated
AES-CBC (A7624) Encrypt	128,192, 256	KAT	CAST	Module started	Encrypt	Module load
AES-CBC (A7624) Decrypt	128,192, 256	KAT	CAST	Module started	Decrypt	Module load
AES-GCM (A7624) Encrypt	128,192, 256	KAT	CAST	Module started	Encrypt	Module load
AES-GCM (A7624) Decrypt	128,192, 256	KAT	CAST	Module started	Decrypt	Module load
AES-CCM (A7624) Encrypt	128,192, 256	KAT	CAST	Module started	Encrypt	Module load
AES-CCM (A7624) Decrypt	128,192, 256	KAT	CAST	Module started	Decrypt	Module load
HMAC-SHA2-256 (A7624)		KAT	CAST	Module started	Compute MAC	Module load
HMAC-SHA2-384 (A7624)		KAT	CAST	Module started	Compute MAC	Module load
HMAC-SHA2-512 (A7624)		KAT	CAST	Module started	Compute MAC	Module load
SHA2-256 (A7624)		KAT	CAST	Module started	Hash	Module load
SHA2-384 (A7624)		KAT	CAST	Module started	Hash	Module load
SHA2-512 (A7624)		KAT	CAST	Module started	Hash	Module load

Table 21: Conditional Self-Tests

The module performs all Known Answer Tests for the above approved cryptographic algorithms immediately upon module initialization.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256	KAT	SW/FW Integrity	On-Demand	Reboot

Table 22: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-256 (A7623)	KAT	CAST	On demand	Manual
SHA2-512 (A7623)	KAT	CAST	On demand	Manual
IKEv2 KDF (A7623)	KAT	CAST	On demand	Manual
HMAC-SHA2-256 (A7623)	KAT	CAST	On demand	Manual
AES-ECB (A7623) Encrypt	KAT	CAST	On demand	Manual
AES-ECB (A7623) Decrypt	KAT	CAST	On demand	Manual
AES-GCM (A7623) Encrypt	KAT	CAST	On demand	Manual
AES-GCM (A7623) Decrypt	KAT	CAST	On demand	Manual
Counter DRBG (A7623)	KAT	CAST	On demand	Manual
ECDSA SigGen (FIPS186-5) (A7623)	KAT	CAST	On demand	Manual
ECDSA SigVer (FIPS186-5) (A7623)	KAT	CAST	On demand	Manual
RSA SigGen (FIPS186-5) (A7623)	KAT	CAST	On demand	Manual
RSA SigVer (FIPS186-5) (A7623)	KAT	CAST	On demand	Manual
KAS-ECC-SSC Sp800-56Ar3 (A7623)	KAT	CAST	On demand	Manual
KAS-FFC-SSC Sp800-56Ar3 (A7623)	KAT	CAST	On demand	Manual
ECDH PCT	PCT	PCT	On demand	Manual
DH PCT	PCT	PCT	On demand	Manual
AES-CBC (A7624) Encrypt	KAT	CAST	On demand	Manual
AES-CBC (A7624) Decrypt	KAT	CAST	On demand	Manual
AES-GCM (A7624) Encrypt	KAT	CAST	On demand	Manual
AES-GCM (A7624) Decrypt	KAT	CAST	On demand	Manual
AES-CCM (A7624) Encrypt	KAT	CAST	On demand	Manual
AES-CCM (A7624) Decrypt	KAT	CAST	On demand	Manual
HMAC-SHA2-256 (A7624)	KAT	CAST	On demand	Manual
HMAC-SHA2-384 (A7624)	KAT	CAST	On demand	Manual
HMAC-SHA2-512 (A7624)	KAT	CAST	On demand	Manual
SHA2-256 (A7624)	KAT	CAST	On demand	Manual
SHA2-384 (A7624)	KAT	CAST	On demand	Manual
SHA2-512 (A7624)	KAT	CAST	On demand	Manual

Table 23: Conditional Periodic Information

The module does not implement any periodic self-tests.

10.4 Error States

During the Hard Error state, the entire module, including the IKE Daemon and swanctl interfaces are uninitialized, meaning the module has no operational interfaces.

Failure of the pre-operational self-tests and CASTs results in the module failing to initialize. Failure of either the ECDH or DH PCTs results in the module immediately terminating.

To recover from the error state, the module must be restarted and perform self-tests again. If the module cannot recover from the error state, the operator must contact the vendor.

Name	Description	Conditions	Recovery Method	Indicator
Hard Error	Module is in an inoperable state. No services are available in this state.	Any pre-operational or conditional self-test failure.	Power cycle module.	The charon daemon is not running

Table 24: Error States

10.5 Operator Initiation of Self-Tests

All self-tests can be performed by re-initializing the module, or rebooting the host device.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The module is preinstalled on the Ad Noctem GB-100 Hardware. There are no installation steps specific to the module.

Upon receipt of the Module Platform, the end user should be able to power-on the GB-100 device (TOEPP) and after successfully boot, initialize the module and verify that it is running:

```
systemctl start strongswan
```

If the module is running, all pre-operational and conditional algorithm self-tests have been run successfully.

11.2 Administrator Guidance

The module is delivered in the approved mode by the vendor. It is pre-configured with:

- connection templates
- endpoint details for connection
- credentials for the endpoint

The module configuration is stored on the device under the directory `/etc/swanctl/swanctl.conf`. It may be edited by the crypto-officer. If the configuration is changed, the module must be rebooted by the operator before using any approved or non-approved services.

The module must be configured and run using only the approved algorithms. Using any of the VPN functionality with non-approved functions results in the module entering the non-approved mode.

All configuration data within `/etc/swanctl/swanctl.conf` must adhere to [strongSwan documentation](#).

The operator must ensure that no authentication data, SSPs or configurations are shared between the approved and non-approved mode. If the operator intends to switch between modes, a new configuration that does not reuse any authentication data or SSPs must be used.

To ensure AES-GCM IV uniqueness, the module configuration must include:

- The `esn = yes` flag to ensure that Extended Sequence Numbers (ESN) are enabled.
- Sufficiently low `rekey_time` to enforce IKE SA rekeying after a given time.
- Sufficiently low `life_time`, `life_bytes`, `life_packets` to limit the lifetime of IKE SAs.

11.3 Non-Administrator Guidance

The module is delivered pre-installed and pre-configured. The end user should verify that the module is operating in the approved mode by checking the approved service indicators.

Show Status

The status of the module can be obtained from the command `swanctl --list-sas`, which outputs all active IKE and IPsec Security Associations.

Show Version

The module version should be obtained from the command `cat /etc/adnoctem-release`, which outputs the module version, along with the board support package version.

The output of this command shows both the version of the module, and the board support package. The version of the module is the `ADNOCTEM_VERSION`.

Approved Service Indicators

The LIST-SAS-IKE and LIST-SAS-ESP indicators are obtained by executing the `swanctl --list-sas` command, which outputs the established connection algorithms. The operator can query the connection information to determine whether an Approved service has been invoked.

The LIST-SAS-IKE indicator is the section of the command output which specifies ciphersuite algorithms used for the IKEv2 connection. If all algorithms are Approved, then the service is deemed also Approved. The LIST-SAS-IKE indicator consists of three components:

Authentication Encryption, Pseudo-Random Functions, and Diffie-Hellman groups. The approved algorithms, as output by the LIST-SAS-IKE indicator, are as follows:

Authenticated Encryption:

- AES_GCM_16-128
- AES_GCM_16-192
- AES_GCM_16-256
- AES_CCM_128
- AES_CCM_192
- AES_CCM_256
- AES_CTR_128
- AES_CBC-128/HMAC_SHA2_512_256
- AES_CBC-128/HMAC_SHA2_384_192
- AES_CBC-128/HMAC_SHA2_256_128
- AES_CBC-192/HMAC_SHA2_512_256
- AES_CBC-192/HMAC_SHA2_384_192
- AES_CBC-192/HMAC_SHA2_256_128
- AES_CTR-128/HMAC_SHA2_512_256
- AES_CTR-128/HMAC_SHA2_384_192
- AES_CTR-128/HMAC_SHA2_256_128
- AES_CTR-192/HMAC_SHA2_512_256
- AES_CTR-192/HMAC_SHA2_384_192
- AES_CTR-192/HMAC_SHA2_256_128
- AES_CTR-256/HMAC_SHA2_512_256
- AES_CTR-256/HMAC_SHA2_384_192
- AES_CTR-256/HMAC_SHA2_256_128

Pseudo-Random Functions:

- PRF_HMAC_SHA2_256
- PRF_HMAC_SHA2_384
- PRF_HMAC_SHA2_512

Diffie-Hellman groups:

- ECP_224
- ECP_256
- ECP_384
- ECP_521
- MODP_2048
- MODP_3072
- MODP_4096
- MODP_6144
- MODP_8192

The LIST-SAS-ESP indicator is the section of the command output which specifies ciphersuite algorithms used for the IPsec ESP encryption. The approved algorithms, as output by the LIST_SAS-ESP indicator, are as follows:

- AES_GCM_16-128
- AES_GCM_16-192
- AES_GCM_16-256
- AES_CCM_128
- AES_CCM_192
- AES_CCM_256
- AES_CTR_128
- AES_CBC-128/HMAC_SHA2_512_256
- AES_CBC-128/HMAC_SHA2_384_192
- AES_CBC-128/HMAC_SHA2_256_128
- AES_CBC-192/HMAC_SHA2_512_256
- AES_CBC-192/HMAC_SHA2_384_192
- AES_CBC-192/HMAC_SHA2_256_128
- AES_CTR-128/HMAC_SHA2_512_256
- AES_CTR-128/HMAC_SHA2_384_192
- AES_CTR-128/HMAC_SHA2_256_128
- AES_CTR-192/HMAC_SHA2_512_256
- AES_CTR-192/HMAC_SHA2_384_192
- AES_CTR-192/HMAC_SHA2_256_128
- AES_CTR-256/HMAC_SHA2_512_256
- AES_CTR-256/HMAC_SHA2_384_192
- AES_CTR-256/HMAC_SHA2_256_128

The signature algorithms can be verified from the swanctl --LIST-CERTS indicators and checking the pubkey value:

- RSA 2048 bits, has private key
- RSA 3072 bits, has private key
- RSA 4096 bits, has private key
- ECDSA 224 bits, has private key (P-224)
- ECDSA 256 bits, has private key (P-256)
- ECDSA 384 bits, has private key (P-384)
- ECDSA 521 bits, has private key (P-521)

11.4 End of Life

The module does not store persistent SSPs, as such, to securely sanitize the module the software should be unloaded or the host OS should be restarted.

12 Mitigation of Other Attacks

The module does not implement security mechanisms to mitigate other attacks.