NetApp Cryptographic Security Module

Module Version 2.0

FIPS 140-2 Level 1 Non-Proprietary Security Policy

Document Version: 1.0

Last Updated: August 26th, 2022

Table of Contents

1 Introduction

This document is the non-proprietary Cryptographic Module Security Policy for the NetApp Cryptographic Security Module (NCSM). This security policy describes how the NCSM (Software Version: 2.0) meets the security requirements of FIPS 140-2, and how to operate it in a secure FIPS 140-2 mode. This policy was prepared as part of the Level 1 FIPS 140-2 validation of the NetApp Cryptographic Security Module.

This document provides an overview of the NetApp Cryptographic Security Module and explains the secure configuration and operation of the module. With the exception of this Non-Proprietary Security Policy, the FIPS 140-2 Validation Submission Documentation is NetApp-proprietary and is releasable only under appropriate non- disclosure agreements. For access to these documents, please contact NetApp Inc.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2 — *Security Requirements for Cryptographic Modules*) details the U.S. Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the NIST website at http://csrc.nist.gov/groups/STM/index.html.

2 Cryptographic Module Description

The NetApp Cryptographic Security Module is a software library that provides cryptographic services to a vast array of NetApp's storage and networking products.

The module provides FIPS 140-2 validated cryptographic algorithms for services such as IPSEC, SRTP, SSH, TLS, 802.1x, etc. The module does not directly implement any of these protocols, instead it provides the cryptographic primitives and functions to allow a developer to implement the various protocols.

In this document, the NetApp Cryptographic Security Module is referred to as NCSM, the library, or the module.

2.1 Module Specification

The module is a multi-chip standalone cryptographic module. For the purposes of the FIPS 140-2 level 1 validation, the NCSM is a single object module file named fipscanister.o. The object code in the object module file is incorporated into the runtime executable application at the time the binary executable is generated. The Module performs no communications other than with the consuming application (the process that invokes the Module services via the Module's API).

2.2 Module Block Diagram

The module's logical block diagram is shown in Figure 1 below. The dashed red border denotes the logical cryptographic boundary of the module. The physical cryptographic boundary of the module is the enclosure of the system on which it is executing and is denoted by the solid red boundary.

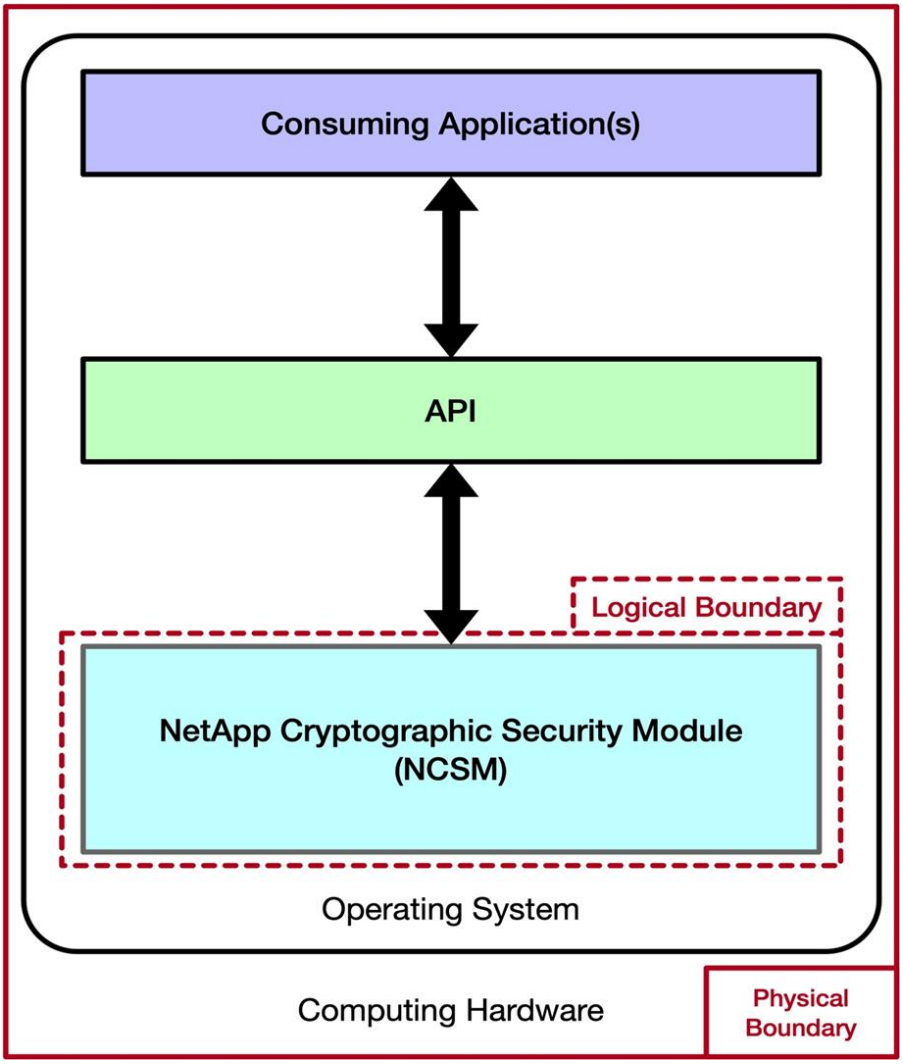

Figure 1 – NCSM block diagram

2.3 Validation Level

The following table lists the level of validation for each area in the FIPS PUB 140-2.

No.	Area Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1

No.	Area Title	Level
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key management	1
8	Electromagnetic Interface/Electromagnetic Compatibility	1
9	Self-Tests	1
10	Design Assurance	3
11	Mitigation of Other Attacks	N/A
	Overall module validation level	**1**

Table 1 – Module Validation Level

2.4 Tested Platforms

This module was tested on the following platforms for the purposes of this FIPS validation:

#	Platform	Operating System	Processor
1	ONTAP (AFF-A250)	ONTAP 9.10.1	Intel(R) Xeon(R) D-2164IT CPU @ 2.10GHz (2095.14-MHz K8-class CPU) with PAA
2	ONTAP (AFF-A250)	ONTAP 9.10.1	Intel(R) Xeon(R) D-2164IT CPU @ 2.10GHz (2095.14-MHz K8-class CPU) without PAA
3	ONTAP (FAS-2750)	ONTAP 9.10.1	Intel(R) Xeon(R) CPU D-1557 @ 1.50GHz with PAA
4	ONTAP (FAS-2750)	ONTAP 9.10.1	Intel(R) Xeon(R) CPU D-1557 @ 1.50GHz without PAA
5	SolidFire (H610S)	Element 12.5	Intel(R) Xeon(R) Gold 5120 CPU @ 2.20GHz with PAA
6	SolidFire (H610S)	Element 12.5	Intel(R) Xeon(R) Gold 5120 CPU @ 2.20GHz without PAA

Table 2 – Tested Operational Environments (OEs)

2.5 Vendor-Affirmed Platforms

This module is vendor-affirmed to operate on the following platforms in single-user mode. The CMVP makes no claim as to the correct operation of the module or the security strengths of the generated keys when ported to an operational environment which is not listed on the validation certificate.

- AFF C190 running ONTAP 9.10.1
- AFF A900 running ONTAP 9.10.1
- AFF A800 running ONTAP 9.10.1
- AFF A700 running ONTAP 9.10.1
- AFF A400 running ONTAP 9.10.1
- FAS9000 running ONTAP 9.10.1
- FAS8700 running ONTAP 9.10.1
- FAS8300 running ONTAP 9.10.1
- FAS500f running ONTAP 9.10.1
- FAS2720 running ONTAP 9.10.1
- ASA AFF A800 running ONTAP 9.10.1
- ASA AFF A700 running ONTAP 9.10.1
- ASA AFF A400 running ONTAP 9.10.1
- ASA AFF A250 running ONTAP 9.10.1

3 Cryptographic Module Ports and Interfaces

The physical ports of the Module are the same as the system on which it is executing. The logical interface is a C-language application program interface (API).

The Data Input interface consists of the input parameters of the API functions. The Data Output interface consists of the output parameters of the API functions. The Control Input interface consists of the actual API functions. The Status Output interface includes the return values of the API functions.

The module provides a number of physical and logical interfaces to the device, and the physical interfaces provided by the module are mapped to the following FIPS 140-2 defined logical interfaces: data input, data output, control input, status output, and power. The logical interfaces and their mapping are described in the following table:

Interface	Description
Data Input	API input parameters - plaintext and/or ciphertext data
Data Output	API output parameters - plaintext and/or ciphertext data
Control Input	API function calls - function calls, or input arguments that specify commands and control data used to control the operation of the module
Status Output	API return codes- function return codes, error codes, or output arguments that receive status information used to indicate the status of the module

Table 3 – FIPS 140-2 Logical Interfaces

4 Roles, Services and Authentication

4.1 Roles

The Module meets all FIPS 140-2 level 1 requirements for Roles and Services, implementing both Crypto-User and Crypto-Officer roles. The User and Crypto Officer roles are implicitly assumed by the entity accessing services implemented by the Module. The Crypto Officer can install and initialize the Module. The Crypto Officer role is implicitly entered when installing the Module or performing system administration functions on the host operating system.

- User Role: Loading the Module and calling any of the API functions. This role has access to all of the services provided by the Module.
- Crypto-Officer Role: All of the User Role functionality as well as installation of the Module on the host computer system. This role is assumed implicitly when the system administrator installs the Module library file.

4.2 Services

Service	Role	CSP	Access
Module Installation	Crypto Officer	None	N/A
Symmetric encryption/decryption	User, Crypto Officer	Symmetric keys AES, Triple- DES	Execute
Symmetric Digest	User, Crypto Officer	AES CMAC key	Execute
Key transport	User, Crypto Officer	Asymmetric private key RSA	Execute
Key agreement	User, Crypto Officer	DH[1] and ECDH private key	Execute
Key derivation (TLS KDF)	User, Crypto Officer	TLS Pre-Master Secret, TLS Master Secret	Write/Execute
Digital signature	User, Crypto Officer	Asymmetric private key RSA, DSA, ECDSA	Execute
Key Generation (Asymmetric)	User, Crypto Officer	Asymmetric keys RSA, DSA, and ECDSA	Write/Execute
Key Generation (Symmetric)	User, Crypto Officer	Symmetric keys AES, Triple- DES	Write/Execute
Keyed Hash (HMAC)	User, Crypto Officer	HMAC key	Execute
Message digest (SHS)	User, Crypto Officer	None	N/A
Random Number Generation	User, Crypto Officer	Seed/entropy input, C, and V	Write/Execute

[1] Diffie-Hellman key agreement is tested but not used on the H610s platform.

Service	Role	CSP	Access
Show status	User, Crypto Officer	None	N/A
Module initialization	User, Crypto Officer	None	N/A
Perform Self-test	User, Crypto Officer	None	N/A
Zeroization	User, Crypto Officer	All CSPs	N/A

Table 4 – Roles, Services, and Keys

4.3 Authentication

As allowed by FIPS 140-2 at Security Level 1, the Module does not support user authentication for the provided roles. Only one role may be active at a time and the Module does not allow concurrent operators.

5 Physical Security

The module is comprised of software only and thus does not claim any physical security.

6 Operational Environment

The Module operates in a modifiable operational environment.

The tested operating systems segregate user processes into separate process spaces. Each process space is an independent virtual memory area that is logically separated from all other processes by the operating system software and hardware. The Module functions entirely within the process space of the process that invokes it, and thus satisfies the FIPS 140-2 requirement for a single-user mode of operation.

7 Cryptographic Key Management

7.1 Cryptographic Algorithms

The module implements a variety of approved and non-approved algorithms.

7.1.1 Approved Cryptographic Algorithms

The module supports the following FIPS 140-2 approved algorithm implementations:

Algorithm	Supported Modes	Algorithm Certificate Numbers
AES	**[SP 800-38A]** **ECB** (encrypt/decrypt; **key sizes**: 128, 192, 256); **CBC** (encrypt/decrypt; **key sizes**: 128, 192, 256); **CFB1** (encrypt/decrypt; **key sizes**: 128, 192, 256); **CFB8** (encrypt/decrypt; **key sizes**: 128, 192, 256); **CFB128** (encrypt/decrypt; **key sizes**: 128, 192, 256); **OFB** (encrypt/decrypt; **key sizes**: 128, 192, 256); **CTR** (ext only; **key sizes**: 128, 192, 256) **[SP 800-38B]** **CMAC** (generate/verify) **(key sizes**: 128, 192, 256; **Block Size(s)**: Full / Partial; **Msg Len(s)** Min: 0 Max: 2^16; **Tag Len(s)** Min: 2 Max: 16) **[SP 800-38C]** **CCM** (encrypt/decrypt; **key sizes**: 128, 192, 256) (**Assoc. Data Len Range**: 0 - 2^19) (**Payload Length Range**: 0 - 256) (**Nonce Length(s)**: 56, 64, 72, 80, 88, 96, 104) (**Tag Length(s)**: 32, 48, 64, 80, 96, 112, 128) **[SP 800-38D]** **GMAC** (encrypt/decrypt; **key sizes**: 128, 192, 256; **Tag Len(s):** 128, 120, 112, 104, 96, 64, 32; **IV Gen**: external) **GCM** (encrypt/decrypt; **key sizes**: 128, 192, 256; **Tag Length(s)**: 128, 120, 112, 104, 96, 64, 32) **IV Gen**: external) **[SP 800-38E]**	A2157

Algorithm	Supported Modes	Algorithm Certificate Numbers
	XTS[2] (encrypt/decrypt**; key sizes**: 128, 256; **Block Size(s)**: Full / Partial; **Tweak Mode**: hex)	
Triple-DES [3]	**[SP 800-67]** **TDES ECB**(KO 1 e/d); **TDES CBC**(KO 1 e/d); **TDES CFB1**(KO 1 e/d); **TDES CFB8**(KO 1 e/d); **TDES CFB64**(KO 1 e/d); **CMAC**((**KS: 3-Key**; generate/verify; Block Size(s): Full / Partial; Msg Len(s) Min: 0 Max: 2^19; Tag Len(s) Min: 2 Max: 8))	A2157
SHS	**[FIPS 180-4]** **SHA-1** (BYTE-only) **SHA-224** (BYTE-only) **SHA-256** (BYTE-only) **SHA-384** (BYTE-only) **SHA-512** (BYTE-only)	A2157
HMAC	**[FIPS 198-1]** **HMAC-SHA1 (Key Sizes Ranges Tested:** KS<BS KS=BS KS>BS) **HMAC-SHA224 (Key Size Ranges Tested:** KS<BS KS=BS KS>BS) **HMAC-SHA256 (Key Size Ranges Tested:** KS<BS KS=BS KS>BS) **HMAC-SHA384 (Key Size Ranges Tested:** KS<BS KS=BS KS>BS) **HMAC-SHA512 (Key Size Ranges Tested:** KS<BS KS=BS KS>BS)	A2157
DRBG	**[SP 800-90A]** **Hash_Based DRBG**: [Prediction Resistance Tested: Enabled and Not Enabled]	A2157

[2] The length of a single data unit encrypted with using AES-XTS shall not exceed 2^{20} AES blocks (16MB) of data. Please note that the AES-XTS can only be used for storage applications.

[3] There is a limit of 2^16 encryptions with the same Triple-DES key. The user is responsible for ensuring the module does not surpass this limit. IG A.13 states that the same Triple-DES key shall not be used to encrypt more than 2^{28} 64-bit blocks of data. Triple-DES encrypt shall not be used after December 31, 2023. Provides 112 bits of security strength.

Algorithm	Supported Modes	Algorithm Certificate Numbers
	HMAC_Based DRBG: [Prediction Resistance Tested: Enabled and Not Enabled] **CTR_DRBG**: [Prediction Resistance Tested: Enabled and Not Enabled; Options: AES-128, 192, 256; BlockCipher_Use_df]	
RSA	**[FIPS 186-4]** **186-4KEY(gen):** FIPS186-4_Random_e **PGM(ProbPrimeCondition):** 2048, 3072, 4096 **PPTT:**(C.2) **ALG[ANSIX9.31]** Sig(Gen): (2048 SHA(256, 384, 512)) (3072 SHA(256, 384, 512)) (4096 SHA(256, 384, 512)) Sig(Ver): (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) (4096 SHA(1, 256, 384, 512)) **ALG[RSASSA-PKCS1_V1_5]** SIG(gen) (2048 SHA (224, 256, 384, 512)) (3072 SHA(224, 256, 384, 512)) (4096 SHA(224, 256, 384, 512)) SIG(Ver) (1024 SHA(1, 224, 256, 384, 512)) (2048 SHA(1, 224, 256, 384, 512)) (3072 SHA(1, 224, 256, 384, 512)) (4096 SHA(1, 224, 256, 384, 512)) **[RSASSA-PSS]:** Sig(Gen): (2048 SHA(224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (4096 SHA(224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) Sig(Ver): (1024 SHA(1 SaltLen(10), 224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(10), 224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(10), 224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (4096 SHA(1 SaltLen(10), 224 SaltLen(28), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))	A2157
DSA	**[FIPS186-4]** **PQG(gen)PARMS TESTED:** [(2048, 224)SHA(224, 256, 384, 512); (2048,256)SHA(256, 384, 512); (3072,256) SHA(256, 384, 512)] **PQG(ver)PARMS TESTED:** [(1024,160) SHA(1, 224, 256, 384, 512); (2048,224) SHA(224, 256, 384,	A2157

Algorithm	Supported Modes	Algorithm Certificate Numbers
	512); (2048,256) SHA(256, 384, 512); (3072,256) SHA(256, 384, 512)] **KeyPairGen:** [(2048,224); (2048,256); (3072,256)] **SIG(gen)PARMS TESTED:** [(2048,224) SHA(224, 256, 384, 512); (2048,256) SHA(224, 256, 384, 512); (3072,256) SHA(224, 256, 384, 512);] **SIG(ver)PARMS TESTED:** [(1024,160) SHA(1, 224, 256, 384, 512); (2048,224) SHA(1, 224, 256, 384, 512); (2048,256) SHA(1, 224, 256, 384, 512); (3072,256) SHA(1, 224, 256, 384, 512)]	
ECDSA	**[FIPS 186-4]** **PKG: CURVES**(P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571 ExtraRandomBits, TestingCandidates) **PKV: CURVES**(P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571) **SigGen: CURVES**(P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571 with hash algorithms (SHA-224, 256, 384, 512)) **SigVer: CURVES**(P-192, P-224, P-256, P-384, P-521, K-163, K-233, K-283, K-409, K-571, B-163, B-233, B-283, B-409, B-571 with hash algorithms (SHA-1, 224, 256, 384, 512))	A2157
CKG	**[SP 800-133rev2]** Symmetric key and asymmetric seed generation in accordance with SP 800-133rev2 and IG D.12	Vendor Affirmed
CVL	**[SP 800-135rev1][4]** Key derivation using TLS 1.0/1.1 and 1.2 KDF (only performed in the context of the TLS protocol)	A2157
KAS-ECC-SSC (Cert. #A2157; key establishment methodology provides between 112	**[SP 800-56rev3]** Key Agreement Scheme Shared Secret Computation (KAS-SSC) per SP 800-56Arev3	A2157

[4] No parts of the TLS protocol, other than the KDF, have been tested by the CAVP and CMVP.

Algorithm	Supported Modes	Algorithm Certificate Numbers
and 256 bits of encryption strength) KAS-FFC-SSC (Cert. #A2157; key establishment methodology provides 112 bits of encryption strength)	EC Diffie-Hellman[5]: P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571 Scheme: ephemeralUnified Diffie-Hellman[6]: FB, FC Scheme: dhEphem	

Table 5 – Approved Cryptographic Algorithms

7.1.2 Non-FIPS Approved Algorithms Allowed in FIPS Mode

The module supports the following non-FIPS approved algorithms which are permitted for use in the FIPS approved mode:

- RSA (key wrapping; key establishment methodology[7] provides between 112 and 150 bits of encryption strength)

7.1.3 Non-FIPS Approved Algorithms Not-Allowed in FIPS Mode

The module supports the following non-FIPS approved algorithms which are not permitted for use in the FIPS approved mode:

- Diffie-Hellman with 1024-bit keys
- EC Diffie-Hellman with B, K, and P curve sizes 163 and 192
- RSA Signature Generation with 1024-bit keys
- Two-Key Triple-DES encryption.

[5] The elliptic curves used in the KAS-SSC shall be the validated NIST-recommended curves and shall provide a minimum of 112 bits of encryption strength.
[6] Diffie-Hellman is tested, but not used, on the H610s platform. FB and FC domain parameters are only recommended for use in legacy applications for backwards compatibility.
[7] Using PKCS#1-v1.5 and 2048 or greater modulus. This allowance as per IG D.9 expires on December 31, 2023.

7.2 Key Generation

The Module supports generation of AES, Triple-DES keys and DH, ECDH, DSA, RSA, and ECDSA public-private key pairs. The Module employs a NIST SP800-90A random number generator for creation of both symmetric keys and the seed for asymmetric key generation. The direct output, U, from the Approved DRBG is used as keying material as discussed in IG D.12 and SP 800-133.

The module passively receives entropy from the calling application via the RAND_add() API. Because the amount of entropy loaded by the application is dependent on the "entropy" parameter used by the calling application, the minimum number of bits of entropy is considered equal to the "entropy" parameter selection of the calling application. The calling application must call the RAND_add() with the "entropy" parameter of at least 32-bytes (256-bits).

7.3 Key Storage

The Module does not perform persistent storage of any keys entered into or generated by the module.

7.4 Key Access

An authorized application as user (the Crypto-User) has access to all key data generated during the operation of the Module.

7.5 Key Protection and Zeroization

Keys residing in internally allocated data structures can only be accessed using the Module defined API. The operating system protects memory and process space from unauthorized access. Zeroization of sensitive data is performed automatically by API function calls for intermediate data items. Only the process that creates or imports keys can use or export them. No persistent storage of key data is performed by the Module. All API functions are executed by the invoking process in a nonoverlapping sequence such that no two API functions will execute concurrently. All CSPs can be zeroized by power-cycling the module (with the exception of the Software Integrity key).

7.6 AES GCM IV Generation

For AES-GCM IV generation, the method is user selectable, and the value can be computed in more than one manner.
For IG A.5 Scenario 1, following RFC 5288 for TLS 1.2, the module ensures that it is strictly increasing and thus cannot repeat. When the IV exhausts the maximum number of possible values for a given session key, the first party, client or server, to encounter this condition may either trigger a handshake to establish a new encryption key in accordance with RFC 5246, or fail. In either case, the module prevents and IV duplication and thus enforces the security property. The module is compliant to Section 3.3.1 for SP 800-52 rev2. The Module's IV is generated internally

by the Module's Approved DRBG. The IV is 96 bits in length per NIST SP 800-38D, Section 8.2.2 and FIPS 140-2 IG A.5 Scenario 2.
The selection of the IV construction method is the responsibility of the user of this Module. In the Approved mode, users of the Module must not utilize GCM with an externally generated IV.
In the event Module power is lost and restored the consuming application must ensure that any AES-GCM keys used for encryption or decryption are re-distributed.

7.7 CSP Information

The module supports the following keys and critical security parameters (CSPs):

ID	Algorithm	Size	Description
Symmetric Keys	AES	AES (ECB, CFB8, OFB, CTR, CCM, GCM): 128, 192, 256 bits AES-XTS (Key1, Key2): 128, 256 bits	Used for symmetric encryption/decryption
	Triple-DES	Triple-DES[8] (TECB, TCFB1, TCFB8, TCFB64, TOFB) Triple-DES Notes Related to use if Two-Key algorithm: *Two-key Triple-DES may only be used to decrypt data. Two-key Triple-DES may not be used to encrypt data*	
Asymmetric Keys	RSA DSA ECDSA	RSA (FIPS 186-4): 1,024-4,096 bits DSA (FIPS 186-4): 1,024, 2,048, 3,072 bits ECDSA (FIPS 186-4): P-192, P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571	Used for signature verification. RSA: Also used for key transport.
Asymmetric Keys	RSA DSA ECDSA	RSA (FIPS 186-4): 2,048-4,096 bits DSA (FIPS 186-4): 2,048, 3,072 bits ECDSA (FIPS 186-4): P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571	Used for signature generation with SHA-2 Used in key pair generation. RSA: Also used for key transport.

[8] The actual size of a Triple-DES key is 192 bits. The key has 168 "independent" (without parity) bits which results in a key strength of 112 bits.

ID	Algorithm	Size	Description
Diffie-Hellman/ EC Diffie-Hellman private key	DH ECDH	DH[9]: Public Key – 2,048-10,000 bits Private Key – 224-512 bits ECDH: P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571	Used for key agreement
TLS Master, Pre-Master Secret	CVL	TLS 1.0/1.1 and TLS 1.2 KDF	Shared secret values comprised of data used in the context of the TLS protocol
Hash_DRBG	DRBG (as per NIST SP 800-90A)	– V (440/888 bits) – C (440/888 bits) – entropy input (The length of the selected hash)	CSPs as per NIST SP800-90A.
HMAC_DRBG	DRBG (as per NIST SP 800-90A)	– V (160/224/256/384/512 bits) – Key (160/224/256/384/512 bits) – entropy input (The length of the selected hash)	CSPs as per NIST SP800-90A.
CTR_DRBG	DRBG (as per NIST SP 800-90A)	– V (128 bits) – Key (AES 128/192/256) – entropy input (The length of the selected AES)	CSPs as per NIST SP800-90A.
Keyed Hash key	HMAC	All supported key sizes for HMAC (Keys must be a minimum 112-bits)	Used for keyed hash

Table 7 – Cryptographic Keys and CSPs

[9] FB and FC domain parameters are only recommended for use in legacy applications for backwards compatibility.

8 Electromagnetic Interference/Compatibility

The Module only electromagnetic interference produced is that of the host platform on which the module resides and executes. FIPS 140-2 requires that the host systems on which FIPS 140-2 testing is performed meet the Federal Communications Commission (FCC) EMI and EMC requirements for business use as defined in Subpart B, Class A of FCC 47 Code of Federal Regulations Part 15. However, all systems sold in the United States must meet these applicable FCC requirements.

9 Self-Tests

The Module performs both power-up self-tests at module initialization[10] and continuous condition tests during operation. Input, output, and cryptographic functions cannot be performed while the Module is in a self-test or error state as the module is single threaded and will not return to the calling application until the power-up self-tests are complete. If the power-up self- tests fail subsequent calls to the module will fail and thus no further cryptographic operations are possible.

The self-tests are called when initializing the module, or alternatively can be invoked at operator discretion using the *FIPS_selftest()* function call.

9.1 Power-On Self Tests (POST)

- Software Integrity Test (HMAC-SHA1)
- AES ECB Known Answer Test (Separate encrypt and decrypt. Key size: 128-bits)
- AES-CCM Known Answer Test (Separate encrypt and decrypt. Key size: 128-bits)
- AES-GCM Known Answer Test (Separate encrypt and decrypt. Key size: 128-bits)
- AES-CMAC Known Answer Test (Key sizes: 128-bits)
- AES-XTS Known Answer Test (Separate encrypt and decrypt, Key sizes: 128, 256-bits)
- Triple-DES ECB Known Answer Test (Separate encrypt and decrypt. Key size: 168-bits)
- Triple-DES CMAC Known Answer Test (Key size: 168-bits)
- RSA Sign/Verify Known Answer Test (Key sizes: 2048, 4096-bit)
- DSA Sign/Verify Pairwise Consistency Test (Key size: 2048-bit)
- FIPS 186-4 ECDSA Sign/Verify Pairwise Consistency Test (Curves: P-224, P-256, K-233)
- HMAC Known Answer Tests
    - HMAC-SHA1 Known Answer Test
    - HMAC-SHA224 Known Answer Test
    - HMAC-SHA256 Known Answer Test
    - HMAC-SHA384 Known Answer Test
    - HMAC-SHA512 Known Answer Test
- DRBG Known Answer Tests
    - HASH_DRBG Known Answer Test (HMAC-SHA-1, SHA2-224, 256, 384, 512)
    - HMAC_DRBG Known Answer Test (SHA-1, SHA2-224, 256, 384, 512)
    - CTR_DRBG Known Answer Test (AES-CTR-128, 192, 256)
- TLS v1.2 KDF KAT
- KAS-ECC-SSC primitive KAT (curve P-224)
- KAS-FFC-SSC primitive KAT (2048-bit)

[10] The FIPS mode initialization is performed when the application invokes the FIPS_mode_set() call which returns a "1" for success and "0" for failure

9.2 Conditional tests

- Pairwise consistency tests for RSA, DSA, and ECDSA (performed on sign and verify)
- Continuous random number generation test for approved DRBG.
- AES-XTS key equality check (Separate encrypt and decrypt)

9.3 Critical Function Tests

Applicable to the DRBG, as per SP800-90A, Section 11:

- Instantiate Test
- Generate Test
- Reseed Test
- Un-instantiate Test

10 Design Assurance

10.1 Secure Distribution and Installation

The NCSM is intended only for use by NetApp personnel and as such is accessible only from the secure NetApp internal web site. Only authorized employees have access to the module.

A complete revision history of the source code from which the Module was generated is maintained in a version control database[11]. The HMAC-SHA-1 of the Module distribution file as tested by the CSTL Laboratory is verified during inclusion of the Module into NetApp products.

The module comes pre-installed on various NetApp products. No customer installation is necessary.

10.2 Secure Operation

The module is architected to be compliant with all FIPS 140-2 power-on requirements. Upon invocation of the shared library or application into which the object module has been compiled, the module begins to execute a prescribed set of startup tasks including both an integrity test and the POSTs described in section 9.1 above.

If any component of the module startup fails, an internal global error flag is set to prevent subsequent invocation of any cryptographic function calls. Any such startup failure is a hard error that can only be recovered by reinstalling the Module. Upon successful completion of all startup tasks, the Module is available to enter a FIPS mode of operation.

The module is in the Approved mode of operation when only the algorithms listed in Section 7.1.1 and 7.1.2 are being used. If any of the algorithms listed in Section 7.1.3 are used the module enters the non-Approved mode of operation.

The module contains a FIPS mode initialization function *FIPS_mode_set()*. This inhibits some of the non-Approved functionality, but not all of it. When the application invokes the *FIPS_mode_set()* call, it returns a "1" for success and "0" for failure. Interpretation of this return code is the responsibility of the host application. Prior to this invocation the Module is operating in the non-FIPS mode by default. None of algorithms identified in section 7.1.2 may be used in the FIPS-approved mode of operation. The operator must ensure that these are not chosen/selected.

No operator intervention is required during the running of the self-tests.

[11] This database is internal to NetApp since the source code is not distributed with NetApp products