Brocade Fabric OS FIPS Cryptographic Module
Firmware Version 9.1.1
FIPS 140-3 Non-Proprietary Security Policy

Document Revision 1.3

Last Update: October 21, 2024

Brocade Communications Systems LLC
1320 Ridder Park Drive
San Jose, CA 95131
USA

BROCADE
A BROADCOM INC. COMPANY

Table of contents

BROCADE
A BROADCOM INC. COMPANY

List of tables

Table of figures

1. General

This non-proprietary FIPS 140-3 security policy for the Brocade Fabric OS FIPS Cryptographic Module with firmware version 9.1.1 (hereinafter referred to the module) details the secure operation of the Brocade Communications Systems LLC Brocade Fabric OS FIPS Cryptographic Module as required in Federal Information Processing Standards Publication 140-3 (FIPS 140-3) as published by the National Institute of Standards and Technology (NIST) of the United State Department of Commerce. This document, the Cryptographic Module Security Policy, also referred to as the Security Policy, specifies the security rules under which the module must operate.

The Brocade Fabric OS FIPS Cryptographic Module underpins Brocade's Fabric Operating System equipment and is used as a shared library object by applications in Brocade's Fabric Operating System, for its various cryptographic requirements. The calling applications leverage the module's well-defined APIs to initialize the module and call cryptographic algorithms for encryption/decryption, key generation, signature generation/verification, and hashing. The Brocade Fabric OS is the firmware foundation for Brocade's purpose-built network infrastructure for mission-critical storage. The Brocade Fabric OS family of supported products includes Fiber Channel directors, switches, embedded switches and network extension switches. In addition to supporting the switching functionality of these product lines, Fabric OS supports Fabric Vision Technology features for network monitoring, management, and diagnostics, as well as advanced features that help ensure the highest level of reliability, availability, and serviceability.

ISO/IEC 24759:2017 Section 6	FIPS 140-3 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	1
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

Table 1 - Security levels

The module is designed to meet an overall security level of 1.

2. Cryptographic module specification

The module is a single binary object file (libfipscrypto.so) running on the tested platform defined in Table 2, and is classified as a multi-chip standalone firmware module.

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	Fabric OS 9.1.1	Brocade G630 Switch	NXP Semiconductors T1042 (e5500 core)	Not applicable
2	Fabric OS 9.1.1	Brocade X7-8 Director	NXP Semiconductors P4080 (e500mc core)	Not applicable
3	Fabric OS 9.1.1	Brocade G730 Switch	Intel(R) Atom(TM) CPU C3338R (2 cores)	With PAA
4	Fabric OS 9.1.1	Brocade G730 Switch	Intel(R) Atom(TM) CPU C3338R (2 cores)	Without PAA

Table 2 - Tested operational environments

The following platforms have not been tested as part of the FIPS 140-3 Level 1 certification however Brocade affirms that these platforms are compliance to the tested and validated platforms. Additionally, Brocade also affirms that the Module will function the same way and provide the same security services on any of the operating systems listed below.

#	Operating System	Hardware Platform
1	Fabric OS 9.1.1	Brocade X7-4 Director
2	Fabric OS 9.1.1	Brocade X6-8 Switch
3	Fabric OS 9.1.1	Brocade X6-4 Switch
4	Fabric OS 9.1.1	Brocade G610 Switch
5	Fabric OS 9.1.1	Brocade G620 Switch
6	Fabric OS 9.1.1	Brocade G720 Switch
7	Fabric OS 9.1.1	Brocade 7810 Extension Switch

Table 3 - Vendor affirmed operational environments

Please note that the CMVP makes no statement 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.

Modes of operation

The module supports both approved and non-approved mode of operation. The module will be in approved mode when all pre-operational self-tests have completed successfully and only approved algorithms/services are invoked. See Table 4 and Table 8 below for a list of the supported approved/allowed algorithms/services. The non-approved mode is entered when a non-approved algorithm/non-approved service is invoked. See Table 5 and Table 9 below for a list of non-approved algorithms/non-approved services. The Approved mode of operation can only be transitioned into the non-Approved mode by calling one of the non-Approved services listed in Table 9.

Table 4 below lists all Approved or Vendor-affirmed security functions of the module, including specific key size(s) -in bits otherwise noted- employed for approved services, and implemented modes of operation.

CAVP / ACVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2604	AES • FIPS 197 • SP800-38D	ECB	128, 192, 256 bits	Encryption/decryption
A2604	AES • FIPS 197 • SP800-38D	CBC	128, 192, 256 bits	Encryption/decryption
A2604	AES • FIPS 197 • SP800-38D	CTR	128, 192, 256 bits	Encryption/decryption
A2604	AES-CCM • SP800-38B	AES-CCM	128, 192, 256 bits	Authenticated encryption/decryption
A2604	AES-CMAC • SP800-38B	AES-CMAC	128 or 256 bits	MAC generation/verification
A2604	Counter DRBG • SP800-90Arev1	Counter DRBG (AES-256) with Derivation Function (df)	N/A	Deterministic random bit generation
A2604	ECDSA • FIPS186-4	ECDSA KeyGen	Curves: P-256, P-384, P-521	ECDSA key generation
A2604	ECDSA • FIPS186-4	ECDSA KeyVer	Curves: P-256, P-384, P-521	ECDSA key verification
A2604	ECDSA • FIPS186-4	ECDSA SigGen	Curves: P-256, P-384, P-521	ECDSA signature generation
A2604	ECDSA • FIPS186-4	ECDSA SigVer	Curves: P-256, P-384, P-521	ECDSA signature verification
N/A	ENT (NP) • SP800-90B	N/A	N/A	Non-physical entropy source used for seeding DRBG
A2604	HMAC • FIPS198-1	HMAC-SHA1	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA2-256	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA2-384	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA2-512	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA3-224	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA3-256	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA3-384	112 bits (minimum)	Message authentication code
A2604	HMAC • FIPS198-1	HMAC-SHA3-512	112 bits (minimum)	Message authentication code

CAVP / ACVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2604	KAS-ECC-SSC • SP800-56Arev3	KAS-ECC-SSC ephemeralUnified: KAS Role: initiator, responder	Curve: P-256, P-384 and P-521 Key establishment methodology provides between 128 and 256 bits of encryption strength	SP800-56Arev3 compliant KAS-ECC shared secret computation
A2604	KAS-FFC-SSC • SP800-56Arev3	KAS-FFC-SSC Scheme: dhEphem: KAS Role: initiator, responder	MODP-2048, MODP-3072 and MODP-4096 Key establishment methodology provides between 112 and 152 bits of encryption strength	SP800-56Arev3 compliant KAS-FFC shared secret computation
A2604	RSA • FIPS186-4	RSA KeyGen	Modulus: 2048, 3072, 4096 bits	RSA key generation
A2604	RSA • FIPS186-4	RSA SigGen	Modulus: 2048, 3072, 4096 bits	RSA digital signature generation
A2604	RSA • FIPS186-4	RSA SigVer	Modulus: 2048, 3072, 4096 bits	RSA digital signature verification
A2604	Safe Primes Key Generation • SP800-56Arev3	Safe Primes KeyGen	MODP-2048, MODP-3072, MODP-4096, and MODP-8192	Keys generation with SafePrimes groups
A2604	Safe Primes Key Verification • SP800-56Arev3	Safe Primes KeyVer	MODP-2048, MODP-3072, MODP-4096, and MODP-8192	Keys verification with SafePrimes groups
A2604	SHS • FIPS180-4	SHA-1	Message Length: 0-65024 Increment 32	Hashing Note: SHA-1 is not used for digital signature generation
A2604	SHS • FIPS180-4	SHA2-224	Message Length: 0-65024 Increment 32	Hashing
A2604	SHS • FIPS180-4	SHA2-256	Message Length: 0-65024 Increment 32	Hashing
A2604	SHS • FIPS180-4	SHA2-384	Message Length: 0-65024 Increment 32	Hashing
A2604	SHS • FIPS180-4	SHA2-512	Message Length: 0-65024 Increment 32	Hashing
A2604	SHA-3 • FIPS202	SHA3-224	Message Length: 0-65536 Increment 8	Hashing
A2604	SHA-3 • FIPS202	SHA3-256	Message Length: 0-65536 Increment 8	Hashing
A2604	SHA-3 • FIPS202	SHA3-384	Message Length: 0-65536 Increment 8	Hashing

CAVP / ACVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Key Strength(s)	Use / Function
A2604	SHA-3 • FIPS202	SHA3-512	Message Length: 0-65536 Increment 8	Hashing
N/A	CKG (vendor affirmed) [SP 800-133rev2]	N/A	N/A	Cryptographic Key Generation (CKG) compliant with SP800-133rev2 and IG D.H The cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per sections 4 and 5 in SP800-133rev2 (vendor affirmed). A seed (i.e., the random value) used in asymmetric key generation is a direct output from SP800-90Arev1 CTR_DRBG

Table 4 - Approved algorithms

Notes:

- There are some algorithm modes that were tested but not implemented by the module. Only the algorithms, modes, and key sizes that are implemented by the module are shown in this table.

Algorithm/Function	Use / Function
RSA key wrapping	RSA key wrapping
HMAC-MD5	Authentication
MD5	Hashing
AES-GCM	Authenticated encryption/decryption
DSA	DSA signature sign/verify
Triple-DES	Encryption/decryption
Camellia	Encryption/decryption
SEED	Encryption/decryption
RC4	Encryption/decryption
ARIAGCM	Encryption/decryption
CHACHA20/POLY1305	Encryption/decryption

Table 5 - Non-approved algorithms not allowed in approved mode

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

In addition, the module's entropy source implements SHA3-256 algorithm with SHA-3 Cert. #A2610 as a vetted conditioning component in ENT (NP).

Cryptographic boundary

Figure 1 below depicts the cryptographic boundary (red dashed line) and physical perimeter (solid red line). The cryptographic boundary is defined as the cryptographic library. The physical perimeter is the Tested Operational Environment's Physical Perimeter (TOEPP) on which the module runs.

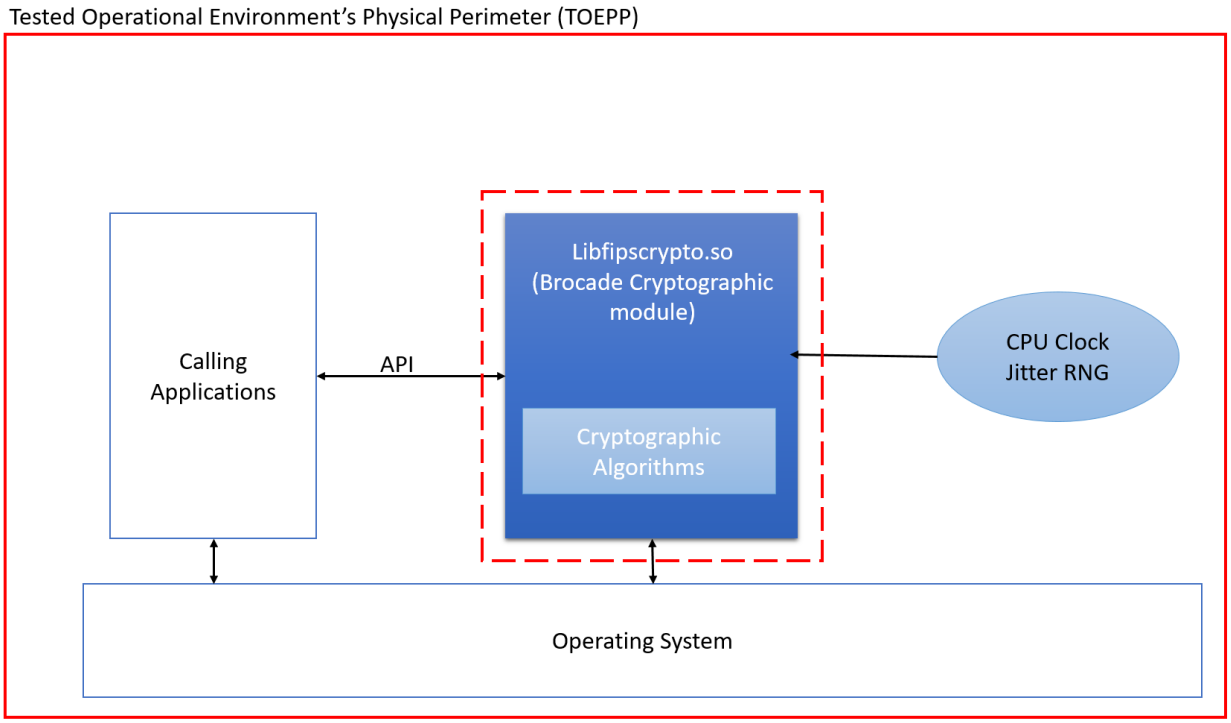

Figure 1 - Cryptographic boundary

3. Cryptographic module interfaces

The module's physical perimeter encompasses the peripheral's devices (USB devices, network devices [Ethernet and Wireless adapters], and power adapter) on the tested platform running Brocade's Fabric Operating System.

However, the module provides only a logical interface via Application Programming Interface (API) calls and does not interface or communicate with or across any of the physical ports of the GPC. This logical interface exposes service that calling applications may use directly.

The logical interfaces (APIs) provided by the module are mapped onto the FIPS 140-3 defined logical interfaces (data input, data output, control input, control output and status output). It is through this logical API that the module logically separates them into distinct and separate interfaces. Please note that the module does not implement Control Output Interface. The mapping of the module's API to the FIPS 140-3 interfaces is as follows.

Physical Port	Logical Interface Type	Data that passes over port/interface
N/A	Data Input Interface	Arguments for an API call that provide the data to be used or processed by the module (input arguments to all functions specifying input parameters).
N/A	Data Output Interface	Arguments output from an API call (includes modified input arguments (those passed by reference) and return values for all functions modifying input arguments and returning values).
N/A	Control Input Interface	Arguments for an API call used to control and configure module operation.
N/A	Control Output Interface	N/A
N/A	Status Output Interface	Return values from firmware API commands used to obtain information on the status of the module. The Status Output Interface also includes the log file where the module messages are output.

Table 6 - Cryptographic module interfaces

4. Roles, services, and authentication

The module supports both Crypto Officer (CO) role and User role. No authentication is required at security level 1 and the assumption of the role is implicit by the service being performed. The module provides the following services to the User and Crypto Officer.

Role	Service	Input	Output
Crypto Officer (CO)	Module initialization	Command to initialize the module	Module initialization status
User	Show status	API command	Module's current status
User	Show version	API command	Displays the module's name/ID and versioning information
User	Zeroization	Module reboot or power down the tested platform	SSPs Zeroization status
User	Firmware integrity test	Command to enable the firmware integrity test	Firmware Integrity Test completion status
User	Self-tests	Command to enable self-test (bootup / on-demand)	Self-Tests or conditional tests completion status
User	Encryption and decryption	Command to conduct the encryption and decryption operation	Encrypted or decrypted completion status
User	Keyed hash	Command to conduct the HMAC/CMAC operation	Keyed Hash completion status
User	Message digest	Command to conduct the Message Digest operation	Hashed completion status
User	Random number generation	Command to conduct the Counter DRBG generation	Random value generation status
User	Key agreement shared secret computation	Command to run key agreement shared secret computation	Key agreement shared secret computation status
User	Signature generation and verification	Command to conduct the signature generation and verification	Signature generation or verification completion status
User	Asymmetric key generation and verification	Command to generate/verify asymmetric cryptographic keypair	Keypair generation/verification status

Table 7 - Roles, service commands, input and output

Table 8 below defines the relationship between access to CSPs and the different module services. The modes of access shown in the table are defined as:

G = Generate: The module generates or derives the SSP.

R = Read: The SSP is read from the module (e.g., the SSP is output).

W = Write: The SSP is updated, imported, or written to the module.

E = Execute: The module uses the SSP in performing a cryptographic operation.

Z = Zeroize: The module zeroizes the SSP.

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access Rights to Keys and/or SSPs	Indicator
Module initialization	Initialize the module	N/A	N/A	Crypto Officer	E	N/A
Show status	Display running status of the module	N/A	None	User	N/A	N/A
Show version	Provide module's name and version information	N/A	None	CO	N/A	N/A
Zeroization	Zeroize all SSPs	N/A	All SSPs	User	Z	N/A
Firmware integrity test	Check signature during the firmware integrity test	HMAC-SHA-1	Firmware integrity test key (non-SSP)	User	E	Success or error code
Self-tests	Run pre-operational and conditional Algorithm Self-Tests	N/A	N/A	User	E	Pass/fail status output
Encryption and decryption	Conduct symmetric encryption and decryption	AES-ECB; AES-CBC; AES-CTR; AES-CCM	AES Key	User	E	Service Indicator log (per every use of approved algorithm)
Keyed hash	Authentication/integrity checks	AES-CMAC; HMAC	HMAC Key; CMAC Key	User	E	Service Indicator log (per every use of approved algorithm)
Message digest	Conduct message digest operation	SHA-1; SHA2-256; SHA2-384; SHA2-512; SHA3-224; SHA3-256; SHA3-384; SHA3-512	N/A	User	E	Service Indicator log (per every use of approved algorithm)
Random number generation	Generate random numbers for use by crypto module	Counter DRBG; ENT (NP)	DRBG entropy inputs; DRBG seed; DRBG internal state v; DRBG key	User	E	Service Indicator log (per every use of approved algorithm)

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access Rights to Keys and/or SSPs	Indicator
Key agreement shared secret computation	Conduct key agreement shared secret computation	KAS-ECC-SSC; KAS-FFC-SSC; Safe Prime Key Generation (for KAS-FFC-SSC only)	DH private key; DH public key; DH shared secret; ECDH private key; ECDH public key; ECDH shared secret	User	E	Service Indicator log (per every use of approved algorithm)
Signature generation and verification	Generate and verify signatures	RSA SigGen; RSA SigVer; ECDSA SigGen; ECDSA SigVer	RSA private key; RSA public key; ECDSA private key; ECDSA public key	User	E	Service Indicator log (per every use of approved algorithm)
Asymmetric key generation and verification	Generate/verify asymmetric key pair	CKG; Counter DRBG; KAS-ECC-SSC; KAS-FFC-SSC; Safe Primes KeyGen; Safe Primes KeyVer; RSA KeyGen; ECDSA KeyGen; ECDSA KeyVer	DH private key; DH public key; ECDH private key; ECDH public key; RSA private Key; RSA public key; ECDSA private key; ECDSA public key	User	G; E	Service Indicator log (per every use of approved algorithm)

Table 8 - Approved services

Service	Description	Algorithms Accessed	Roles	Indicator
RSA key wrapping	RSA key wrapping	RSA	User	
Authentication	Authentication by using HMAC-MD5	HMAC-MD5	User	N/A
Hashing	Hashing by using MD5	MD5	User	N/A
Authenticated encryption/decryption	Authenticated encryption/decryption by using AES-GCM	AES-GCM	User	N/A
DSA signature sign/verify	DSA Signature sign/verify	DSA	User	N/A
Encryption/decryption	Encryption/decryption by using Triple-DES	Triple-DES	User	N/A

Service	Description	Algorithms Accessed	Roles	Indicator
Encryption/decryption	Encryption/decryption by using Camellia	Camellia	User	N/A
Encryption/decryption	Encryption/decryption by using SEED	SEED	User	N/A
Encryption/decryption	Encryption/decryption by using RC4	RC4	User	N/A
Encryption/decryption	Encryption/decryption by using ARIAGCM	ARIAGCM	User	N/A
Encryption/decryption	Encryption/decryption by using CHACHA20/POLY1305	CHACHA20/POLY1305	User	N/A

Table 9 - Non-approved services

5. Software/Firmware security

Integrity techniques

The cryptographic module is a binary file (libfipscrypto.so) dynamically linked within the application in FOS (Fabric Operating System). To ensure firmware security, the module is protected by an HMAC-SHA-1 (HMAC Cert. #A2604) algorithm. The firmware integrity test key was preloaded to the module's binary at the factory and used only for the pre-operational firmware integrity self-test. During initialization of the module, the integrity of the runtime executable is verified using an HMAC-SHA-1 which is compared to a value computed at build time. If at load time the MAC does not match the stored, known MAC value, the module enters a critical error state where all crypto functionality inhibited. The module must be reloaded to attempt the integrity test again.

Integrity test on-demand

The integrity test is performed as part of the pre-operational self-tests. It is automatically executed at power-on. The operator can power-cycle or reboot the tested platform to initiate the integrity test on-demand.

6. Operational environment

The module will operate in a non-modifiable operational environment per the definition in FIPS 140-3 level 1 specifications. The module runs on the operating system executing on the tested platforms listed in Table 2.

The operational environment shall be restricted to a single operator mode of operation (i.e., concurrent operators are explicitly excluded). The application that requests cryptographic services is the single user of the module.

All cryptographic keys and CSPs are under the control of the OS, which protects its SSPs against unauthorized disclosure, modification, and substitution. Additionally, the OS provides dedicated process space to each executing process, and the module operates entirely within the process space.

7. Physical security

The module is running on the multi-chip standalone production grade platform to meet physical security requirements from FIPS 140-3 level 1. The module's Tested Operational Environment's Physical Perimeter (TEOPP) is drawn at the casing of the tested platforms in Table 2. The module's tested platforms consist of production-grade components. All ICs are coated with industry standard passivation.

8. Non-invasive security

The module does not claim to implement non-invasive security beyond the FIPS 140-3 Level 1 requirements for validation.

9. Sensitive security parameter management

The module possesses firmware integrity test HMAC key (non-SSP). Beyond that key, the module does not store any other keys persistently, and it is the calling applications responsibility to appropriately manage keys.

Key/SSP/ Name Type	Strength	Security Function and Cert.	Generation	Import / Export	Establish ment	Storage	Zeroization	Use & related keys
AES key	128-256 bits	AES-CBC; AES-ECB; AES-CTR; AES-CCM Cert. #A2604	N/A	Import: Module's API Export: No	MD/EE	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Data protection
DRBG entropy inputs	384 bits	ENT (NP)	Obtained from the entropy source ENT (NP)	Import: From the entropy source ENT (NP) via the API Export: No	MD/EE	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DRBG generation
DRBG seed	256 bits	Counter DRBG Cert. #A2604	Derived from entropy input string as defined by SP800-90Arev1	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DRBG generation
DRBG internal state v	256 bits	Counter DRBG Cert. #A2604	Derived from entropy input string as defined by SP800-90Arev1	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DRBG generation
DRBG key	256 bits	Counter DRBG Cert. #A2604	Derived from entropy input string as defined by SP800-90Arev1	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DRBG generation

Key/SSP/ Name Type	Strength	Security Function and Cert.	Generation	Import / Export	Establish ment	Storage	Zeroization	Use & related keys
DH private key	112-152 bits (MODP-2048, MODP-3072, MODP-4096)	CKG; Counter DRBG; Safe Primes KeyGen; Safe Primes KeyVer; KAS-FFC-SSC Cert. #A2604	Internally generated conformant to SP800-133r2 (CKG) using SP800-56Arev3 Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90Arev1 DRBG	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DH key agreement
DH public key	112-152 bits (MODP-2048, MODP-3072, MODP-4096)	Safe Primes KeyGen; Safe Primes KeyVer; KAS-FFC-SSC Cert. #A2604	Internally derived per Diffie-Hellman key agreement (SP800-56Arev3)	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DH key agreement
DH shared secret	112-152 bits (MODP-2048, MODP-3072, MODP-4096)	KAS-FFC-SSC Cert. #A2604	Internally derived per SP800-56Arev3 Diffie-Hellman shared secret computation method	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for DH key agreement
ECDH private key	128-256 bits (Curves: P-256, P-384, P-521)	CKG; Counter DRBG; KAS-ECC-SSC Cert. #A2604	Internally generated conformant to SP800-133r2 (CKG) using SP800-56Arev3 EC Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90Arev1 DRBG	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for ECDH key agreement

Key/SSP/ Name Type	Strength	Security Function and Cert.	Generation	Import / Export	Establish ment	Storage	Zeroization	Use & related keys
ECDH public key	128-256 bits (Curves: P-256, P-384, P-521)	KAS-ECC-SSC Cert. #A2604	Internally derived per EC Diffie-Hellman key agreement (SP800-56Arev3)	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for ECDH key agreement
ECDH shared secret	128-256 bits (Curves: P-256, P-384, P-521)	KAS-ECC-SSC Cert. #A2604	Internally derived per SP800-56Arev3 EC Diffie-Hellman shared secret computation method	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for ECDH key agreement
RSA private key	112-152 bits (Modulus: 2048, 3072, 4096 bits)	CKG; Counter DRBG; RSA KeyGen; RSA SigGen Cert. #A2604	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 RSA key generation method, and the random value used in key generation is generated using SP800-90Arev1 DRBG	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Digital signature generation
RSA public key	112-152 bits (Modulus: 2048, 3072, 4096 bits)	RSA SigVer Cert. #A2604	Internally derived per FIPS186-4 RSA Keypair generation method	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Digital signature verification

Key/SSP/ Name Type	Strength	Security Function and Cert.	Generation	Import / Export	Establish ment	Storage	Zeroization	Use & related keys
ECDSA private key	128-256 bits (Curves: P-256, P-384, P-521)	CKG; Counter DRBG; ECDSA KeyGen; ECDSA SigGen Cert. #A2604	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 ECDSA key generation method, and the random value used in key generation is generated using SP800-90Arev1 DRBG	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Digital signature generation
ECDSA public key	128-256 bits (Curves: P-256, P-384, P-521)	ECDSA SigVer Cert. #A2604	Internally derived per FIPS186-4 ECDSA Keypair generation method	Import: No Export: No	N/A	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Digital signature verification
HMAC key	112 bits (minimum)	HMAC-SHA-1; HMAC-SHA2-256; HMAC-SHA2-384; HMAC-SHA2-512; HMAC-SHA3-224; HMAC-SHA3-256; HMAC-SHA3-384; HMAC-SHA3-512 Cert. #A2604	N/A	Import: Module's API Export: No	MD/EE	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for keyed hash
CMAC key	128 or 256 bits	AES-CMAC Cert. #A2604	N/A	Import: Module's API Export: No	MD/EE	N/A: The module does not provide persistent keys/SSPs storage	Module reboot or power down the tested platform	Used for keyed hash

Table 10 - Sensitive security parameters

Note: All SSPs will be zeroized by all “0”s and cannot be retrievable or reusable after zeroization operation.

RBG entropy source

Entropy sources	Minimum number of bits of entropy	Details
ENT (NP): CPU Jitter (libjitterentropy v3.0.1)	0.9075 bits per bit	CPU Jitter Random Number Generator (Jitter Entropy Library v3.0.1) from Stephen Muller provides at least 256 bits entropy. Please see https://www.chronox.de/jent/doc/CPU-Jitter-NPTRNG.pdf for more information. The SHA3-256 algorithm as a vetted conditioner used in Jitter Entropy Library has been ACVP tested with the SHS Cert. #A2610

Table 11 - Non-deterministic random number generation specification

10. Self-tests

The module automatically performs both Pre-Operational Self-Tests and Cryptographic Algorithm Self-Tests (CASTs) after the power is on. Prior to providing any data output via the data output interface, the module would perform and pass the pre-operational self-tests. Following the successful pre-operational self-tests, the module would execute the Conditional Cryptographic Algorithm Self-tests (CASTs). The remaining self-tests for all approved algorithms are performed right before first instance of use of the respective algorithm. In the event a self-test fails, the module enters the critical error state and an error message is logged. In this state, cryptographic operations are halted and the module inhibits all data output from the module as the API interface is disabled. In order to attempt to exit the error state, the module must be rebooted. If the error persists, the module must be reinitialized.

Pre-operational self-tests:

- HMAC-SHA-1 KAT
- Firmware integrity test by using HMAC-SHA-1

Conditional self-tests

Cryptographic algorithm self-tests (CASTs):

- AES-ECB Encryption KAT (128 bits)
- AES-ECB Decryption KAT (128 bits)
- AES-CBC Encryption KAT (128 bits)
- AES-CBC Decryption KAT (128 bits)
- AES-CCM Encryption KAT (128 bits)
- AES-CCM Decryption KAT (128 bits)
- DRBG Instantiate KAT
- DRBG Generate KAT
- DRBG Reseed KAT

Note: DRBG Health Tests as specified in SP800-90A Section 11.3 are performed

- ECDSA SigGen KAT (Curve P-256 with SHA2-256)
- ECDSA SigVer KAT (Curve P-256 with SHA2-256)
- HMAC-SHA-1 KAT
- HMAC-SHA2-256 KAT
- HMAC-SHA2-512 KAT
- HMAC-SHA3-256 KAT
- HMAC-SHA3-512 KAT
- KAS-ECC-SSC primitive Z value KAT
- KAS-FFC-SSC primitive Z value KAT
- RSA SigGen KAT (Modulus: 2048 with SHA2-256)
- RSA SigVer KAT (Modulus: 2048 with SHA2-256)
- SHA-1 KAT
- SHA2-256 KAT
- SHA2-512 KAT

- SHA3-256 KAT
- SHA3-512 KAT

In addition, below is a list of module's entropy source health tests.

- ENT (NP) SP800-90B Start-Up Health Tests.
  - Repetition Count Test (RCT)
  - Adaptive Proportion Test (APT)

Note: Please refer to SP800-90B, sections 4.4.1 and 4.4.2 for more information about the RCT and APT.

- ENT (NP) SP800-90B Continuous Health Tests:
  - Repetition Count Test (RCT)
  - Adaptive Proportion Test (APT)

Conditional pair-wise consistency tests (PCT)

- RSA PCT
- ECDSA PCT
- KAS-ECC-SSC PCT
- KAS-FFC-SSC PCT

An operator has no access to cryptographic functionality unless the library initialization succeeds and the cryptographic module self-tests pass. The module performs the firmware integrity check of the module's firmware using verification of an HMAC-SHA-1 signature calculated over the module's file image. Please note that the module performs the CASTs prior to Firmware Integrity Test. Should the module fail a self-test, the module will return an error and inhibit all cryptographic operations. Finally, an operator may invoke all CASTs at any time by power-cycling the GPC and then reloading the module or invoke self-test(s) on demand.

Periodic self-test

The module performs on-demand self-tests initiated by the operator, by power cycling or rebooting the tested platform. The full suite of self-tests is then executed. The same procedure may be employed by the operator to perform periodic self-tests.

It is recommended that the User perform periodic testing of the module's on-demand self-tests every 60 days to ensure all components are functioning correctly.

Error handling

If any of the above-mentioned self-tests fail, the module reports the cause of the error and enters an error state (there is only one error state). In the Error State, no cryptographic services are provided, and data output is prohibited. The only method to recover from the error state is to reboot the module and perform the self-tests, including the pre-operational firmware integrity test and the conditional CASTs. The module will only enter into the operational state after successfully passing the pre-operational firmware integrity test and the conditional CASTs. Table 12 below shows the different causes that lead to the Error State and the status indicators reported.

Cause of Error	Error State Indicator
Failed Pre-Operational Firmware Integrity Test	ERROR: Libfipscrypto Critical Failure! Integrity check failed... System going for reboot!
Failed Conditional CAST	Selftest of <algorithm name> failed. System going for reboot!
Failed Conditional PCT	<Algorithm> pairwise consistency test failed
SP 800-90B Entropy Source (Start-up/Continuous health tests)	Entropy request is not serviced and error is returned to crypto module.

Table 12 - Error state

11. Life-cycle assurance

The module design corresponds to the module security rules. This section documents the security rules enforced by the cryptographic module to implement the security requirements of this FIPS 140-3 Level 1 module.

General guidance

The module meets all the Level 1 requirements for FIPS 140-3. The module functions entirely within the process space of the process that invokes it, and thus satisfies the FIPS 140-3 requirement for a single user mode of operation.

During system start-up the OS will call the fipscrypto_init_func() function. The fipscrypto_init_func() function is the default entry point for the module. The function initiates all self-tests and does not return to the OS until all self-tests are completed successfully and the module is in an approved mode of operation. No other tasks are executed while the self-tests are performed so no data is passed and all cryptographic operations are prohibited. If a self-test fails, the module enters a critical error state and must be reloaded to clear the error state and retry the self-tests.

End of life

In addition, the module is not distributed as a standalone library and is only used in conjunction with the solution. The end user of the operating system is also responsible for SSPs zeroization based on the methods listed in Table 12.

12. Mitigation of other attacks

The module does not claim to mitigate any attacks beyond the FIPS 140-3 Level 1 requirements for validation.