SUSE LLC

SUSE Linux Enterprise OpenSSL 1 Cryptographic Module

FIPS 140-3 Non-Proprietary Security Policy

Prepared by:

atsec information security corporation
4516 Seton Center Pkwy, Suite 250
Austin, TX 78759
www.atsec.com

Document version: 1.3
Last update: 2026-03-19

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for version 4.3 of the SUSE Linux Enterprise OpenSSL 1 Cryptographic Module. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for an overall Security Level 1 module.

This Non-Proprietary Security Policy may be reproduced and distributed, but only whole and intact and including this notice. Other documentation is proprietary to their authors.

1.2 Security Levels

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	1
	Overall Level	1

Table 1: Security Levels

1.3 Additional Information

In preparing the Security Policy document, the laboratory formatted the vendor-supplied documentation for consolidation without altering the technical statements therein contained. The further refining of the Security Policy document was conducted iteratively throughout the conformance testing, wherein the Security Policy was submitted to the vendor, who would then edit, modify, and add technical contents. The vendor would also supply additional documentation, which the laboratory formatted into the existing Security Policy, and resubmitted to the vendor for their final editing.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The SUSE Linux Enterprise OpenSSL 1 Cryptographic Module (hereafter referred to as "the module") is defined as a software module in a multi-chip standalone embodiment. It provides a C language application program interface (API) for use by other applications that require cryptographic functionality. The module is a software library supporting FIPS 140-3 approved algorithms developed by SUSE LLC for its use by other applications that require cryptographic functionality.

Module Type: Software

Module Embodiment: MultiChipStand

Cryptographic Boundary:

The cryptographic boundary of the module is defined as the libcrypto.so and libssl.so shared libraries and their respective integrity check files. libcrypto.so is the shared library that implements the cryptographic algorithms, whereas libssl.so is the shared library that implements the TLS/DTLS network protocols.

Tested Operational Environment's Physical Perimeter (TOEPP):

The TOEPP of the module is defined as the general-purpose computer on which the module is installed.

Figure 1 shows a block diagram that represents the design of the module when the module is operational and providing services to other user space applications. In this diagram, the physical perimeter of the operational environment is the general-purpose computer on which the module is installed. The cryptographic boundary is represented by the libssl and libcrypto shared libraries and their respective integrity check files.

The "Data/Control Input" and "Data/Status Output" arrows indicate the flow of data between the cryptographic module and its operator application, through the logical interfaces defined in Section 3 Cryptographic Module Interfaces.

Other components are only included in the diagram for informational purposes. They are not included in the cryptographic boundary (and therefore not part of the module's validation). For example, the kernel is responsible for managing system calls issued by the module itself, as well as other applications using the module for cryptographic services.

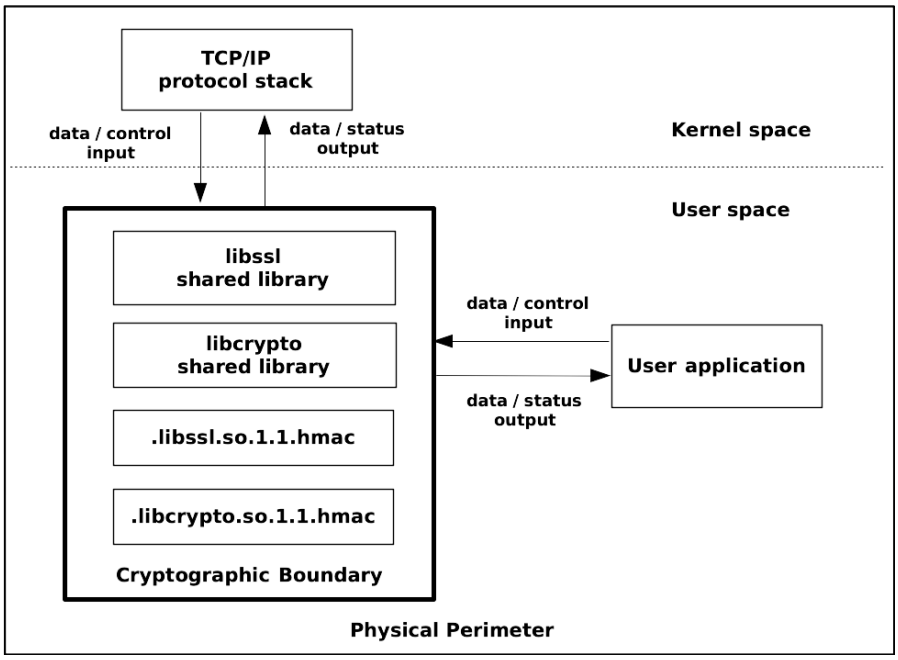

Figure 1: Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
libcrypto.so, libssl.so, .libcrypto.so.1.1.hmac, .libssl.so.1.1.hmac on SUSE Linux Enterprise Server 15 SP6 and AMD EPYC(TM) 7343 or Intel® Xeon® Gold 5416S	4.3	N/A	HMAC-SHA2-256
libcrypto.so, libssl.so, .libcrypto.so.1.1.hmac, .libssl.so.1.1.hmac on SUSE Linux Enterprise Server 15 SP6 and IBM® Telum(TM)	4.3	N/A	HMAC-SHA2-256

Package or File Name	Software/ Firmware Version	Features	Integrity Test
libcrypto.so, libssl.so, .libcrypto.so.1.1.hmac, .libssl.so.1.1.hmac on SUSE Linux Enterprise Server 15 SP6 and Ampere® Altra® Q80-30	4.3	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 Linux Enterprise Server 15 SP6	ASUS RS700-E11-RS4U	Intel® Xeon® Gold 5416S	Yes	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	ASUS RS700-E11-RS4U	Intel® Xeon® Gold 5416S	No	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	IBM z16 A01	IBM® Telum(TM)	Yes	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	IBM z16 A01	IBM® Telum(TM)	No	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	GIGABYTE R152-P30	Ampere® Altra® Q80-30	Yes	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	GIGABYTE R152-P30	Ampere® Altra® Q80-30	No	N/A	4.3

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
SUSE Linux Enterprise Server 15 SP6	SuperMicro SuperChassis 825BTQC-R1K23LPB and Motherboard H12DSi-NT6	AMD EPYC(TM) 7343	Yes	N/A	4.3
SUSE Linux Enterprise Server 15 SP6	SuperMicro SuperChassis 825BTQC-R1K23LPB and Motherboard H12DSi-NT6	AMD EPYC(TM) 7343	No	N/A	4.3

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

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform
SUSE Linux Enterprise Server for SAP 15SP6	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Server for SAP 15SP6	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Desktop 15SP6	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Desktop 15SP6	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Base Container Image 15SP6	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Base Container Image 15SP6	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Base Container Image 15SP6	GIGABYTE R152-P30 on Ampere® Altra® Q80-30
SUSE Linux Enterprise Base Container Image 15SP6	IBM z16 A01 on IBM® Telum(TM)
SUSE Linux Enterprise Server 15SP6	IBM LinuxONE III Model LT1 on z15

Operating System	Hardware Platform
SUSE Linux Enterprise Server Real Time 15SP6	QEMU VM on AMD EPYC(TM) 7773X
SUSE Linux Enterprise Desktop 15SP6	QEMU VM on AMD EPYC(TM) 7773X
SUSE Linux Enterprise Desktop 15SP6	QEMU VM on Intel® i7-1195G7
SUSE Linux Enterprise Base Container Image 15SP6	Dell XPS 13 on Intel® i7-1195G7
SUSE Linux Enterprise Base Container Image 15SP6	QEMU VM on Ampere® Altra® Q80-30
SUSE Linux Enterprise Base Container Image 15SP6	IBM LinuxONE III Model LT1 QEMU VM on z15
SUSE Linux Enterprise Server 15SP6	IBM LinuxONE III Model LT1 QEMU VM on z15
SUSE Linux Enterprise Server 15SP6	QEMU VM on AMD EPYC(TM) 7773X
SUSE Linux Enterprise Server 15SP6	QEMU VM on Ampere® Altra® Q80-30
SUSE Linux Enterprise Server 15SP6	QEMU VM on Intel® Xeon® Gold 6338
SUSE Linux Enterprise Server for SAP 15SP6	QEMU VM on Intel® Xeon® Gold 5218R
SUSE Linux Enterprise Server for SAP 15SP7	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Server for SAP 15SP7	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Desktop 15SP7	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S

Operating System	Hardware Platform
SUSE Linux Enterprise Desktop 15SP7	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Base Container Image 15SP7	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Base Container Image 15SP7	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Base Container Image 15SP7	GIGABYTE R152-P30 on Ampere® Altra® Q80-30
SUSE Linux Enterprise Base Container Image 15SP7	IBM z16 A01 on IBM® Telum(TM)
SUSE Linux Enterprise Base Container Image 15SP7	IBM LinuxONE III Model LT1 on z15
SUSE Linux Enterprise Server 15SP7	IBM LinuxONE III Model LT1 on z15
SUSE Linux Enterprise Server 15SP7	IBM z16 A01 on IBM® Telum(TM)
SUSE Linux Enterprise Server 15SP7	ASUS RS700-E11-RS4U on Intel® Xeon® Gold 5416S
SUSE Linux Enterprise Server 15SP7	SuperMicro SuperChassis 825BTQCR1K23LPB and Motherboard H12DSi-NT6 on AMD EPYC(TM) 7343
SUSE Linux Enterprise Server 15SP7	GIGABYTE R152-P30 on Ampere® Altra® Q80-30

Table 4: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid

The module is considered to maintain compliance with the FIPS 140-3 validation for SUSE products when operating on any general-purpose platform/processor that supports the SUSE Linux Enterprise Server operating system per the vendor affirmation from SUSE based on the allowance FIPS 140-3 management manual [FIPS140-3_MM] section 7.9.1 bullet 1 a i).

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Excluded Components

There are no components excluded from the requirements of the FIPS 140-3 standard.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved mode	Automatically entered whenever an approved service is requested	Approved	Equivalent to the indicator of the requested service
Non-approved mode	Automatically entered whenever a non-approved service is requested	Non-Approved	Equivalent to the indicator of the requested service

Table 5: Modes List and Description

After passing all pre-operational self-tests and cryptographic algorithm self-tests executed on start-up, the module automatically transitions to the approved mode. No operator intervention is required to reach this point. The module operates in the approved mode of operation by default and can only transition into the non-approved mode by calling one of the non-approved services listed in the Non-Approved Services table of the Security Policy.

In the operational state, the module accepts service requests from calling applications through its logical interfaces. At any point in the operational state, a calling application can end its process, causing the module to end its operation.

Mode Change Instructions and Status:

The module automatically switches between the approved and non-approved modes depending on the services requested by the operator. The status indicator of the mode of operation is equivalent to the indicator of the service that was requested.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Key Length - 128, 192, 256	SP 800-38C

Algorithm	CAVP Cert	Properties	Reference
AES-CFB1	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB128	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Generation Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A6075, A6076, A6077, A6078, A6079, A6080, A6081, A6082, A6083, A6100, A6101, A6102, A6106, A6107, A6108	Direction - Decrypt, Encrypt IV Generation - External, Internal Key Length - 128, 192, 256 IV Generation Mode - 8.2.1, 8.2.2	SP 800-38D
AES-GMAC	A6075, A6076, A6077, A6078, A6079, A6080, A6081, A6082, A6083, A6100, A6101, A6102, A6106, A6107, A6108	Direction - Decrypt, Encrypt IV Generation - External Key Length - 128, 192, 256	SP 800-38D
AES-KW	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F

Algorithm	CAVP Cert	Properties	Reference
AES-KWP	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-OFB	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E
Counter DRBG	A5553, A5554, A5555, A5556, A5557, A5558, A5659	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - No, Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6092, A6093, A6094, A6099, A6105, A6111	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA3-224, SHA3-256, SHA3-384, SHA3-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6092, A6093, A6094, A6099, A6105, A6111	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA3-224, SHA3-256, SHA3-384, SHA3-512	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-224	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-224	A6092, A6093, A6094, A6099, A6105, A6111	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-256	A6092, A6093, A6094, A6099, A6105, A6111	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A6092, A6093, A6094, A6099, A6105, A6111	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-512	A6092, A6093, A6094, A6099, A6105, A6111	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3

Algorithm	CAVP Cert	Properties	Reference
KAS-FFC-SSC Sp800-56Ar3	A6096	Domain Parameter Generation Methods - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF SP800-56Cr2	A6095	Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-3072 Increment 8 HMAC Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512	SP 800-56C Rev. 2
KDF SSH (CVL)	A6085, A6087, A6089, A6091, A6098, A6104	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
PBKDF	A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111	Iteration Count - Iteration Count: 1000-10000 Increment 1 Password Length - Password Length: 8-128 Increment 1	SP 800-132
RSA KeyGen (FIPS186-5)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Key Generation Mode - probable Modulo - 2048, 3072, 4096, 6144, 8192 Primality Tests - 2powSecStr Private Key Format - standard	FIPS 186-5
RSA SigGen (FIPS186-5)	A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5
RSA SigVer (FIPS186-5)	A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111	Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
Safe Primes Key Generation	A6096	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192	SP 800-56A Rev. 3
Safe Primes Key Verification	A6096	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192	SP 800-56A Rev. 3
SHA2-224	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/224	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA3-224	A6092, A6093, A6094, A6099, A6105, A6111	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-256	A6092, A6093, A6094, A6099, A6105, A6111	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202

Algorithm	CAVP Cert	Properties	Reference
SHA3-384	A6092, A6093, A6094, A6099, A6105, A6111	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-512	A6092, A6093, A6094, A6099, A6105, A6111	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHAKE-128	A6092, A6093, A6094, A6099, A6105, A6111	Output Length - Output Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A6092, A6093, A6094, A6099, A6105, A6111	Output Length - Output Length: 16-65536 Increment 8	FIPS 202
TLS v1.2 KDF RFC7627 (CVL)	A6084, A6086, A6088, A6090, A6097, A6103, A6109	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
TLS v1.3 KDF (CVL)	A6095	HMAC Algorithm - SHA2-256, SHA2-384 KDF Running Modes - DHE, PSK, PSK-DHE	SP 800-135 Rev. 1

Table 6: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
Asymmetric Cryptographic Key Generation (CKG)	Key type:Asymmetric	N/A	SP 800-133r2, section 4, example 1

Table 7: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
AES-GCM with external IV	Authenticated encryption
HMAC with less than 112-bit keys, SipHash	Message authentication code (MAC)
Diffie-Hellman with domain parameters other than safe primes	Key pair generation; Diffie-Hellman public key validation; Shared secret computation
DSA with any key sizes	Digital signature verification
EC Diffie-Hellman with P-192 curve, K curves, B curves and non-NIST curves	Shared secret computation
ECDSA with P-192 curve, K curves, B curves and non-NIST curves	Key pair generation; Digital signature generation; Digital signature verification
PBKDF with non-approved message digest algorithms or using input parameters not meeting requirements stated in section 2.7.3	Key derivation
RSA with keys smaller than 2048 bits	Key pair generation; Digital signature generation; Digital signature verification
RSA encryption with any key sizes	Key encapsulation
RSA decryption with any key sizes	Key un-encapsulation
TLS v1.0, v1.1 KDF	Key derivation
SHA-1	Message digest; Digital signature generation; Digital signature verification; Message authentication code (MAC); Key derivation

Table 8: Non-Approved, Not Allowed Algorithms

The table above lists all non-approved cryptographic algorithms of the module employed by the non-approved services of the Non-Approved Services table in Section 4.4 Non-Approved Services.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Message digest	SHA XOF	Message digest using SHA or SHAKE algorithms		SHA2-224: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) SHA2-256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110) SHA2-384: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) SHA2-512: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) SHA2-512/224: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) SHA2-512/256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) SHA3-224: (A6092, A6093, A6094, A6099, A6105, A6111) SHA3-256: (A6092, A6093, A6094, A6099, A6105, A6111) SHA3-384: (A6092, A6093, A6094, A6099, A6105, A6111) SHA3-512: (A6092,

Name	Type	Description	Properties	Algorithms
				A6093, A6094, A6099, A6105, A6111) SHAKE-128: (A6092, A6093, A6094, A6099, A6105, A6111) SHAKE-256: (A6092, A6093, A6094, A6099, A6105, A6111)
Encryption	BC-UnAuth	Encryption with AES		AES-CBC: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB1: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB128: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB8: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CTR: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-ECB: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-OFB: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-XTS Testing

Name	Type	Description	Properties	Algorithms
				Revision 2.0: (A5553, A5554, A5555, A5556, A5557, A5558, A5659)
Decryption	BC-UnAuth	Decryption with AES		AES-CBC: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB1: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB128: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CFB8: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CTR: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-ECB: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-OFB: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-XTS Testing Revision 2.0: (A5553, A5554, A5555, A5556, A5557, A5558, A5659)

Name	Type	Description	Properties	Algorithms
Authenticated encryption	BC-Auth	Authenticated encryption	CBC+HMAC:Key lengths: 128, 256 bits; Security strength: 128, 256 bits; Used as part of the cipher suites listed in Appendix A for the TLS protocol CCM, GCM:Key lengths: 128, 192, 256 bits; Security strength: 128, 192, 256 bits CCM, GCM (in TLS):Key lengths: 128, 256 bits; Security strength: 128, 256 bits; Used as part of the cipher suites listed in Appendix A for the TLS protocol	AES-CCM: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-GCM: (A6075, A6076, A6077, A6078, A6079, A6080, A6081, A6082, A6083, A6100, A6101, A6102, A6106, A6107, A6108) AES-KW: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-KWP: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CBC: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) HMAC-SHA2-256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110) HMAC-SHA2-384: (A6084, A6086, A6088, A6090, A6097, A6103, A6109)
Authenticated decryption	BC-Auth	Authenticated decryption	CBC+HMAC:Key lengths: 128, 256 bits; Security strength: 128, 256 bits; Used as part of the cipher suites listed in Appendix	AES-CCM: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-GCM: (A6075, A6076, A6077, A6078, A6079,

Name	Type	Description	Properties	Algorithms
			A for the TLS protocol CCM, GCM:Key lengths: 128, 192, 256 bits; Security strength: 128, 192, 256 bits CCM, GCM (in TLS):Key lengths: 128, 256 bits; Security strength: 128, 256 bits; Used as part of the cipher suites listed in Appendix A for the TLS protocol	A6080, A6081, A6082, A6083, A6100, A6101, A6102, A6106, A6107, A6108) AES-KW: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-KWP: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-CBC: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) HMAC-SHA2-256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110) HMAC-SHA2-384: (A6084, A6086, A6088, A6090, A6097, A6103, A6109)
Message authentication code (MAC)	MAC	Message authentication code computation using AES or HMAC		AES-CMAC: (A5553, A5554, A5555, A5556, A5557, A5558, A5659) AES-GMAC: (A6075, A6076, A6077, A6078, A6079, A6080, A6081, A6082, A6083, A6100, A6101, A6102, A6106, A6107, A6108)

Name	Type	Description	Properties	Algorithms
				HMAC-SHA2-224: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) HMAC-SHA2-256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109, A6110) HMAC-SHA2-384: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) HMAC-SHA2-512: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) HMAC-SHA2-512/224: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) HMAC-SHA2-512/256: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) HMAC-SHA3-224: (A6092, A6093, A6094, A6099, A6105, A6111) HMAC-SHA3-256: (A6092, A6093, A6094, A6099, A6105, A6111) HMAC-SHA3-384: (A6092, A6093, A6094, A6099, A6105, A6111)

Name	Type	Description	Properties	Algorithms
				HMAC-SHA3-512: (A6092, A6093, A6094, A6099, A6105, A6111)
Random number generation	DRBG	Random number generation using a SP 800-90A Rev. 1 compliant DRBG		Counter DRBG: (A5553, A5554, A5555, A5556, A5557, A5558, A5659)
Key pair generation	AsymKeyPair-KeyGen CKG	Key pair generation using RSA, ECDSA or Safe Primes	RSA:Key size: 2048-16384 bits; Security strength: 112-256 bits	RSA KeyGen (FIPS186-5): (A6084, A6086, A6088, A6090, A6097, A6103, A6109) ECDSA KeyGen (FIPS186-5): (A6084, A6086, A6088, A6090, A6097, A6103, A6109) Safe Primes Key Generation: (A6096) Asymmetric Cryptographic Key Generation (CKG): () Key type: Asymmetric
Key pair verification	AsymKeyPair-KeyVer	Key pair verification using ECDSA or Safe Primes		ECDSA KeyVer (FIPS186-5): (A6084, A6086, A6088, A6090, A6097, A6103, A6109) Safe Primes Key Verification: (A6096)

Name	Type	Description	Properties	Algorithms
Digital signature generation	DigSig-SigGen	Digital signature generation using RSA or ECDSA	RSA:Modulus Size: 2048-16384 bits; Security strength: 112-256 bits	RSA SigGen (FIPS186-5): (A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111) ECDSA SigGen (FIPS186-5): (A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111)
Digital signature verification	DigSig-SigVer	Digital signature verification using RSA or ECDSA	RSA (FIPS 186-5):Key size: 2048-16384 bits; Security strength: 112-256 bits	RSA SigVer (FIPS186-5): (A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111) ECDSA SigVer (FIPS186-5): (A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111)
Shared secret computation	KAS-SSC	Shared secret computation using DH or ECDH		KAS-FFC-SSC Sp800-56Ar3: (A6096) KAS-ECC-SSC

Name	Type	Description	Properties	Algorithms
				Sp800-56Ar3: (A6084, A6086, A6088, A6090, A6097, A6103, A6109)
Key derivation with HKDF	KAS-56CKDF	Key derivation using HKDF		KDA HKDF SP800-56Cr2: (A6095)
Key derivation with TLS	KAS-135KDF	Key derivation using TLS v1.2, v1.3	TLS v1.2 KDF RFC7627:Support: extended master secret	TLS v1.2 KDF RFC7627: (A6084, A6086, A6088, A6090, A6097, A6103, A6109) TLS v1.3 KDF: (A6095)
Key derivation with SSH KDF	KAS-135KDF	Key derivation using SSH KDF		KDF SSH: (A6085, A6087, A6089, A6091, A6098, A6104)
Password-based key derivation	PBKDF	Key derivation using PBKDF		PBKDF: (A6084, A6086, A6088, A6090, A6092, A6093, A6094, A6097, A6099, A6103, A6105, A6109, A6111)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 AES GCM IV

The AES GCM IV generation is in compliance with RFC 5288 and RFC 8446 and shall only be used for the TLS protocol version 1.2 and 1.3 to be compliant with FIPS 140-3 IG C.H, provisions 1 (“TLS protocol IV generation”) and 5 (“Provisions of an industry protocol supporting AES-GCM encryption, not included among the acceptable protocols in scenario 1”); in addition, the module is compliant with section 3.3.1 of SP 800-52 Rev. 2.

In both scenarios, the nonce_explicit part of the IV does not exhaust the maximum number of possible values for a given session key. The design of the TLS protocol in this module implicitly ensures that the nonce_explicit, or counter portion of the IV will not exhaust all of its possible values.

In the event the module's power is lost and restored, the consuming application must ensure that a new key for use with the AES GCM key encryption or decryption under this scenario shall be established.

When a GCM IV is used for decryption, the responsibility for the IV generation lies with the party that performs the AES GCM encryption.

2.7.2 AES XTS

The length of a single data unit encrypted or decrypted with AES XTS shall not exceed 2^{20} AES blocks, that is 16MB, of data per XTS instance. An XTS instance is defined in Section 4 of SP 800-38E.

To meet the requirement stated in IG C.I, the module implements a check that ensures, before performing any cryptographic operation, that the two AES keys used in AES XTS mode are not identical. As the module does not generate symmetric keys, the check is performed when keys are input the service APIs.

Key_1 and Key_2 shall be generated and/or established independently according to the rules for component symmetric keys from NIST SP 800-133rev2, Sec. 6.3.

The XTS mode shall only be used for the cryptographic protection of data on storage devices. It shall not be used for other purposes, such as the encryption of data in transit.

2.7.3 Key Derivation using SP 800-132 PBKDF2

The module provides password-based key derivation (PBKDF2), compliant with SP 800-132. The module supports option 1a from Section 5.4 of SP 800-132, in which the Master Key (MK) or a segment of it is used directly as the Data Protection Key (DPK).

In accordance to SP 800-132 and FIPS 140-3 IG D.N, the following requirements are met:

- Derived keys shall be used only for storage applications, and shall not be used for any other purposes. The length of the MK or DPK is 112 bits or more.
- Passwords or passphrases, used as an input for the PBKDF2, shall not be used as cryptographic keys.
- The minimum length of the password or passphrase accepted by the module is 20 characters. The probability of guessing the value, assuming a worst-case scenario of all digits, is estimated to be at most 10^{-20}. Combined with the minimum iteration count as described below, this provides an acceptable trade-off between user experience and security against brute-force attacks.
- A portion of the salt shall be generated randomly using the SP 800-90A Rev. 1 DRBG provided by the module. The minimum length required is 128 bits.
- The iteration count shall be selected as large as possible, as long as the time required to generate the key using the entered password is acceptable for the users. The minimum value accepted by the module is 1000.

If any of these requirements are not met, the requested service is non-approved (see Non-Approved Services table in Section 4.4 Non-Approved Services).

2.7.4 SP 800-56A Rev. 3 Assurances

To comply with the assurances found in Section 5.6.2 of SP 800-56A Rev. 3, the operator must use the module in the context of the TLS or SSH protocols. Additionally, the module's approved key pair generation service (see Approved Services table in Section 4.3 Approved Services) must be used to generate ephemeral Diffie-Hellman or EC Diffie-Hellman key pairs, or the key pairs must be obtained from another FIPS-validated module. As part of this service, the module will internally perform the full public key validation of the generated public key.

The module's shared secret computation service will internally perform the full public key validation of the peer public key, complying with Sections 5.6.2.2.1 and 5.6.2.2.2 of SP 800-56A Rev. 3.

2.7.5 RSA Signatures

Approved moduli for 2048, 3072, and 4096 bits are CAVP tested in compliance with FIPS 186-5 for RSA key generation, signature generation, and signature verification.

All other RSA moduli mentioned in the Security Function Implementation table in Section 2.6 Security Function Implementations and not mentioned above cannot be tested by CAVP but are approved for RSA key generation, signature generation, and signature verification in IG C.F.

2.8 RBG and Entropy

Cert Number	Vendor Name
E209	SUSE LLC

Table 10: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
SUSE OpenSSL1 CPU Time Jitter RNG Entropy Source	Non-Physical	SUSE Linux Enterprise Server 15 SP6 on AMD EPYC(TM) 7343; SUSE Linux Enterprise Server 15 SP6 on Ampere® Altra® Q80-30; SUSE Linux Enterprise Server 15 SP6 on Intel® Xeon® Gold 5416S; SUSE Linux Enterprise Server 15 SP6 on IBM® Telum(TM)	256 bits	full entropy	SHA3-256 (A5411); AES-256-CTR-DRBG (A5555); AES-256-CTR-DRBG (A5558)

Table 11: Entropy Sources

As shown in the Figure 1 of the PUD for cert #E209, the noise source along with first conditioning component of SHA-3 is implemented by the Userspace CPU Time Jitter RNG library implemented inside the OE but outside of the module's cryptographic boundary. The second conditioning component DRBG (AES-256-CTR-

DRBG (A5555, A5558)) is implemented within the module's cryptographic boundary. This DRBG is used internally by the module (e.g. to generate seeds for asymmetric key pairs and random numbers for security functions). It can also be accessed using the specified API functions.

The DRBG is initialized during module initialization; the module loads by default the DRBG using the CTR_DRBG mechanism with AES-256, with derivation function, and without prediction resistance.

As per the Public document of entropy certificate E209, the entropy source provides full entropy of 256 bits.

The operational environment on the ESV certificate is identical to the operational environment listed in this document. There are no maintenance requirements for the entropy source.

2.9 Key Generation

The module implements Cryptographic Key Generation (CKG, vendor affirmed), compliant with SP 800-133 Rev. 2 as listed in the Security Function Implementation table in Section 2.6.

When random values are required, they are obtained from the SP 800-90Ar1 approved DRBG, compliant with Section 4 of SP 800-133 Rev. 2. Additionally, the module implements key derivation as listed in the Security Function Implementation table in Section 2.6.

Intermediate key generation values are not output from the module and are explicitly zeroized after processing the service.

2.10 Key Establishment

The module implements shared secret computation and key transport methods as listed in the Section 2.6 Security Function Implementations.

2.11 Industry Protocols

The module implements KDF for the TLS protocol TLSv1.2, TLSv1.3.

No parts of the TLS 1.2/1.3, other than the key derivation functions mentioned above, have been tested by the CAVP and CMVP.

The module implements HKDF for the TLS protocol TLSv1.3. The implementation of the KDA HKDF has been tested by the CAVP and CMVP.

AES-GCM with internal IV generation is offered in the approved mode compliant with TLS 1.2 and TLS 1.3 (RFC 5288 and RFC 8446). This functionality shall only be used in conjunction with the TLS protocol.

The module implements the SSH key derivation function for use in the SSH protocol (RFC 4253 and RFC 6668). No parts of the SSH protocol, other than those mentioned above, have been tested by the CAVP and CMVP.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	API input parameters, kernel I/O network or files on filesystem, TLS protocol input messages.
N/A	Data Output	API output parameters, kernel I/O network or files on filesystem, TLS protocol output messages.
N/A	Control Input	API function calls, API input parameters for control
N/A	Status Output	API return codes, API output parameters for status output

Table 12: Ports and Interfaces

The logical interfaces are the APIs through which the applications request services. These logical interfaces are logically separated from each other by the API design. The module does not implement a control output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

The module does not support authentication methods.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 13: Roles

The module does not support multiple concurrent operators.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Message digest	Used to generate a SHA-2, or SHA-3/SHAKE message digest	fips_sli_SHA*_is_approved returns 1	Message	Message digest	Message digest	Crypto Officer
Encryption	Perform AES encryption	fips_sli_is_approved_EVP_CIPHER_CTX returns 1	Plaintext, AES key, IV	Ciphertext	Encryption	Crypto Officer - AES key: W,E
Decryption	Perform AES decryption	fips_sli_is_approved_EVP_CIPHER_CTX returns 1	Ciphertext, AES key, IV	Plaintext	Decryption	Crypto Officer - AES key: W,E
Authenticated encryption	Perform authenticated encryption	fips_sli_is_approved_EVP_CIPHER_CTX returns 1	Plaintext, AES key, IV	Ciphertext, MAC tag	Authenticated encryption	Crypto Officer - AES key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Authenticated decryption	Perform authenticated decryption	fips_sli_is_approved_EVP_CIPHER_CTX returns 1	Ciphertext, AES key, IV, MAC tag	Plaintext or failure	Authenticated decryption	Crypto Officer - AES key: W,E
Message authentication code (MAC)	Compute a MAC tag using AES or HMAC	fips_sli_is_approved_CMAC_CTX, fips_sli_HMAC_is_approved, or fips_sli_is_approved_EVP_CIPHER_CTX return 1	Message, AES key or HMAC key	MAC tag	Message authentication code (MAC)	Crypto Officer - AES key: W,E - HMAC key: W,E
Random number generation	Generate random bytes using CTR-DRBG	fips_sli_RAND_bytes_is_approved or fips_sli_RAND_priv_bytes_is_approved returns 1	Output length	Random bytes	Random number generation	Crypto Officer - Entropy input: W,E,Z - DRBG seed: G,E,Z - DRBG internal state (V value, Key): G,W,E
Key pair generation	Generate an asymmetric key pair	fips_sli_is_approved_EVP_PKEY_CTX returns 1	Group or Curve or Modulus bits	DH key pair; EC key pair; RSA key pair	Key pair generation Random number generation	Crypto Officer - Module-generated RSA private key: G,R - Module-generated RSA public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						key: G,R - Module-generated DH private key: G,R - Module-generated DH public key: G,R - Module-generated EC private key: G,R - Module-generated EC public key: G,R - Intermediate key generation value: G,E,Z - DRBG internal state (V value, Key): W,E
Key pair verification	Verify a generated asymmetric key pair	fips_sli_is_approved_EVP_PKEY_CTX returns 1	Safe Primes key pair	Pass/Fail	Key pair verification	Crypto Officer - DH private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			or EC key pair			key: W,E - DH public key: W,E - EC private key: W,E - EC public key: W,E
Digital signature generation	Generate a digital signature	fips_sli_is_approved_EVP_PKEY_CTX returns 1	Message, private key	Signature	Digital signature generation Random number generation	Crypto Officer - RSA private key: W,E - EC private key: W,E - DRBG internal state (V value, Key): W,E
Digital signature verification	Verify a digital signature	fips_sli_is_approved_EVP_PKEY_CTX returns 1	Message, public key, signature	Pass/Fail	Digital signature verification	Crypto Officer - RSA public key: W,E - EC public key: W,E
Shared secret computation	Compute a shared secret	fips_sli_is_approved_EVP_PKEY_CTX returns 1	Private key, public key (peer)	Shared secret	Shared secret computation	Crypto Officer - DH private key: W,E - DH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: W,E - EC private key: W,E - EC public key: W,E - Shared secret: G,R
Key derivation with HKDF	Key derivation with KDA HKDF in the context of TLS v1.3	fips_sli_is_approved_EVP_KDF_CTX returns 1	Shared secret	HKDF derived key	Key derivation with HKDF	Crypto Officer - Shared secret: W,E - HKDF derived key: G,R
Key derivation with TLS	Perform key derivation using TLS KDF	fips_sli_is_approved_EVP_KDF_CTX returns 1	TLS pre-master secret	TLS derived key	Key derivation with TLS	Crypto Officer - TLS pre-master secret: W,E - TLS master secret: E,G,Z - TLS derived key: G,R
Key derivation with SSH	Perform key derivation with SSH KDF	fips_sli_is_approved_EVP_KDF_CTX returns 1	Shared secret	SSH derived key	Key derivation with SSH KDF	Crypto Officer - Shared secret: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH derived key: G,R
Key derivation from a password	Derive a key from a password or passphrase	fips_sli_PKCS5_PBKDF2_HMAC returns 1	Password or passphrase	PBKDF Derived key	Password-based key derivation	Crypto Officer - Password or passphrase: W,E - PBKDF derived key: G,R
Transport Layer Security (TLS) Network Protocol	Provide supported cipher suites (listed in Appendix A) in approved mode	SSL_CIPHER_get_protocol_ id or SSL_get_current_cipher return a two-byte ID matching an approved cipher suite (listed in Appendix A)	Cipher-suites listed in Appendix A, Digital Certificate, Public and Private Keys, Application Data	Return codes and/or log messages, Application data	Message digest Authenticated encryption Authenticated decryption Message authentication code (MAC) Key pair generation Key pair verification Digital signature generation Digital signature verification Shared secret	Crypto Officer - RSA private key: W,E - RSA public key: W,E - EC private key: W,E - EC public key: W,E - TLS pre-master secret: G,E - TLS master secret: G,E,Z - DH private key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					computation Key derivation with HKDF Key derivation with TLS	G,W,E - DH public key: G,W,E - TLS derived key: G,W,E - HKDF derived key: G,W,E
Self-test	Perform CASTs and integrity test	None	N/A	Pass/fail result of self-tests	Message digest Encryption Decryption Authenticated encryption Authenticated decryption Message authentication code (MAC) Random number generation Digital signature generation Digital signature verification Shared secret	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					computation Key derivation with HKDF Key derivation with SSH KDF Key derivation with TLS Password-based key derivation	
Show status	Show the current status of the module	None	N/A	Module status	None	Crypto Officer
Show module name and version	Show module name and the version of the module	None	N/A	Name and version information	None	Crypto Officer
Zeroization	Zeroize SSPs	None	Any SSP	N/A	None	Crypto Officer - AES key: Z - HMAC key: Z - Module-generated DH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						private key: Z - Module-generated DH public key: Z - Module-generated RSA private key: Z - Module-generated RSA public key: Z - Module-generated EC private key: Z - Module-generated EC public key: Z - DH private key: Z - DH public key: Z - RSA private key: Z - RSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: Z - EC private key: Z - EC public key: Z - Shared secret: Z - Password or passphrase: Z - HKDF derived key: Z - SSH derived key: Z - TLS derived key: Z - PBKDF derived key: Z - Entropy input: Z - DRBG seed: Z - DRBG internal state (V value, Key): Z - Intermediate key generation value: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS pre-master secret: Z - TLS master secret: Z

Table 14: Approved Services

The module provides services to operators that assume the available role. All services are described in detail in the API documentation (manual pages). The convention below applies when specifying the access permissions (types) that the service has for each SSP.

- **Generate (G)**: The module generates or derives the SSP.
- **Read (R)**: The SSP is read from the module (e.g. the SSP is output).
- **Write (W)**: The SSP is updated, imported, or written to the module.
- **Execute (E)**: The module uses the SSP in performing a cryptographic operation.
- **Zeroize (Z)**: The module zeroizes the SSP.
- **N/A**: The module does not access any SSP or key during its operation.

The "Indicator" column shows the service indicator API functions that must be used to verify the service indicator for each of the services. A value of 1 indicates that the service is approved, and 0 indicates that the service is non-approved.

Additionally there is a separate indicator used for the following services:

- The API function used to determine the indicator for the "TLS network protocol" service returns the cipher suite established for the TLS session. If the returned cipher suite ID belongs to one of the cipher suites listed in Appendix A, then the service is approved, otherwise, it is non-approved.

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Message digest	Compute a message digest	SHA-1	CO
Authenticated encryption	Perform authenticated encryption using AES-GCM with an externally provided IV	AES-GCM with external IV	CO

Name	Description	Algorithms	Role
Message authentication code (MAC)	Compute a MAC tag	HMAC with less than 112-bit keys, SipHash SHA-1	CO
Shared secret computation	Perform shared secret computation	Diffie-Hellman with domain parameters other than safe primes EC Diffie-Hellman with P-192 curve, K curves, B curves and non-NIST curves	CO
Signature generation	Generate a digital signature	ECDSA with P-192 curve, K curves, B curves and non-NIST curves RSA with keys smaller than 2048 bits SHA-1	CO
Signature verification	Verify a digital signature	DSA with any key sizes ECDSA with P-192 curve, K curves, B curves and non-NIST curves RSA with keys smaller than 2048 bits SHA-1	CO
Key pair generation	Generate a key pair	Diffie-Hellman with domain parameters other than safe primes ECDSA with P-192 curve, K curves, B curves and non-NIST curves RSA with keys smaller than 2048 bits	CO
Key derivation	Derive a symmetric key	PBKDF with non-approved message digest algorithms or using input parameters not meeting requirements stated in section 2.7.3 TLS v1.0, v1.1 KDF SHA-1	CO
Key encapsulation	Encapsulate a symmetric key using RSA	RSA encryption with any key sizes	CO
Key un-encapsulation	Un-encapsulate a symmetric key using RSA	RSA decryption with any key sizes	CO
Public key validation	Validate a public key	Diffie-Hellman with domain parameters other than safe primes	CO

Table 15: Non-Approved Services

4.5 External Software/Firmware Loaded

The module does not load external software or firmware.

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module is verified by comparing a HMAC-SHA2-256 value calculated at run time with the HMAC-SHA2-256 value that was computed at build time (stored in the .hmac file) for each software component. The MAC key is hardcoded in the module.

5.2 Initiate on Demand

Integrity tests are performed as part of the pre-operational self-tests, which are executed when the module is initialized. The integrity test may be invoked on-demand by unloading and subsequently re-initializing the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied:

Any SSPs contained within the module are protected by the process isolation and memory separation mechanisms, and only the module has control over these SSPs.

If properly installed, the operating system provides process isolation and memory protection mechanisms that ensure appropriate separation for memory access among the processes on the system. Each process has control over its own data and uncontrolled access to the data of other processes is prevented.

6.2 Configuration Settings and Restrictions

The module shall be installed as stated in Section 11 Life-Cycle Assurance.

Instrumentation tools like the ptrace system call, gdb and strace, userspace live patching, as well as other tracing mechanisms offered by the Linux environment such as ftrace or systemtap, shall not be used in the operational environment. The use of any of these tools implies that the cryptographic module is running in a non-validated operational environment.

7 Physical Security

The module is comprised of software only and therefore this section is not applicable.

8 Non-Invasive Security

This module does not implement any non-invasive security mechanisms, and therefore this section is not applicable.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	Temporary storage for SSPs used by the module as part of service execution.	Dynamic

Table 16: Storage Areas

The module does not perform persistent storage of SSPs. The SSPs are temporarily stored in the RAM in plaintext form. SSPs are stored until they are zeroized by the operator (using a zeroization call or removing power from the module) or zeroized automatically.

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API input parameters	Operator calling application (TOEPP)	Cryptographic module	Plaintext	Manual	Electronic	
API output parameters	Cryptographic module	Operator calling application (TOEPP)	Plaintext	Manual	Electronic	

Table 17: SSP Input-Output Methods

The module only supports SSP entry and output to and from the calling application running on the same operational environment. This corresponds to manual distribution, electronic entry/output (“CM Software to/from App via TOEPP Path”) per FIPS 140-3 IG 9.5.A Table 1.

There is no entry or output of cryptographically protected SSPs.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Free cipher handle	Zeroizes the SSPs contained within the cipher handle	Memory occupied by SSPs is overwritten with zeros, which renders the SSP values irretrievable. The successful completion of the zeroization routine indicates that the zeroization procedure succeeded.	By calling the appropriate zeroization functions: EVP_CIPHER_CTX_free, EVP_CIPHER_CTX_reset, HMAC_CTX_free, RSA_free, EC_KEY_free, DH_free, EVP_PKEY_free, FIPS_drbg_free, SSL_free, SSL_clear
Automatic	Automatically zeroized by the module when no longer needed	Memory occupied by SSPs is overwritten with zeroes, which renders the SSP values irretrievable. The successful completion of the running service indicates that zeroization has completed.	N/A
Module reset	De-allocates the volatile memory used to store SSPs	Volatile memory used by the module is overwritten within nanoseconds when the module is unloaded. The successful completion of the module reset indicates that zeroization has completed.	By unloading and reloading the module

Table 18: SSP Zeroization Methods

The application that uses the module is responsible for the appropriate zeroization of SSPs. The module provides key allocation and destruction functions, which overwrites the memory occupied by the SSP´s information with zeros before its deallocation.

Calling the SSL_free() and SSL_clear() will zeroize the SSPs stored in the TLS protocol internal state and also invoke the corresponding API functions listed in Table 18 to zeroize SSPs.

All data output is inhibited during zeroization.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES key	AES key used for encryption, decryption, and computing MAC tags	AES-XTS: 128, 256 bits; Other modes: 128, 192, 256 bits - AES-XTS: 128, 256 bits; Other modes: 128, 192, 256 bits	Symmetric key - CSP			Encryption Decryption Authenticated encryption Authenticated decryption Message authentication code (MAC)
HMAC key	HMAC key used for computing MAC tags	112-524288 bits - 112-256 bits	Authentication key - CSP			Authenticated encryption Authenticated decryption Message authentication code (MAC)
Module-generated RSA private key	RSA private key generated by the module	2048-16384 bits - 112-256 bits	Private key - CSP	Key pair generation		Key pair generation
Module-generated RSA public key	RSA public key generated by the module	2048-16384 bits - 112-256 bits	Public key - PSP	Key pair generation		Key pair generation
RSA private key	RSA private key written to the module	2048-16384 bits - 112-256 bits	Private key - CSP			Digital signature generation
RSA public key	RSA public key written to the module	2048-16384 bits - 112-256 bits	Public key - PSP			Digital signature verification

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Module-generated DH private key	DH private key generated by the module	2048-8192 bits - 112-200 bits	Private key - CSP	Key pair generation		Key pair generation
Module-generated DH public key	DH public key generated by the module	2048-8192 bits - 112-200 bits	Public key - PSP	Key pair generation		Key pair generation
DH private key	DH private key written to the module	2048-8192 bits - 112-200 bits	Private key - CSP			Key pair verification Shared secret computation
DH public key	DH public key written to the module	2048-8192 bits - 112-200 bits	Public key - PSP			Key pair verification Shared secret computation
Module-generated EC private key	EC private key generated by the module	P-224, P-256, P-384, P-521 bits - 112, 128, 192, 256 bits	Private key - CSP	Key pair generation		Key pair generation
Module-generated EC public key	EC public key generated by the module	P-224, P-256, P-384, P-521 bits - 112, 128, 192, 256 bits	Public key - PSP	Key pair generation		Key pair generation
EC private key	EC private key written to the module	P-224, P-256, P-384, P-521 bits - 112, 128, 192, 256 bits	Private key - CSP			Key pair verification Digital signature generation Shared secret computation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
EC public key	EC public key written to the module	P-224, P-256, P-384, P-521 bits - 112, 128, 192, 256 bits	Public key - PSP			Key pair verification Digital signature verification Shared secret computation
Shared secret	Shared secret generated by ECDH or DH shared secret computation	224-8912 bits - 112-256 bits	Shared Secret - CSP		Shared secret computation	Shared secret computation Key derivation with HKDF Key derivation with SSH KDF
Password or passphrase	Password or passphrase used by PBKDF to derive symmetric keys	20-128 characters - N/A	Password - CSP			Password-based key derivation
TLS pre-master secret	Used to derive the master secret in the TLS protocol	224-8912 bits - 112-256 bits	Shared secret - CSP		Shared secret computation	Key derivation with TLS
TLS master secret	Derived from the pre-master secret using the TLS KDF per SP 800-135 Rev. 1	384 bits - 128-256 bits	Master secret - CSP	Key derivation with TLS		Key derivation with TLS
TLS derived key	Generated using the TLS v1.2, v1.3 KDF	112-4096 bits - 112-256 bits	Symmetric key - CSP	Key derivation with TLS		Key derivation with TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
HKDF derived key	Derived using the HKDF	112-4096 bits - 112-256 bits	Symmetric key - CSP	Key derivation with HKDF		Key derivation with HKDF
SSH derived key	Generated using the SSH KDF	112-4096 bits - 112-256 bits	Symmetric key - CSP	Key derivation with SSH KDF		Key derivation with SSH KDF
PBKDF derived key	Derived using the PBKDF	112-4096 bits - 112-256 bits	Symmetric key - CSP	Password-based key derivation		Password-based key derivation
Entropy input	Entropy input string used to seed the DRBG (IG D.L compliant)	128-384 bits - 128-256 bits	Entropy input - CSP			Random number generation
DRBG seed	DRBG seed derived from entropy input (IG D.L compliant)	256, 320, 384 bits - 128, 192, 256 bits	Seed - CSP	Random number generation		Random number generation
DRBG internal state (V value, Key)	Internal state of the CTR_DRBG	256, 320, 384 bits - 128, 192, 256 bits	Internal state - CSP	Random number generation		Random number generation
Intermediate key generation value	Intermediate key generation value generated during key pair generation (SP 800-133	112-16384 bits - 112-256 bits	Intermediate value - CSP	Key pair generation		Key pair generation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	Rev. 2 Section 4, 5.1, and 5.2)					

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	
HMAC key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	
Module-generated RSA private key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated RSA public key:Paired With Intermediate key generation value:Generated From
Module-generated RSA public key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated RSA private key:Paired With Intermediate key generation value:Generated From
RSA private key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	RSA public key:Paired With
RSA public key	API input parameters	RAM:Plaintext	From service invocation until	Free cipher handle	RSA private key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
			cipher handle is freed	Module reset	
Module-generated DH private key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated DH public key:Paired With Intermediate key generation value:Generated From
Module-generated DH public key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated DH private key:Paired With Intermediate key generation value:Generated From
DH private key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	DH public key:Paired With TLS pre-master secret:Derivation of Shared secret:Derives
DH public key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	DH private key:Paired With TLS pre-master secret:Derivation of Shared secret:Derives
Module-generated EC private key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated EC public key:Paired With Intermediate key generation value:Generated From
Module-generated EC public key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Module-generated EC private key:Paired With Intermediate key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					generation value:Generated From
EC private key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	EC public key:Paired With TLS pre-master secret:Derivation of Shared secret:Derives
EC public key	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	EC private key:Paired With TLS pre-master secret:Derivation of Shared secret:Derives
Shared secret	API input parameters API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	DH private key:Established By DH public key:Established By EC private key:Established By EC public key:Established By HKDF derived key:Derives SSH derived key:Derives
Password or passphrase	API input parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	PBKDF derived key:Derives
TLS pre-master secret		RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	DH private key:Established By DH public key:Established By EC private key:Established By EC public key:Established By

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					TLS master secret:Derives
TLS master secret		RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	TLS pre-master secret:Derived From TLS derived key:Derives
TLS derived key		RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	TLS master secret:Derived From
HKDF derived key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Shared secret:Derived From
SSH derived key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Shared secret:Derived From
PBKDF derived key	API output parameters	RAM:Plaintext	From service invocation until cipher handle is freed	Free cipher handle Module reset	Password or passphrase:Derived From
Entropy input		RAM:Plaintext	From generation until DRBG seed is created	Automatic Module reset	DRBG seed:Derives
DRBG seed		RAM:Plaintext	While the DRBG is instantiated	Automatic Module reset	Entropy input:Derived From DRBG internal state (V value, Key):Generates
DRBG internal state (V value, Key)		RAM:Plaintext	From DRBG instantiation	Free cipher handle	DRBG seed:Generated From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
			until DRBG termination	Module reset	
Intermediate key generation value		RAM:Plaintext	From service invocation until cipher handle is freed	Automatic Module reset	Module-generated RSA private key:Generates Module-generated RSA public key:Generates Module-generated ECDSA private key:Generates Module-generated ECDSA public key:Generates Module-generated DH private key:Generates Module-generated DH public key:Generates

Table 20: SSP Table 2

9.5 Transitions

Not applicable.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256	256-bit key	Message authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test of the shared library component of the module. Verified by comparing an HMAC SHA-256 value calculated at run time with the HMAC SHA-256 that was computed at build time

Table 21: Pre-Operational Self-Tests

The pre-operational software integrity tests are performed automatically when the module is initialized, before the module transitions into the operational state. While the module is executing the self-tests, services are not available, and data output (via the data output interface) is inhibited until the tests are successfully completed. The module transitions to the operational state only after the pre-operational self-tests are passed successfully.

Prior the first use, a CAST is executed for the algorithms used in the Pre-operational Self-Tests.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-256 (A6084)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6086)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6088)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-256 (A6090)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6097)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6103)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6109)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-256 (A6110)	72-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6084)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6086)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6088)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-512 (A6090)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6097)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6103)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA2-512 (A6109)	120-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-256 (A6092)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-256 (A6093)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-256 (A6094)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-256 (A6099)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA3-256 (A6105)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-256 (A6111)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6092)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6093)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6094)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6099)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6105)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHA3-512 (A6111)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHAKE-128 (A6092)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-128 (A6093)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-128 (A6094)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-128 (A6099)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-128 (A6105)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-128 (A6111)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-256 (A6092)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-256 (A6093)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHAKE-256 (A6094)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-256 (A6099)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-256 (A6105)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
SHAKE-256 (A6111)	144-bit message	KAT	CAST	Module becomes operational	Message Digest	Test runs at power-on before the integrity test
AES-ECB (A5553) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5554) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5555) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5556) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A5557) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5558) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5659) - Encrypt	128-bit key, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-ECB (A5553) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-ECB (A5554) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-ECB (A5555) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-ECB (A5556) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-ECB (A5557) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A5558) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-ECB (A5659) - Decrypt	128-bit key, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-CCM (A5553) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5554) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5555) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5556) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5557) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5558) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CCM (A5659) - Encrypt	192-bit key, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-CCM (A5553) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5554) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5555) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5556) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5557) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5558) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-CCM (A5659) - Decrypt	192-bit key, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-CMAC (A5553)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5554)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5555)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5556)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5557)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5558)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-CMAC (A5659)	128-, 192-, 256-bit keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
AES-GCM (A6075) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A6076) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6077) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6078) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6079) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6080) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6081) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6082) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6083) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A6100) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6101) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6102) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6106) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6107) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6108) - Encrypt	256-bit key, 96-bit IV, encrypt	KAT	CAST	Module becomes operational	Authenticated encryption	Test runs at power-on before the integrity test
AES-GCM (A6075) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6076) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A6077) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6078) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6079) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6080) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6081) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6082) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6083) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6100) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A6101) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6102) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6106) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6107) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-GCM (A6108) - Decrypt	256-bit key, 96-bit IV, decrypt	KAT	CAST	Module becomes operational	Authenticated decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5553) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5554) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
(A5555) - Encrypt						
AES-XTS Testing Revision 2.0 (A5556) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5557) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5558) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5659) - Encrypt	256, 512-bit keys, encrypt	KAT	CAST	Module becomes operational	Encryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5553) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5554) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-XTS Testing Revision 2.0 (A5555) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5556) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5557) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5558) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
AES-XTS Testing Revision 2.0 (A5659) - Decrypt	256, 512-bit keys, decrypt	KAT	CAST	Module becomes operational	Decryption	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6084)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6086)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-224 (A6088)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6090)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6097)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6103)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A6109)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6084)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6086)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6088)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-256 (A6090)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6097)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6103)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6109)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A6110)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6084)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6086)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6088)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-384 (A6090)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6097)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6103)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A6109)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6084)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6086)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6088)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6090)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-512 (A6097)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6103)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A6109)	160-bit key	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A6092)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A6093)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A6094)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A6099)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A6105)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA3-224 (A6111)	224, 1152, 1376 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6092)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6093)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6094)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6099)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6105)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A6111)	384, 1088, 1344 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A6092)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA3-384 (A6093)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A6094)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A6099)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A6105)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A6111)	384, 832, 1216 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A6092)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A6093)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A6094)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA3-512 (A6099)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A6105)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A6111)	512, 576, 1088 bits keys	KAT	CAST	Module becomes operational	Message authentication code computation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6084)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6086)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6088)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6090)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6092)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigGen (FIPS186-5) (A6093)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6094)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6097)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6099)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6103)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6105)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6109)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5) (A6111)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigVer (FIPS186-5) (A6084)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6086)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6088)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6090)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6092)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6093)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6094)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6097)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigVer (FIPS186-5) (A6099)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6103)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6105)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6109)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5) (A6111)	2048-bit key with SHA-256, with PKCS#1 v1.5 and PSS padding	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6084)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6086)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6088)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA SigGen (FIPS186-5) (A6090)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6092)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6093)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6094)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6097)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6099)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6103)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6105)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA SigGen (FIPS186-5) (A6109)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5) (A6111)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6084)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6086)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6088)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6090)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6092)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6093)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA SigVer (FIPS186-5) (A6094)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6097)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6099)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6103)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6105)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6109)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5) (A6111)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power-on before the integrity test
KAS-FFC-SSC Sp800-56Ar3 (A6096)	ffdhe2048	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KAS-ECC-SSC Sp800-56Ar3 (A6084)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6086)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6088)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6090)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6097)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6103)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
KAS-ECC-SSC Sp800-56Ar3 (A6109)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6084)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
TLS v1.2 KDF RFC7627 (A6086)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6088)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6090)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6097)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6103)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.2 KDF RFC7627 (A6109)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
TLS v1.3 KDF (A6095)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDA HKDF SP800-56Cr2 (A6095)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KDF SSH (A6085)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDF SSH (A6087)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDF SSH (A6089)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDF SSH (A6091)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDF SSH (A6098)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
KDF SSH (A6104)	SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6084)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6086)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
PBKDF (A6088)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6090)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6092)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6093)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6094)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6097)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6099)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
PBKDF (A6103)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6105)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6109)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
PBKDF (A6111)	24 characters password, 288-bit salt, 4096 iterations, SHA2-256	KAT	CAST	Module becomes operational	Key derivation	Test runs at power-on before the integrity test
Counter DRBG (A5553)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Counter DRBG (A5554)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Counter DRBG (A5555)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Counter DRBG (A5556)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Counter DRBG (A5557)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Counter DRBG (A5558)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Counter DRBG (A5659)	CTR-DRBG with AES with 256-bit keys with and without DF, with and without PR	KAT	CAST	Module becomes operational	Random number generation	Test runs at power-on before the integrity test
Safe Primes Key Generation (A6096)	N/A	PCT	PCT	Key pair generation is successfull	PCT according to SP 800-56A Rev.3, Section 5.6.2.1.4	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6084)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6086)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA KeyGen (FIPS186-5) (A6088)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6090)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6097)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6103)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
ECDSA KeyGen (FIPS186-5) (A6109)	SHA2-256	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6084)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6086)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6088)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6090)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA KeyGen (FIPS186-5) (A6097)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6103)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation
RSA KeyGen (FIPS186-5) (A6109)	SHA2-256 with PKCS#1 v1.5 and PSS paddings	PCT	PCT	Key pair generation is successfull	Signature generation and verification	Key pair generation

Table 22: Conditional Self-Tests

Data output through the data output interface is inhibited during the conditional self-tests. The module does not return control to the calling application until the tests are completed. If any of these tests fails, the module transitions to the error state (Section 10.4 Error States).

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256	Message authentication	SW/FW Integrity	On demand	Manually

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-256 (A6084)	KAT	CAST	On demand	Manually
SHA2-256 (A6086)	KAT	CAST	On demand	Manually
SHA2-256 (A6088)	KAT	CAST	On demand	Manually
SHA2-256 (A6090)	KAT	CAST	On demand	Manually
SHA2-256 (A6097)	KAT	CAST	On demand	Manually
SHA2-256 (A6103)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHA2-256 (A6109)	KAT	CAST	On demand	Manually
SHA2-256 (A6110)	KAT	CAST	On demand	Manually
SHA2-512 (A6084)	KAT	CAST	On demand	Manually
SHA2-512 (A6086)	KAT	CAST	On demand	Manually
SHA2-512 (A6088)	KAT	CAST	On demand	Manually
SHA2-512 (A6090)	KAT	CAST	On demand	Manually
SHA2-512 (A6097)	KAT	CAST	On demand	Manually
SHA2-512 (A6103)	KAT	CAST	On demand	Manually
SHA2-512 (A6109)	KAT	CAST	On demand	Manually
SHA3-256 (A6092)	KAT	CAST	On demand	Manually
SHA3-256 (A6093)	KAT	CAST	On demand	Manually
SHA3-256 (A6094)	KAT	CAST	On demand	Manually
SHA3-256 (A6099)	KAT	CAST	On demand	Manually
SHA3-256 (A6105)	KAT	CAST	On demand	Manually
SHA3-256 (A6111)	KAT	CAST	On demand	Manually
SHA3-512 (A6092)	KAT	CAST	On demand	Manually
SHA3-512 (A6093)	KAT	CAST	On demand	Manually
SHA3-512 (A6094)	KAT	CAST	On demand	Manually
SHA3-512 (A6099)	KAT	CAST	On demand	Manually
SHA3-512 (A6105)	KAT	CAST	On demand	Manually
SHA3-512 (A6111)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
SHAKE-128 (A6092)	KAT	CAST	On demand	Manually
SHAKE-128 (A6093)	KAT	CAST	On demand	Manually
SHAKE-128 (A6094)	KAT	CAST	On demand	Manually
SHAKE-128 (A6099)	KAT	CAST	On demand	Manually
SHAKE-128 (A6105)	KAT	CAST	On demand	Manually
SHAKE-128 (A6111)	KAT	CAST	On demand	Manually
SHAKE-256 (A6092)	KAT	CAST	On demand	Manually
SHAKE-256 (A6093)	KAT	CAST	On demand	Manually
SHAKE-256 (A6094)	KAT	CAST	On demand	Manually
SHAKE-256 (A6099)	KAT	CAST	On demand	Manually
SHAKE-256 (A6105)	KAT	CAST	On demand	Manually
SHAKE-256 (A6111)	KAT	CAST	On demand	Manually
AES-ECB (A5553) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5554) - Encrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-ECB (A5555) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5556) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5557) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5558) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5659) - Encrypt	KAT	CAST	On demand	Manually
AES-ECB (A5553) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5554) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5555) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5556) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5557) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5558) - Decrypt	KAT	CAST	On demand	Manually
AES-ECB (A5659) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5553) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5554) - Encrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CCM (A5555) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5556) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5557) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5558) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5659) - Encrypt	KAT	CAST	On demand	Manually
AES-CCM (A5553) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5554) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5555) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5556) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5557) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5558) - Decrypt	KAT	CAST	On demand	Manually
AES-CCM (A5659) - Decrypt	KAT	CAST	On demand	Manually
AES-CMAC (A5553)	KAT	CAST	On demand	Manually
AES-CMAC (A5554)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-CMAC (A5555)	KAT	CAST	On demand	Manually
AES-CMAC (A5556)	KAT	CAST	On demand	Manually
AES-CMAC (A5557)	KAT	CAST	On demand	Manually
AES-CMAC (A5558)	KAT	CAST	On demand	Manually
AES-CMAC (A5659)	KAT	CAST	On demand	Manually
AES-GCM (A6075) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6076) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6077) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6078) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6079) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6080) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6081) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6082) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6083) - Encrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A6100) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6101) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6102) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6106) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6107) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6108) - Encrypt	KAT	CAST	On demand	Manually
AES-GCM (A6075) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6076) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6077) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6078) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6079) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6080) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6081) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6082) - Decrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A6083) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6100) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6101) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6102) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6106) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6107) - Decrypt	KAT	CAST	On demand	Manually
AES-GCM (A6108) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5553) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5554) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5555) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5556) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5557) - Encrypt	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-XTS Testing Revision 2.0 (A5558) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5659) - Encrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5553) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5554) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5555) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5556) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5557) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5558) - Decrypt	KAT	CAST	On demand	Manually
AES-XTS Testing Revision 2.0 (A5659) - Decrypt	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6084)	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6086)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-224 (A6088)	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6090)	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6097)	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6103)	KAT	CAST	On demand	Manually
HMAC-SHA2-224 (A6109)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6084)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6086)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6088)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6090)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6097)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6103)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6109)	KAT	CAST	On demand	Manually
HMAC-SHA2-256 (A6110)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6084)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-384 (A6086)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6088)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6090)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6097)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6103)	KAT	CAST	On demand	Manually
HMAC-SHA2-384 (A6109)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6084)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6086)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6088)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6090)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6097)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6103)	KAT	CAST	On demand	Manually
HMAC-SHA2-512 (A6109)	KAT	CAST	On demand	Manually
HMAC-SHA3-224 (A6092)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA3-224 (A6093)	KAT	CAST	On demand	Manually
HMAC-SHA3-224 (A6094)	KAT	CAST	On demand	Manually
HMAC-SHA3-224 (A6099)	KAT	CAST	On demand	Manually
HMAC-SHA3-224 (A6105)	KAT	CAST	On demand	Manually
HMAC-SHA3-224 (A6111)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6092)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6093)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6094)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6099)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6105)	KAT	CAST	On demand	Manually
HMAC-SHA3-256 (A6111)	KAT	CAST	On demand	Manually
HMAC-SHA3-384 (A6092)	KAT	CAST	On demand	Manually
HMAC-SHA3-384 (A6093)	KAT	CAST	On demand	Manually
HMAC-SHA3-384 (A6094)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA3-384 (A6099)	KAT	CAST	On demand	Manually
HMAC-SHA3-384 (A6105)	KAT	CAST	On demand	Manually
HMAC-SHA3-384 (A6111)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6092)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6093)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6094)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6099)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6105)	KAT	CAST	On demand	Manually
HMAC-SHA3-512 (A6111)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6084)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6086)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6088)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6090)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigGen (FIPS186-5) (A6092)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6093)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6094)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6097)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6099)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6103)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6105)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6109)	KAT	CAST	On demand	Manually
RSA SigGen (FIPS186-5) (A6111)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6084)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-5) (A6086)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6088)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6090)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6092)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6093)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6094)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6097)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6099)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6103)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6105)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigVer (FIPS186-5) (A6109)	KAT	CAST	On demand	Manually
RSA SigVer (FIPS186-5) (A6111)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6084)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6086)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6088)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6090)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6092)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6093)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6094)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6097)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA SigGen (FIPS186-5) (A6099)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6103)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6105)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6109)	KAT	CAST	On demand	Manually
ECDSA SigGen (FIPS186-5) (A6111)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6084)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6086)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6088)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6090)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6092)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA SigVer (FIPS186-5) (A6093)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6094)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6097)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6099)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6103)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6105)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6109)	KAT	CAST	On demand	Manually
ECDSA SigVer (FIPS186-5) (A6111)	KAT	CAST	On demand	Manually
KAS-FFC-SSC Sp800-56Ar3 (A6096)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6084)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 (A6086)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6088)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6090)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6097)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6103)	KAT	CAST	On demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A6109)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6084)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6086)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6088)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6090)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6097)	KAT	CAST	On demand	Manually
TLS v1.2 KDF RFC7627 (A6103)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
TLS v1.2 KDF RFC7627 (A6109)	KAT	CAST	On demand	Manually
TLS v1.3 KDF (A6095)	KAT	CAST	On demand	Manually
KDA HKDF SP800-56Cr2 (A6095)	KAT	CAST	On demand	Manually
KDF SSH (A6085)	KAT	CAST	On demand	Manually
KDF SSH (A6087)	KAT	CAST	On demand	Manually
KDF SSH (A6089)	KAT	CAST	On demand	Manually
KDF SSH (A6091)	KAT	CAST	On demand	Manually
KDF SSH (A6098)	KAT	CAST	On demand	Manually
KDF SSH (A6104)	KAT	CAST	On demand	Manually
PBKDF (A6084)	KAT	CAST	On demand	Manually
PBKDF (A6086)	KAT	CAST	On demand	Manually
PBKDF (A6088)	KAT	CAST	On demand	Manually
PBKDF (A6090)	KAT	CAST	On demand	Manually
PBKDF (A6092)	KAT	CAST	On demand	Manually
PBKDF (A6093)	KAT	CAST	On demand	Manually
PBKDF (A6094)	KAT	CAST	On demand	Manually
PBKDF (A6097)	KAT	CAST	On demand	Manually
PBKDF (A6099)	KAT	CAST	On demand	Manually
PBKDF (A6103)	KAT	CAST	On demand	Manually
PBKDF (A6105)	KAT	CAST	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
PBKDF (A6109)	KAT	CAST	On demand	Manually
PBKDF (A6111)	KAT	CAST	On demand	Manually
Counter DRBG (A5553)	KAT	CAST	On demand	Manually
Counter DRBG (A5554)	KAT	CAST	On demand	Manually
Counter DRBG (A5555)	KAT	CAST	On demand	Manually
Counter DRBG (A5556)	KAT	CAST	On demand	Manually
Counter DRBG (A5557)	KAT	CAST	On demand	Manually
Counter DRBG (A5558)	KAT	CAST	On demand	Manually
Counter DRBG (A5659)	KAT	CAST	On demand	Manually
Safe Primes Key Generation (A6096)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6084)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6086)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6088)	PCT	PCT	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA KeyGen (FIPS186-5) (A6090)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6097)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6103)	PCT	PCT	On demand	Manually
ECDSA KeyGen (FIPS186-5) (A6109)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6084)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6086)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6088)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6090)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6097)	PCT	PCT	On demand	Manually
RSA KeyGen (FIPS186-5) (A6103)	PCT	PCT	On demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA KeyGen (FIPS186-5) (A6109)	PCT	PCT	On demand	Manually

Table 24: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Abort	If the module fails any of the preoperational self-tests or CASTs, the module enters this error state. In this error state, the module immediately stops functioning and ends the application process	Integrity test failure Any CAST failure	Module reinitialization	Message to stderr: "OpenSSL internal error, assertion failed: FATAL FIPS SELFTEST FAILURE". Module does not load.
PCT error	A PCT fails after generation of a key pair is requested	Any PCT failure	Module reinitialization	FIPS_selftest_failed() returns 1. The module returns an error code and stops functioning. Any cryptographic operation is inhibited.

Table 25: Error States

In the “PCT Error” state, errors are reported through the regular ERR interface of the modules and can be queried by functions such as ERR_get_error(). See the OpenSSL man pages for the function description.

10.5 Operator Initiation of Self-Tests

Both conditional and pre-operational self-tests can be executed on-demand by unloading and subsequently re-initializing the module, or by calling the OSSL_PROVIDER_self_test function. The pair-wise consistency tests can be invoked on demand by requesting the key pair generation service.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Before the libopenssl1_1-1.1.1w-150600.5.15.1 RPM package is installed, the SUSE Linux Enterprise SP6 system must operate in the FIPS validated configuration. This can be achieved by:

- Adding the fips=1 option to the kernel command line during the system installation. During the software selection stage, do not install any third-party software.
- Switching the system into the FIPS validated configuration after the installation. Execute the fips-mode-setup --enable command. Restart the system.

In both cases, the Crypto Officer must verify the system operates in the FIPS validated configuration by executing the fips-mode-setup --check command, which should output “FIPS mode is enabled.”

11.2 Administrator Guidance

After the installation of the libopenssl1_1-1.1.1w-150600.5.15.1 RPM package, the Crypto Officer must execute the “Show module name and version” service by issuing the openssl version command. The output of this command must read:

OpenSSL 1.1.1w-fips 11 Sep 2023 SUSE release 150600.5.15.1

11.3 Non-Administrator Guidance

There is no administrator guidance.

11.4 Design and Rules

Not applicable.

11.5 Maintenance Requirements

Not applicable.

11.6 End of Life

As the module does not persistently store SSPs, secure sanitization of the module consists of unloading the module. This will zeroize all SSPs in volatile memory. Then, if desired, the libopenssl1_1-1.1.1w-150600.5.15.1 RPM package can be uninstalled from the SUSE Linux Enterprise SP6 system.

12 Mitigation of Other Attacks

12.1 Attack List

The module implements blinding against RSA timing attacks.

RSA is vulnerable to timing attacks. In a setup where attackers can measure the time of RSA decryption or signature operations, blinding must be used to protect the RSA operation from that attack.

The module provides the API functions RSA_blinding_on() and RSA_blinding_off() to turn the blinding on and off for RSA. When the blinding is on, the module generates a random value to form a blinding factor in the RSA key before the RSA key is used in the RSA cryptographic operations.

Appendix A. TLS Cipher Suites

The module supports the following cipher suites for the TLS protocol versions 1.2 and 1.3 compliant with section 3.3.1 of SP 800-52 Rev. 2. Each cipher suite defines the key exchange algorithm, the bulk encryption algorithm (including the symmetric key size) and the MAC algorithm.

Cipher Suite	ID	Reference
TLS_DH_RSA_WITH_AES_128_CBC_SHA	{ 0x00, 0x31 }	RFC3268
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	{ 0x00, 0x33 }	RFC3268
TLS_DH_RSA_WITH_AES_256_CBC_SHA	{ 0x00, 0x37 }	RFC3268
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	{ 0x00, 0x39 }	RFC3268
TLS_DH_RSA_WITH_AES_128_CBC_SHA256	{ 0x00,0x3F }	RFC5246
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	{ 0x00,0x67 }	RFC5246
TLS_DH_RSA_WITH_AES_256_CBC_SHA256	{ 0x00,0x69 }	RFC5246
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	{ 0x00,0x6B }	RFC5246
TLS_PSK_WITH_AES_128_CBC_SHA	{ 0x00, 0x8C }	RFC4279
TLS_PSK_WITH_AES_256_CBC_SHA	{ 0x00, 0x8D }	RFC4279
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	{ 0x00, 0x9E }	RFC5288
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	{ 0x00, 0x9F }	RFC5288
TLS_DH_RSA_WITH_AES_128_GCM_SHA256	{ 0x00, 0xA0 }	RFC5288
TLS_DH_RSA_WITH_AES_256_GCM_SHA384	{ 0x00, 0xA1 }	RFC5288
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	{ 0xC0, 0x04 }	RFC4492
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	{ 0xC0, 0x05 }	RFC4492
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	{ 0xC0, 0x09 }	RFC4492
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	{ 0xC0, 0x0A }	RFC4492
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA	{ 0xC0, 0x0E }	RFC4492
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	{ 0xC0, 0x0F }	RFC4492

Cipher Suite	ID	Reference
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	{ 0xC0, 0x13 }	RFC4492
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	{ 0xC0, 0x14 }	RFC4492
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	{ 0xC0, 0x23 }	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	{ 0xC0, 0x24 }	RFC5289
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	{ 0xC0, 0x25 }	RFC5289
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	{ 0xC0, 0x26 }	RFC5289
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	{ 0xC0, 0x27 }	RFC5289
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	{ 0xC0, 0x28 }	RFC5289
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	{ 0xC0, 0x29 }	RFC5289
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384	{ 0xC0, 0x2A }	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	{ 0xC0, 0x2B }	RFC5289
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	{ 0xC0, 0x2C }	RFC5289
TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	{ 0xC0, 0x2D }	RFC5289
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384	{ 0xC0, 0x2E }	RFC5289
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	{ 0xC0, 0x2F }	RFC5289
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	{ 0xC0, 0x30 }	RFC5289
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256	{ 0xC0, 0x31 }	RFC5289
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384	{ 0xC0, 0x32 }	RFC5289
TLS_DHE_RSA_WITH_AES_128_CCM	{ 0xC0, 0x9E }	RFC6655
TLS_DHE_RSA_WITH_AES_256_CCM	{ 0xC0, 0x9F }	RFC6655
TLS_DHE_RSA_WITH_AES_128_CCM_8	{ 0xC0, 0xA2 }	RFC6655
TLS_DHE_RSA_WITH_AES_256_CCM_8	{ 0xC0, 0xA3 }	RFC6655
TLS_AES_128_GCM_SHA256	{ 0x13, 0x01 }	RFC8446
TLS_AES_256_GCM_SHA384	{ 0x13, 0x02 }	RFC8446
TLS_AES_128_CCM_SHA256	{ 0x13, 0x04 }	RFC8446

Cipher Suite	ID	Reference
TLS_AES_128_CCM_8_SHA256	{ 0x13, 0x05 }	RFC8446

Appendix B. Glossary and Abbreviations

AES	Advanced Encryption Standard
API	Application Programming Interface
CAST	Cryptographic Algorithm Self-Test
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CKG	Cryptographic Key Generation
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter
CVL	Component Validation List
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EVP	Envelope
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standards

GCM	Galois Counter Mode
GMAC	Galois Counter Mode Message Authentication Code
HKDF	HMAC-based Key Derivation Function
HMAC	Keyed-Hash Message Authentication Code
IKE	Internet Key Exchange
KAS	Key Agreement Scheme
KAT	Known Answer Test
KDA	Key Derivation Algorithm
KDF	Key Derivation Function
KTS	Key Transport Scheme
KW	Key Wrap
KWP	Key Wrap with Padding
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PAI	Processor Algorithm Implementation
PCT	Pair-wise Consistency Test
PBKDF2	Password-based Key Derivation Function v2
PKCS	Public Key Cryptography Standard
PRF	Pseudo-Random Function
PSP	Public Security Parameter
PSS	Probabilistic Signature Scheme

RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SSC	Shared Secret Computation
SSH	Secure Shell
SSP	Sensitive Security Parameter
TLS	Transport Layer Security
XOF	Extendable Output Function
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

Appendix C. References

FIPS 140-3	**FIPS PUB 140-3 - Security Requirements for Cryptographic Modules** March 2019 https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.140-3.pdf
FIPS 140-3 IG	**Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program** 2 September 2025 https://csrc.nist.gov/csrc/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/FIPS%20140-3%20IG.pdf
FIPS 140-3 Management Manual	**FIPS 140-3 Cryptographic Module Validation Program Management Manual** 17 December 2024 https://csrc.nist.gov/csrc/media/Projects/cryptographic-module-validation-program/documents/fips%20140-3/FIPS-140-3-CMVP%20Management%20Manual.pdf
FIPS 180-4	**Secure Hash Standard (SHS)** August 2015 https://doi.org/10.6028/NIST.FIPS.180-4
FIPS 186-5	**Digital Signature Standard (DSS)** February 2023 https://doi.org/10.6028/NIST.FIPS.186-5
FIPS 197	**Advanced Encryption Standard** May 2023 https://doi.org/10.6028/NIST.FIPS.197-upd1
FIPS 198-1	**The Keyed Hash Message Authentication Code (HMAC)** July 2008 https://doi.org/10.6028/NIST.FIPS.198-1

FIPS 202	**SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions** August 2015 https://doi.org/10.6028/NIST.FIPS.202
RFC 3268	**Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)** June 2002 https://www.rfc-editor.org/rfc/rfc3268.txt
RFC 4279	**Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)** December 2005 https://www.rfc-editor.org/rfc/rfc4279.txt
RFC 4492	**Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)** May 2006 https://www.rfc-editor.org/rfc/rfc4492.txt
RFC 5246	**The Transport Layer Security (TLS) Protocol Version 1.2** August 2008 https://www.rfc-editor.org/rfc/rfc4492.txt
RFC 5288	**AES Galois Counter Mode (GCM) Cipher Suites for TLS** August 2008 https://www.rfc-editor.org/rfc/rfc5246.txt
RFC 5289	**TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)** August 2008 https://www.ietf.org/rfc/rfc5289.txt

RFC 6655	**AES-CCM Cipher Suites for Transport Layer Security (TLS)** July 2012 https://www.rfc-editor.org/rfc/rfc6655.txt
RFC 7919	**Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for Transport Layer Security (TLS)** August 2016 https://www.ietf.org/rfc/rfc7919.txt
RFC 8446	**The Transport Layer Security (TLS) Protocol Version 1.3** August 2018 https://www.ietf.org/rfc/rfc8446.txt
SP 800-38A	**Recommendation for Block Cipher Modes of Operation Methods and Techniques** December 2001 https://doi.org/10.6028/NIST.SP.800-38A
SP 800-38B	**Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication** May 2005 https://doi.org/10.6028/NIST.SP.800-38B
SP 800-38C	**Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality** July 2007 https://doi.org/10.6028/NIST.SP.800-38C
SP 800-38D	**Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC** November 2007 https://doi.org/10.6028/NIST.SP.800-38A

SP 800-38E	**Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices** January 2010 https://doi.org/10.6028/NIST.SP.800-38E
SP 800-38F	**Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping** December 2012 https://doi.org/10.6028/NIST.SP.800-38F
SP 800-52 Rev. 2	**Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations** August 2019 https://doi.org/10.6028/NIST.SP.800-52r2
SP 800-56A Rev. 3	**Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography** April 2018 https://doi.org/10.6028/NIST.SP.800-56Ar3
SP 800-56C Rev. 2	**Recommendation for Key-Derivation Methods in Key-Establishment Schemes** August 2020 https://doi.org/10.6028/NIST.SP.800-56Cr2
SP 800-90A Rev. 1	**Recommendation for Random Number Generation Using Deterministic Random Bit Generators** June 2015 https://doi.org/10.6028/NIST.SP.800-90Ar1
SP 800-90B	**Recommendation for the Entropy Sources Used for Random Bit Generation** January 2018 https://doi.org/10.6028/NIST.SP.800-90B

SP 800-131A Rev. 2	**Transitioning the Use of Cryptographic Algorithms and Key Lengths** March 2019 https://doi.org/10.6028/NIST.SP.800-131Ar2
SP 800-132	**Recommendation for Password-Based Key Derivation - Part 1: Storage Applications** December 2010 https://doi.org/10.6028/NIST.SP.800-132
SP 800-133 Rev. 2	**Recommendation for Cryptographic Key Generation** June 2020 https://doi.org/10.6028/NIST.SP.800-133r2
SP 800-135 Rev. 1	**Recommendation for Existing Application-Specific Key Derivation Functions** December 2011 https://doi.org/10.6028/NIST.SP.800-135r1