



Chainguard, Inc.

Chainguard FIPS Provider for OpenSSL

FIPS 140-3 Non-Proprietary Security Policy

Document Version: 1.2

Last Updated: 2026-08-21

Prepared by:

atsec information security corporation

4516 Seton Center Pkwy, Suite 250

Austin, TX 78759

www.atsec.com

Table of Contents

1 General	6
1.1 Overview	6
1.2 Security Levels	6
1.3 Additional Information	6
2 Cryptographic Module Specification	7
2.1 Description	7
2.2 Tested and Vendor Affirmed Module Version and Identification.....	8
2.3 Excluded Components	11
2.4 Modes of Operation	11
2.5 Algorithms.....	11
2.6 Security Function Implementations	27
2.7 Algorithm Specific Information.....	59
2.7.1 AES XTS	59
2.7.2 Key Derivation using SP 800-132 PBKDF2	59
2.7.3 Compliance to SP 800-56Arev3 Assurances	59
2.7.4 SHA-3 and SHAKE.....	60
2.7.5 Legacy Algorithms.....	60
2.7.6 RSA Signatures	60
2.7.7 RSA Key Generation	60
2.7.8 Key Wrapping with AES.....	60
2.7.9 AES GCM IV	60
2.7.10 Compliance to SP 800-56Br2 Assurances.....	61
2.7.11 Compliance to SP 800-227	61
2.7.12 Compliance to FIPS 203	62
2.7.13 Compliance to FIPS 204.....	62
2.8 RBG and Entropy	62

2.9	Key Generation	63
2.10	Key Establishment	63
2.11	Industry Protocols.....	63
3	Cryptographic Module Interfaces	65
3.1	Ports and Interfaces	65
4	Roles, Services, and Authentication	66
4.1	Authentication Methods	66
4.2	Roles	66
4.3	Approved Services	66
4.4	Non-Approved Services.....	90
5	Software/Firmware Security	92
5.1	Integrity Techniques	92
5.2	Initiate on Demand.....	92
6	Operational Environment	93
6.1	Operational Environment Type and Requirements	93
6.2	Configuration Settings and Restrictions.....	93
7	Physical Security.....	94
8	Non-Invasive Security	95
9	Sensitive Security Parameters Management	96
9.1	Storage Areas.....	96
9.2	SSP Input-Output Methods.....	96
9.3	SSP Zeroization Methods	96
9.4	SSPs.....	97
10	Self-Tests	119
10.1	Pre-Operational Self-Tests	119
10.2	Conditional Self-Tests.....	120
10.3	Periodic Self-Test Information.....	126
10.4	Error States.....	127
10.5	Operator Initiation of Self-Tests	128
11	Life-Cycle Assurance	129
11.1	Installation, Initialization, and Startup Procedures	129
11.2	Administrator Guidance.....	129
11.3	Non-Administrator Guidance	129

11.4 End of Life	129
12 Mitigation of Other Attacks	130
12.1 Attack List	130
12.2 Mitigation Effectiveness	130
Appendix A. Glossary and Abbreviations	131
Appendix B. References.....	133

List of Tables

Table 1: Security Levels.....	6
Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)	8
Table 3: Tested Operational Environments - Software, Firmware, Hybrid	9
Table 4: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid	11
Table 5: Modes List and Description	11
Table 6: Approved Algorithms.....	26
Table 7: Vendor-Affirmed Algorithms.....	26
Table 8: Non-Approved, Allowed Algorithms with No Security Claimed	26
Table 9: Non-Approved, Not Allowed Algorithms.....	27
Table 10: Security Function Implementations	59
Table 11: Entropy Certificates	62
Table 12: Entropy Sources.....	62
Table 13: Ports and Interfaces.....	65
Table 14: Roles.....	66
Table 15: Approved Services	89
Table 16: Non-Approved Services	91
Table 17: Storage Areas	96
Table 18: SSP Input-Output Methods	96
Table 19: SSP Zeroization Methods	97
Table 20: SSP Table 1	111
Table 21: SSP Table 2	118
Table 22: Pre-Operational Self-Tests.....	119
Table 23: Conditional Self-Tests	126
Table 24: Pre-Operational Periodic Information	126
Table 25: Conditional Periodic Information	127
Table 26: Error States	128

List of Figures

Figure 1: Block Diagram.....	8
------------------------------	---

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for the Chainguard FIPS Provider for OpenSSL. 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.

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 Chainguard FIPS Provider for OpenSSL (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 consists of one software component, the “FIPS provider” i.e., `fips.so`, which implements the FIPS requirements, and the cryptographic functionality provided to the operator.

Module Type: Software

Module Embodiment: Multi-Chip Standalone

Module Characteristics

Cryptographic Boundary:

The cryptographic boundary is represented by the component in the orange block, that is, the shared library implementing the FIPS provider (`fips.so`).

Tested Operational Environment’s Physical Perimeter (TOEPP):

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 (a general-purpose computer on which the module is installed) is indicated by a purple dashed line.

Components in white 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.

The connecting lines indicate the flow of data between the cryptographic module and its operator application, through the logical interfaces defined in Section 3.

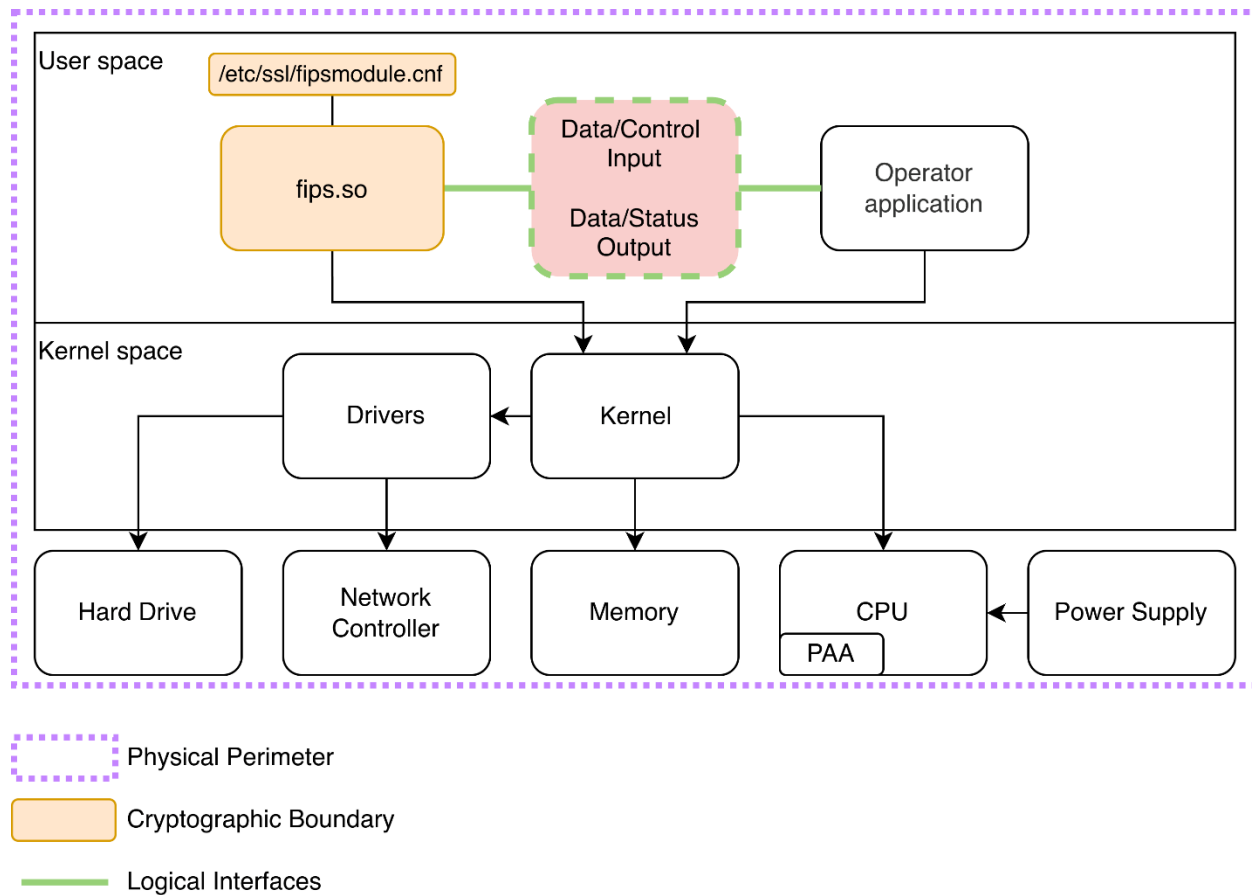


Figure 1: Block Diagram

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
fips.so on Chainguard Image 20230214 on Amazon Linux 2023 on EC2 m7i.metal-24xl on Intel Sapphire Rapids Xeon Platinum 8488C	3.6.0-r4	N/A	HMAC-SHA-256
fips.so on Chainguard Image 20230214 on Amazon Linux 2023 on EC2 m7g.metal on Amazon Graviton3 AWS Graviton3	3.6.0-r4	N/A	HMAC-SHA-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)
Chainguard Image 20230214 on Amazon Linux 2023	EC2 m7i.metal-24xl	Intel Xeon Platinum 8488C	Yes	N/A	3.6.0-r4
Chainguard Image 20230214 on Amazon Linux 2023	EC2 m7i.metal-24xl	Intel Xeon Platinum 8488C	No	N/A	3.6.0-r4
Chainguard Image 20230214 on Amazon Linux 2023	EC2 m7g.metal	AWS Graviton3	Yes	N/A	3.6.0-r4
Chainguard Image 20230214 on Amazon Linux 2023	EC2 m7g.metal	AWS Graviton3	No	N/A	3.6.0-r4

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

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

Operating System	Hardware Platform
glibc 2.34+ (Host)	Generic Hardware Platform ELF x86-64-v2+
glibc 2.34+ (Host)	Generic Hardware Platform ELF ARMv8a+
AlmaLinux 9	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
AlmaLinux 9	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
AlmaLinux 10	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
AlmaLinux 10	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Amazon Linux 2	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Amazon Linux 2	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Amazon Linux 2023	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Amazon Linux 2023	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Azure Linux 2	Azure Esv5-series with Intel Xeon Platinum 8473C, PAA/PAI, under Azure Host Hypervisor
Azure Linux 2	Azure Dpsv6-series with Azure Cobalt 100, PAA/PAI, under Azure Host Hypervisor
Azure Linux 3	Azure Esv5-series with Intel Xeon Platinum 8473C, PAA/PAI, under Azure Host Hypervisor
Azure Linux 3	Azure Dpsv6-series with Azure Cobalt 100, PAA/PAI, under Azure Host Hypervisor
Bottlerocket	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488CC, PAA/PAI
Bottlerocket	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Centos Stream 9	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Centos Stream 9	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Centos Stream 10	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Centos Stream 10	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Chainguard	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI

Operating System	Hardware Platform
Chainguard	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Chainguard (Host)	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Chainguard (Host)	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Chainguard (Host)	Azure Esv5-series with Intel Xeon Platinum 8473C, PAA/PAI, under Azure Host Hypervisor
Chainguard (Host)	Azure Dpsv6-series with Azure Cobalt 100, PAA/PAI, under Azure Host Hypervisor
Chainguard (Host)	GCP c3-highcpu-192-metal with Intel Xeon Platinum 8481C, PAA/PAI, under Titanium
Chainguard (Host)	GCP c4a with Google Axion, PAA/PAI
Chainguard (Host)	Raspberry Pi 5 B Rev 1.0 8GB with Broadcom BCM2712, PAA/PAI
Debian 12 Bookworm	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Debian 12 Bookworm	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Debian 13 Trixie	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Debian 13 Trixie	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Google COS	GCP c3-highcpu-192-metal with Intel Xeon Platinum 8481C
Google COS	GCP c4a with Google Axion, PAA/PAI, under Titanium
RHEL 9	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
RHEL 9	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
RHEL 10	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
RHEL 10	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
RockyLinux 9	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
RockyLinux 9	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
RockyLinux 10	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
RockyLinux 10	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
SUSE Linux Enterprise Server 15 SP6	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
SUSE Linux Enterprise Server 15 SP6	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
SUSE Linux Enterprise Server 16	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
SUSE Linux Enterprise Server 16	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Ubuntu 20.04	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Ubuntu 20.04	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Ubuntu 22.04	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Ubuntu 22.04	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Ubuntu 24.04	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI

Operating System	Hardware Platform
Ubuntu 24.04	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI
Ubuntu 26.04	AWS EC2 m7i.metal-24x with Intel Xeon Platinum 8488C, PAA/PAI
Ubuntu 26.04	AWS EC2 m7g.metal with AWS Graviton3, PAA/PAI

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

The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated SSPs when the module is ported if the specific operational environments are not listed on the validation certificate.

2.3 Excluded Components

There are no components excluded from the module.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Types of Services	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-test and cryptographic algorithm self-tests executed on start-up, the module automatically transitions to the approved mode of 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:

The table below lists all implemented modes or methods of operation for the approved cryptographic algorithms of the module that are employed for approved services (Approved Services table).

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CBC-CS1	A7579, A7587, A7588, A7594, A7595, A7613	Direction - decrypt, encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 128-65536 Increment 8	SP 800-38A
AES-CBC-CS2	A7579, A7587, A7588, A7594, A7595, A7613	Direction - decrypt, encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 128-65536 Increment 8	SP 800-38A
AES-CBC-CS3	A7579, A7587, A7588, A7594, A7595, A7613	Direction - decrypt, encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 128-65536 Increment 8	SP 800-38A
AES-CCM	A7579, A7587, A7588, A7594, A7595, A7613	Key Length - 128, 192, 256 Tag Length - 112, 128, 32, 48, 64, 80, 96 IV Length - IV Length: 56, 64, 72, 80, 88, 96, 104 Payload Length - Payload Length: 0-256 Increment 8 AAD Length - AAD Length: 0-524288 Increment 8	SP 800-38C
AES-CFB1	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB128	A7579, A7587, A7588, A7594, A7595, A7613, A8282	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Generation Key Length - 128, 192, 256 MAC Length - MAC Length: 128 Message Length - Message Length: 8-524288 Increment 8	SP 800-38B
AES-CTR	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - No Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
AES-ECB	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A7580, A7581, A7582, A7589, A7590, A7596,	Direction - Decrypt, Encrypt IV Generation - External, Internal IV Generation Mode - 8.2.1, 8.2.2	SP 800-38D

Algorithm	CAVP Cert	Properties	Reference
	A7597, A7598, A7599, A7600, A7601, A7602, A7603	Key Length - 128, 192, 256 Tag Length - 104, 112, 120, 128, 32, 64, 96 IV Length - IV Length: 96 Payload Length - Payload Length: 128-65536 Increment 128 AAD Length - AAD Length: 0-65536 Increment 8	
AES-GMAC	A7580, A7581, A7582, A7589, A7590, A7596, A7597, A7598, A7599, A7600, A7601, A7602, A7603	Direction - Decrypt, Encrypt IV Generation - External Key Length - 128, 192, 256 Tag Length - 104, 112, 120, 128, 32, 64, 96 IV Length - IV Length: 96 AAD Length - AAD Length: 0-65536 Increment 8	SP 800-38D
AES-KW	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Cipher - Cipher, Inverse Key Length - 128, 192, 256 Payload Length - Payload Length: 128-4096 Increment 128	SP 800-38F
AES-KWP	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Cipher - Cipher, Inverse Key Length - 128, 192, 256 Payload Length - Payload Length: 8-4096 Increment 8	SP 800-38F
AES-OFB	A7579, A7587, A7588, A7594, A7595, A7613	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A7579, A7587, A7588, A7594, A7595, A7596, A7613	Direction - Decrypt, Encrypt Key Length - 128, 256 Payload Length - Payload Length: 128-65536 Increment 8 Tweak Mode - Hex Data Unit Length Matches Payload Length - Yes	SP 800-38E
Counter DRBG	A7579, A7587, A7588, A7594, A7595, A7613	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - No, Yes Additional Input - Additional Input: 0-512 Increment 128, Additional Input: 256, Additional Input: 320, Additional Input: 384 Entropy Input - Entropy Input: 128, Entropy Input: 192, Entropy Input: 256, Entropy Input: 320, Entropy Input: 384 Nonce - Nonce: 0, Nonce: 128, Nonce: 64, Nonce: 96 Personalization String Length - Personalization String Length: 0-512 Increment 128, Personalization String Length: 256, Personalization String Length: 320, Personalization String Length: 384 Returned Bits - 256	SP 800-90A Rev. 1

Algorithm	CAVP Cert	Properties	Reference
Deterministic ECDSA SigGen (FIPS186-5)	A7583, A7591, A7604, A7606, A7608, A7614	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
Deterministic ECDSA SigGen (FIPS186-5)	A7584	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Component - No	FIPS 186-5
Deterministic ECDSA SigGen (FIPS186-5)	A7585, A7593, A7610, A7611, A7612	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA3-256, SHA3-384, SHA3-512 Component - No	FIPS 186-5
ECDSA KeyGen (FIPS186-5)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-4)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Curve - P-192	FIPS 186-4
ECDSA KeyVer (FIPS186-5)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A7583, A7591, A7604, A7606, A7608, A7614	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No, Yes	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A7584	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Component - No, Yes	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A7585, A7593, A7610, A7611, A7612	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA3-256, SHA3-384, SHA3-512 Component - No, Yes	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A7583, A7591, A7604, A7606, A7608, A7614	Component - No Curve - P-192 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A7584	Component - No Curve - P-192 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A7585, A7593, A7610, A7611, A7612	Component - No Curve - P-192 Hash Algorithm - SHA3-256, SHA3-384, SHA3-512	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigVer (FIPS186-5)	A7583, A7591, A7604, A7606, A7608, A7614	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A7584	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A7585, A7593, A7610, A7611, A7612	Component - No Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA3-256, SHA3-384, SHA3-512	FIPS 186-5
EDDSA KeyGen	A7583, A7591, A7604, A7606, A7608, A7614	Curve - ED-25519	FIPS 186-5
EDDSA KeyGen	A7584	Curve - ED-25519, ED-448	FIPS 186-5
EDDSA KeyGen	A7585, A7593, A7610, A7611, A7612	Curve - ED-448	FIPS 186-5
EDDSA KeyVer	A7583, A7591, A7604, A7606, A7608, A7614	Curve - ED-25519	FIPS 186-5
EDDSA KeyVer	A7584	Curve - ED-25519, ED-448	FIPS 186-5
EDDSA KeyVer	A7585, A7593, A7610, A7611, A7612	Curve - ED-448	FIPS 186-5
EDDSA SigGen	A7583, A7591, A7604, A7606, A7608, A7614	Curve - ED-25519 Context Length - Context Length: 0-255 Increment 1 PreHash - Yes Pure - Yes	FIPS 186-5
EDDSA SigGen	A7584	Curve - ED-25519, ED-448 Context Length - Context Length: 0-255 Increment 1 PreHash - Yes Pure - Yes	FIPS 186-5
EDDSA SigGen	A7585, A7593, A7610, A7611, A7612	Curve - ED-448 Context Length - Context Length: 0-255 Increment 1 PreHash - Yes Pure - Yes	FIPS 186-5
EDDSA SigVer	A7583, A7591, A7604, A7606, A7608, A7614	Curve - ED-25519 PreHash - Yes Pure - Yes	FIPS 186-5
EDDSA SigVer	A7584	Curve - ED-25519, ED-448 PreHash - Yes Pure - Yes	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
EDDSA SigVer	A7585, A7593, A7610, A7611, A7612	Curve - ED-448 PreHash - Yes Pure - Yes	FIPS 186-5
Hash DRBG	A7583, A7591, A7604, A7606, A7608, A7614	Prediction Resistance - No, Yes Mode - SHA2-256, SHA2-512	SP 800-90A Rev. 1
Hash DRBG	A7584	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Entropy Input - Entropy Input: 256 Nonce - Nonce: 128 Personalization String Length - Personalization String Length: 0-512 Increment 128 Additional Input - Additional Input: 0-512 Increment 128 Returned Bits - 256, 384, 512	SP 800-90A Rev. 1
HMAC DRBG	A7583, A7591, A7604, A7606, A7608, A7614	Prediction Resistance - No, Yes Mode - SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC DRBG	A7584	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Entropy Input - Entropy Input: 256 Nonce - Nonce: 128 Personalization String Length - Personalization String Length: 0-512 Increment 128 Additional Input - Additional Input: 0-512 Increment 128 Returned Bits - 256, 384, 512	SP 800-90A Rev. 1
HMAC-SHA2- 256	A7583, A7584, A7591, A7604, A7606, A7608, A7614	MAC - MAC: 256 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2- 384	A7583, A7584, A7591, A7604, A7606, A7608, A7614	MAC - MAC: 384 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2- 512	A7583, A7584, A7591, A7604, A7606, A7608, A7614	MAC - MAC: 512 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2- 512/256	A7583, A7584, A7591, A7604,	MAC - MAC: 256 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
	A7606, A7608, A7614		
HMAC-SHA3-256	A7584, A7585, A7593, A7610, A7611, A7612	MAC - MAC: 256 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A7584, A7585, A7593, A7610, A7611, A7612	MAC - MAC: 384 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-512	A7584, A7585, A7593, A7610, A7611, A7612	MAC - MAC: 512 Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A7586	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
KAS-IFC-SSC	A8281	Modulo - 2048, 3072, 4096, 6144, 8192 Key Generation Methods - rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2-basic, rsakpg2-crt, rsakpg2-prime-factor Scheme - KAS1 - KAS Role - initiator, responder KAS2 - KAS Role - initiator, responder Fixed Public Exponent - 010001	SP 800-56A Rev. 3
KDA HKDF SP800-56Cr2	A7583, A7591, A7604, A7606, A7608, A7614	Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 HMAC Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	SP 800-56C Rev. 2
KDA HKDF SP800-56Cr2	A7584	Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048	SP 800-56C Rev. 2

Algorithm	CAVP Cert	Properties	Reference
		Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 HMAC Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	
KDA HKDF SP800-56Cr2	A7585, A7593, A7610, A7611, A7612	Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 HMAC Algorithm - SHA3-256, SHA3-384, SHA3-512 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	SP 800-56C Rev. 2
KDA OneStep SP800-56Cr2	A7583, A7591, A7604, A7606, A7608, A7614	Auxiliary Function Methods - Auxiliary Function Name - SHA2-256 Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8	SP 800-56C Rev. 2
KDA OneStep SP800-56Cr2	A7584	Auxiliary Function Methods - Auxiliary Function Name - HMAC-SHA2-256 MAC Salting Methods - default, random Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8	SP 800-56C Rev. 2
KDA OneStep SP800-56Cr2	A7585, A7593, A7610, A7611, A7612	Auxiliary Function Methods - Auxiliary Function Name - SHA3-256 Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8	SP 800-56C Rev. 2
KDA TwoStep SP800-56Cr2	A7583, A7591, A7604, A7606, A7608, A7614	MAC Salting Methods - default, random Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation KDF Mode - feedback MAC Modes - HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA2-512/256 Fixed Data Order - after fixed data Counter Lengths - 8	SP 800-56C Rev. 2

Algorithm	CAVP Cert	Properties	Reference
		The KDF supports an empty IV - Yes The KDF requires an empty IV - Yes Supported Lengths - Supported Lengths: 2048 Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	
KDA TwoStep SP800-56Cr2	A7584	MAC Salting Methods - default, random Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation KDF Mode - feedback MAC Modes - HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA2-512/256, HMAC-SHA3- 256, HMAC-SHA3-384, HMAC-SHA3-512 Fixed Data Order - after fixed data Counter Lengths - 8 The KDF supports an empty IV - Yes The KDF requires an empty IV - Yes Supported Lengths - Supported Lengths: 2048 Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	SP 800-56C Rev. 2
KDA TwoStep SP800-56Cr2	A7585, A7593, A7610, A7611, A7612	MAC Salting Methods - default, random Fixed Info Pattern - uPartyInfo vPartyInfo Fixed Info Encoding - concatenation KDF Mode - feedback MAC Modes - HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 Fixed Data Order - after fixed data Counter Lengths - 8 The KDF supports an empty IV - Yes The KDF requires an empty IV - Yes Supported Lengths - Supported Lengths: 2048 Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 Perform Multiple Expansion Tests - No Uses Hybrid Shared Secret - No	SP 800-56C Rev. 2
KDF ANS 9.42 (CVL)	A7583, A7591, A7604, A7606, A7608, A7614	KDF Type - DER Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2- 512/256 zz Length - zz Length: 112-4096 Increment 8	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
		Key Data Length - Key Data Length: 112-4096 Increment 8 Supplemental Information Length - Supplemental Information Length: 0-256 Increment 8 OID - AES-128-KW, AES-192-KW, AES-256-KW	
KDF ANS 9.42 (CVL)	A7584	KDF Type - DER Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 zz Length - zz Length: 112-4096 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8 Supplemental Information Length - Supplemental Information Length: 0-256 Increment 8 OID - AES-128-KW, AES-192-KW, AES-256-KW	SP 800-135 Rev. 1
KDF ANS 9.42 (CVL)	A7585, A7593, A7610, A7611, A7612	KDF Type - DER Hash Algorithm - SHA3-256, SHA3-384, SHA3-512 zz Length - zz Length: 112-4096 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8 Supplemental Information Length - Supplemental Information Length: 0-256 Increment 8 OID - AES-128-KW, AES-192-KW, AES-256-KW	SP 800-135 Rev. 1
KDF ANS 9.63 (CVL)	A7583, A7591, A7604, A7606, A7608, A7614	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256 Field Size - 224, 571 Shared Info Length - Shared Info Length: 0-1024 Increment 8 Key Data Length - Key Data Length: 128-4096 Increment 8	SP 800-135 Rev. 1
KDF ANS 9.63 (CVL)	A7584	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Field Size - 224, 571 Shared Info Length - Shared Info Length: 0-1024 Increment 8 Key Data Length - Key Data Length: 128-4096 Increment 8	SP 800-135 Rev. 1
KDF ANS 9.63 (CVL)	A7585, A7593, A7610, A7611, A7612	Hash Algorithm - SHA3-256, SHA3-384, SHA3-512 Field Size - 224, 571 Shared Info Length - Shared Info Length: 0-1024 Increment 8 Key Data Length - Key Data Length: 128-4096 Increment 8	SP 800-135 Rev. 1
KDF KMAC Sp800-108r1	A7584, A7585, A7593, A7610, A7611, A7612	Key Derivation Key Length - Key Derivation Key Length: 128 Context Length - Context Length: 128 Label Length - Label Length: 128 Derived Key Length - Derived Key Length: 256 MAC Modes - KMAC-128, KMAC-256	SP 800-108 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
KDF SP800-108	A7579, A7587, A7588, A7594, A7595, A7613	KDF Mode - Counter, Feedback MAC Mode - CMAC-AES128, CMAC-AES192, CMAC-AES256 Supported Lengths - Supported Lengths: 112-4096 Increment 8 Fixed Data Order - Before Fixed Data Counter Length - 32 Supports Empty IV - Yes Requires Empty IV - No Custom Key In Length - 0	SP 800-108 Rev. 1
KDF SP800-108	A7583, A7591, A7604, A7606, A7608, A7614	KDF Mode - Counter, Feedback MAC Mode - HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA2-512/256 Supported Lengths - Supported Lengths: 112-4096 Increment 8 Fixed Data Order - Before Fixed Data Counter Length - 32 Supports Empty IV - Yes Requires Empty IV - No Custom Key In Length - 0	SP 800-108 Rev. 1
KDF SP800-108	A7584	KDF Mode - Counter, Feedback MAC Mode - HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA2-512/256, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 Supported Lengths - Supported Lengths: 112-4096 Increment 8 Fixed Data Order - Before Fixed Data Counter Length - 32 Supports Empty IV - Yes Requires Empty IV - No Custom Key In Length - 0	SP 800-108 Rev. 1
KDF SP800-108	A7585, A7593, A7610, A7611, A7612	KDF Mode - Counter, Feedback MAC Mode - HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 Supported Lengths - Supported Lengths: 112-4096 Increment 8 Fixed Data Order - Before Fixed Data Counter Length - 32 Supports Empty IV - Yes Requires Empty IV - No Custom Key In Length - 0	SP 800-108 Rev. 1
KDF SSH (CVL)	A7583, A7584, A7591, A7592, A7604, A7605, A7606, A7607,	Cipher - AES-128, AES-192, AES-256, TDES Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
	A7608, A7609, A7614, A7615		
KMAC-128	A7584, A7585, A7593, A7610, A7611, A7612	Message Length - Message Length: 0-65536 Increment 8 MAC Length - MAC Length: 32-65536 Increment 8 Key Data Length - Key Data Length: 128-1024 Increment 8 Hex Customization - Yes Supports eXtendable-Output Functions - No, Yes	SP 800-185
KMAC-256	A7584, A7585, A7593, A7610, A7611, A7612	Message Length - Message Length: 0-65536 Increment 8 MAC Length - MAC Length: 32-65536 Increment 8 Key Data Length - Key Data Length: 128-1024 Increment 8 Hex Customization - Yes Supports eXtendable-Output Functions - No, Yes	SP 800-185
KTS-IFC	A7583, A7591, A7604, A7606, A7608, A7614	Function - keyPairGen, partialVal IUT ID - 0123456789abcdef Modulo - 2048, 3072, 4096, 6144, 8192 Key Generation Methods - rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2-basic, rsakpg2-crt, rsakpg2-prime-factor Fixed Public Exponent - 010001 Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Transport Method - Hash Algorithms - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Supports Null Associated Data - Yes Associated Data Pattern - uPartyInfo vPartyInfo Associated Data Encoding - concatenation Key Length - 768	SP 800-56B Rev. 2
KTS-IFC	A7584	Function - keyPairGen, partialVal IUT ID - 0123456789abcdef Modulo - 2048, 3072, 4096, 6144, 8192 Key Generation Methods - rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2-basic, rsakpg2-crt, rsakpg2-prime-factor Fixed Public Exponent - 010001 Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Transport Method - Hash Algorithms - SHA2-256 Supports Null Associated Data - Yes Associated Data Pattern - uPartyInfo vPartyInfo Associated Data Encoding - concatenation Key Length - 768	SP 800-56B Rev. 2

Algorithm	CAVP Cert	Properties	Reference
ML-DSA KeyGen	A7584, A7585, A7593, A7610, A7611, A7612	Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87	FIPS 204
ML-DSA SigGen	A7584, A7585, A7593, A7610, A7611, A7612	Signature Interfaces - external, internal Pre Hash - pure Deterministic - No, Yes External Mu - No, Yes Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87 Message Length - Message Length: 8-65536 Increment 8 Context Length - Context Length: 0-248 Increment 8	FIPS 204
ML-DSA SigVer	A7584, A7585, A7593, A7610, A7611, A7612	Signature Interfaces - external, internal Pre Hash - pure External Mu - No, Yes Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87 Message Length - Message Length: 8-65536 Increment 8 Context Length - Context Length: 0-248 Increment 8	FIPS 204
ML-KEM EncapDecap	A7584, A7585, A7593, A7610, A7611, A7612	Parameter Sets - ML-KEM-1024, ML-KEM-512, ML-KEM-768 Functions - Decapsulation, Decapsulation Key Check, Encapsulation, Encapsulation Key Check	FIPS 203
ML-KEM KeyGen	A7584, A7585, A7593, A7610, A7611, A7612	Parameter Sets - ML-KEM-1024, ML-KEM-512, ML-KEM-768	FIPS 203
PBKDF	A7583, A7591, A7604, A7606, A7608, A7614	Iteration Count - Iteration Count: 1000-10000 Increment 1 HMAC Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256 Password Length - Password Length: 8-128 Increment 1 Salt Length - Salt Length: 128-4096 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8	SP 800-132
PBKDF	A7584	Iteration Count - Iteration Count: 1000-10000 Increment 1 HMAC Algorithm - SHA2-256, SHA2-384, SHA2-512, SHA2-512/256, SHA3-256, SHA3-384, SHA3-512 Password Length - Password Length: 8-128 Increment 1 Salt Length - Salt Length: 128-4096 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8	SP 800-132
PBKDF	A7585, A7593, A7610, A7611, A7612	Iteration Count - Iteration Count: 1000-10000 Increment 1 HMAC Algorithm - SHA3-256, SHA3-384, SHA3-512 Password Length - Password Length: 8-128 Increment 1 Salt Length - Salt Length: 128-4096 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8	SP 800-132
RSA KeyGen (FIPS186-5)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Key Generation Mode - probableWithProbableAux Modulo - 2048, 3072, 4096, 6144, 8192 p mod 8 - 0 Primality Tests - 2powSecStr	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
		q mod 8 - 0 Info Generated By Server - Yes Private Key Format - standard Public Exponent Mode - random	
RSA SigGen (FIPS186-5)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1	FIPS 186-5
RSA SigGen (FIPS186-5)	A7585, A7593, A7610, A7611, A7612	Hash Pair - Hash Algorithm - SHA3-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1	FIPS 186-5
RSA SigVer (FIPS186-4)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA2-256 Public Exponent Mode - Random	FIPS 186-4
RSA SigVer (FIPS186-5)	A7584, A7591, A7604, A7606, A7608, A7614	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1 Public Exponent Mode - random	FIPS 186-5
RSA SigVer (FIPS186-5)	A7585, A7593, A7610, A7611, A7612	Hash Pair - Hash Algorithm - SHA3-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1 Public Exponent Mode - random	FIPS 186-5
Safe Primes Key Generation	A7586	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	A7586	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP- 4096, MODP-6144, MODP-8192	SP 800-56A Rev. 3
SHA2-256	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A7583, A7584, A7591, A7604,	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
	A7606, A7608, A7614		
SHA2-512	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA3-256	A7584, A7585, A7593, A7610, A7611, A7612	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-384	A7584, A7585, A7593, A7610, A7611, A7612	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-512	A7584, A7585, A7593, A7610, A7611, A7612	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHAKE-128	A7584, A7585, A7593, A7610, A7611, A7612	Supports Bit-Oriented Messages - No Supports Empty Message - Yes Supports Bit-Oriented Output - No Output Length - Output Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A7584, A7585, A7593, A7610, A7611, A7612	Supports Bit-Oriented Messages - No Supports Empty Message - Yes Supports Bit-Oriented Output - No Output Length - Output Length: 16-65536 Increment 8	FIPS 202
SLH-DSA KeyGen	A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614	Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA-SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s	FIPS 205
SLH-DSA SigGen	A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614	Deterministic - No, Yes Signature Interfaces - external, internal Pre Hash - pure Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA-SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s	FIPS 205

Algorithm	CAVP Cert	Properties	Reference
		Message Length - Message Length: 8-65536 Increment 8 Context Length - Context Length: 0-248 Increment 8	
SLH-DSA SigVer	A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614	Signature Interfaces - external, internal Pre Hash - pure Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA-SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s Message Length - Message Length: 8-65536 Increment 8 Context Length - Context Length: 0-248 Increment 8	FIPS 205
TLS v1.2 KDF RFC7627 (CVL)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Key Block Length - Key Block Length: 1024	SP 800-135 Rev. 1
TLS v1.3 KDF (CVL)	A7583, A7584, A7591, A7604, A7606, A7608, A7614	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, direct DRBG output without XOR

Table 7: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

Name	Caveat	Use and Function
X25519 shared secret computation	No security claimed, compliant with IG D.S Scenario 2 and IG 2.4.A	Provides 128 bits of security, Key Agreement; used only in the context of hybrid ML-KEM and X25519
X25519 KeyGen	No security claimed, compliant with IG D.S Scenario 2 and IG 2.4.A	Provides 128 bits of security, Key Generation; used only in the context of hybrid ML-KEM and X25519

Table 8: Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
AES GCM encryption with external IV	Encryption
HMAC with key length < 112 bits	Message Authentication Code
KMAC with key length < 112 bits or tag length < 32 bits	Message Authentication Code
PBKDF2 with password length < 8 characters or salt length < 128 bits or output key length < 112 bits or iteration count < 1000	Key Derivation Function
TLS 1.2 KDF without extended master secret or with SHA2-512/256 or with SHA-3 functions or with input secret length < 112 bits	Key Derivation Function
KBKDF with input key length < 112 bits	Key Derivation Function
Hash_DRBG or HMAC_DRBG with SHA2-384 or SHA2-512/256 or SHA-3 functions or KMAC	Deterministic Random Bit Generation
ECDH with P-192	Shared Secret Computation
RSA SigGen with X9.31 padding or with modulus length < 2048 bits or with PSS salt length > digest length or without hashing (primitive)	Digital Signature Generation
RSA SigVer with modulus length < 2048 bits or with PSS salt length > digest length or without hashing (primitive)	Digital Signature Verification
ECDSA SigGen with P-192 or with SHA2-512/256; ECDSA deterministic SigGen with P-192 or with SHA2-512/256; ECDSA SigGen component with P-192	Digital Signature Generation
ECDSA SigVer with SHA2-512/256; ECDSA SigVer component	Digital Signature Verification
ECDSA KeyGen with P-192	Key Pair Generation
ML-KEM encapsulation with user-provided randomness	Key Encapsulation
SSH KDF with SHA2-512/256 or with SHA-3 functions or with input secret length < 112 bits	Key Derivation Function
TLS 1.3 KDF with SHA2-512 or SHA2-512/256 or SHA-3 functions or input secret length < 112 bits	Key Derivation Function
One step KDF, two step KDF, HKDF, ANS X9.42 KDF, ANS X9.63 KDF with input secret length < 112 bits	Key Derivation Function

Table 9: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Encryption with AES	BC-UnAuth	SP 800-38A and SP 800-38E. Encryption		AES-CBC-CS1: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB128: (A7579, A7587,

Name	Type	Description	Properties	Algorithms
				A7588, A7594, A7595, A8282, A7613) AES-CBC-CS2: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB1: (A7579, A7587, A7588, A7594, A7595, A7613) AES-OFB: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CBC-CS3: (A7579, A7587, A7588, A7594, A7595, A7613) AES-XTS Testing Revision 2.0: (A7579, A7587, A7588, A7594, A7595, A7596, A7613) AES-CTR: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CBC: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB8: (A7579, A7587, A7588, A7594, A7595, A7613)
Decryption with AES	BC-UnAuth	SP 800-38A and SP 800-38E. Decryption		AES-CBC-CS1: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB128: (A7579, A7587, A7588, A7594, A7595, A8282,

Name	Type	Description	Properties	Algorithms
				A7613) AES-CBC-CS2: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB1: (A7579, A7587, A7588, A7594, A7595, A7613) AES-OFB: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CBC-CS3: (A7579, A7587, A7588, A7594, A7595, A7613) AES-XTS Testing Revision 2.0: (A7579, A7587, A7588, A7594, A7595, A7596, A7613) AES-CTR: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CBC: (A7579, A7587, A7588, A7594, A7595, A7613) AES-CFB8: (A7579, A7587, A7588, A7594, A7595, A7613)
Authenticated Encryption with AES	BC-Auth	SP 800-38D. Authenticated encryption		AES-CCM: (A7579, A7587, A7588, A7594, A7595, A7613) AES-GCM: (A7580, A7581, A7582, A7589, A7590, A7596, A7597, A7598, A7599, A7600, A7601,

Name	Type	Description	Properties	Algorithms
				A7602, A7603) AES-KW: (A7579, A7587, A7588, A7594, A7595, A7613) AES-KWP: (A7579, A7587, A7588, A7594, A7595, A7613)
Authenticated Decryption with AES	BC-Auth	SP 800-38D. Authenticated decryption		AES-CCM: (A7579, A7587, A7588, A7594, A7595, A7613) AES-GCM: (A7580, A7581, A7582, A7589, A7590, A7596, A7597, A7598, A7599, A7600, A7601, A7602, A7603) AES-KW: (A7579, A7587, A7588, A7594, A7595, A7613) AES-KWP: (A7579, A7588, A7587, A7594, A7595, A7613)
Message Authentication Generation with AES	MAC	SP 800-38B and SP 800-38D Message authentication generation		AES-GMAC: (A7580, A7581, A7582, A7589, A7590, A7596, A7597, A7598, A7599, A7600, A7601, A7602, A7603, A7584) AES-CMAC: (A7579, A7587, A7588, A7594, A7595, A7613)
Message Authentication Generation with HMAC	MAC	FIPS 198-1. Message authentication generation		HMAC-SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)

Name	Type	Description	Properties	Algorithms
				SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591,

Name	Type	Description	Properties	Algorithms
				A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Message Authentication Generation with KMAC	MAC	SP 800-185. Message authentication generation		KMAC-256: (A7585, A7593, A7610, A7611, A7612, A7584) KMAC-128: (A7585, A7593, A7610, A7611, A7612, A7584)
Random Number Generation with DRBG	DRBG	SP 800-90A Rev1. Random number generation		HMAC DRBG: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) Counter DRBG: (A7579, A7587, A7588, A7594, A7595, A7613) Hash DRBG: (A7583, A7584,

Name	Type	Description	Properties	Algorithms
				A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) AES-ECB: (A7579, A7613, A7595, A7594, A7588, A7587)
Signature Generation with ECDSA	DigSig-SigGen	FIPS 186-5. Signature generation		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) ECDSA SigGen (FIPS186-5): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)

Name	Type	Description	Properties	Algorithms
				Deterministic ECDSA SigGen (FIPS186-5): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614)
Signature Generation with RSA	DigSig-SigGen	FIPS 186-5. Signature generation. Per IG C.F, RSA SigGen was CAVP tested with moduli sizes 2048, 3072, 4096 bits. The module supports moduli sizes larger than 4096 bits, up to 16384 bits.		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) RSA SigGen (FIPS186-5): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604,

Name	Type	Description	Properties	Algorithms
				A7606, A7608, A7614, A7584)
Signature Verification with ECDSA	DigSig-SigVer	FIPS 186-5. Signature verification		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) ECDSA SigVer (FIPS186-5): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Signature Verification with ECDSA (Legacy)	DigSig-SigVer	FIPS 186-4. Signature verification		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608,

Name	Type	Description	Properties	Algorithms
				A7614, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) ECDSA SigVer (FIPS186-4): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614)
Signature Verification with RSA	DigSig-SigVer	FIPS 186-5. Signature verification. Per IG C.F, RSA SigVer was CAVP tested with moduli sizes 2048, 3072, 4096 bits. The module supports moduli sizes larger than 4096 bits, up to 16384 bits.		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) RSA SigVer (FIPS186-5): (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611,

Name	Type	Description	Properties	Algorithms
				A7612, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Signature Verification with RSA (Legacy)	DigSig-SigVer	FIPS 186-4. Signature verification		SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) RSA SigVer (FIPS186-4): (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-384: (A7584) SHA3-512: (A7584)
Key Pair Generation with ECDSA	AsymKeyPair-KeyGen CKG	FIPS 186-5. Key pair generation		ECDSA KeyGen (FIPS186-5): (A7583, A7591,

Name	Type	Description	Properties	Algorithms
				A7604, A7606, A7608, A7614, A7584)
Key Pair Generation with RSA	AsymKeyPair-KeyGen CKG	FIPS 186-5. Key pair generation. Per IG C.F, RSA KeyGen was CAVP tested with moduli sizes of 2048, 3072, 4096, 6144, 8192 bits. The module supports moduli sizes larger than 8192 bits, up to 16384 bits. The number of Miller-Rabin tests is compliant with Table B.1 of FIPS 186-5.		RSA KeyGen (FIPS186-5): (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Key Pair Generation with Safe Primes	AsymKeyPair-KeyGen CKG	SP 800-56Ar3. Key pair generation		Safe Primes Key Generation: (A7586)
Public Key Verification with ECDSA	AsymKeyPair-PubKeyVal	FIPS 186-5. Key verification		ECDSA KeyVer (FIPS186-5): (A7583, A7591, A7604, A7606, A7608, A7614, A7584) ECDSA KeyVer (FIPS186-4): (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Key Pair Verification with Safe Primes	AsymKeyPair-KeyVer	SP 800-56Ar3. Key pair verification		Safe Primes Key Verification: (A7586)
Key Derivation with KBKDF	KBKDF	SP 800-108r1. Key derivation		KDF KMAC Sp800-108r1: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) KDF SP800-108: (A7579, A7583, A7584, A7585,

Name	Type	Description	Properties	Algorithms
				A7587, A7588, A7591, A7593, A7594, A7595, A7604, A7606, A7608, A7610, A7611, A7612, A7613, A7614) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KMAC-256: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2- 512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593, A7610, A7611,

Name	Type	Description	Properties	Algorithms
				A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KMAC-128: (A7585, A7593, A7610, A7611, A7612, A7584)
Key Derivation with KDA OneStep	KAS-56CKDF	SP 800-56Cr2. Key derivation		HMAC-SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585,

Name	Type	Description	Properties	Algorithms
				A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) KDA OneStep SP800-56Cr2: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) KMAC-256:

Name	Type	Description	Properties	Algorithms
				(A7585, A7593, A7610, A7611, A7612) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KMAC-128: (A7585, A7593, A7610, A7611, A7612)
Key Derivation with KDA TwoStep	KAS-56CKDF	SP 800-56Cr2. Key derivation		HMAC-SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KDA TwoStep SP800-56Cr2: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) HMAC-SHA2-512: (A7583, A7591,

Name	Type	Description	Properties	Algorithms
				A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584)
Key Derivation with KDA HKDF	KAS-56CKDF	SP 800-56Cr2. Key derivation		KDA HKDF SP800- 56Cr2: (A7583, A7584, A7585, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614, A7591) HMAC-SHA2-256: (A7583, A7584,

Name	Type	Description	Properties	Algorithms
				A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA2- 512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612,

Name	Type	Description	Properties	Algorithms
				A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Key Derivation with ANS X9.42 KDF	KAS-135KDF	SP 800-135r1. Key derivation		KDF ANS 9.42: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584)

Name	Type	Description	Properties	Algorithms
				SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Key Derivation with X9.63 KDF	KAS-135KDF	SP 800-135r1. Key derivation		KDF ANS 9.63: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Key Derivation with SSH KDF	KAS-135KDF	SP 800-135r1. Key derivation		SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)

Name	Type	Description	Properties	Algorithms
				A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KDF SSH: (A7583, A7584, A7591, A7592, A7604, A7605, A7606, A7607, A7608, A7609, A7614, A7615)
Key Derivation with TLS 1.2 KDF	KAS-135KDF	SP 800-135r1. Key derivation		SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) TLS v1.2 KDF RFC7627: (A7583, A7584, A7591, A7604, A7606, A7608, A7614)
Key Derivation with TLS 1.3 KDF	KAS-135KDF	RFC 8446. Key derivation		TLS v1.3 KDF: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)

Name	Type	Description	Properties	Algorithms
Key Derivation with PBKDF2	PBKDF	SP 800-132. Key derivation		PBKDF: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) HMAC-SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) HMAC-SHA3-384: (A7585, A7593,

Name	Type	Description	Properties	Algorithms
				A7610, A7611, A7612, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Shared Secret Computation	KAS-SSC	SP 800-56Ar3. KAS-ECC-SSC and KAS-FFC-SSC		KAS-FFC-SSC Sp800-56Ar3: (A7586) KAS-ECC-SSC Sp800-56Ar3: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) KAS-IFC-SSC: (A8281)
Message Digest with SHA	SHA	FIPS 180-4 and FIPS 202. Message digest		SHA3-384: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-512/256: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA3-512: (A7585, A7593, A7610, A7611, A7612,

Name	Type	Description	Properties	Algorithms
				A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHA2-384: (A7583, A7591, A7604, A7606, A7608, A7614, A7584)
Message Digest with SHAKE	XOF	FIPS 202. Message digest		SHAKE-128: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-256: (A7585, A7593, A7610, A7611, A7612, A7584)
Key Generation with ML-KEM	AsymKeyPair-KeyGen CKG	FIPS 203. Key generation		ML-KEM KeyGen: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)

Name	Type	Description	Properties	Algorithms
Key Encapsulation with ML-KEM	KEM-Encap	FIPS 203. Key encapsulation	IG:IG D.S Scenario 1	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Decapsulation with ML-KEM	KEM-Decap	FIPS 203. Key decapsulation	IG:IG D.S Scenario 1	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Generation with ML-DSA	AsymKeyPair-KeyGen CKG	FIPS 204. Key generation		ML-DSA KeyGen: (A7585, A7593, A7610, A7611,

Name	Type	Description	Properties	Algorithms
				A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612)
Signature Generation with ML-DSA	DigSig-SigGen	FIPS 204. Signature generation	Signature interfaces:The external and internal interfaces are supported Deterministic:Both the deterministic and hedged variants are supported External mu:Both the external and internal mu computations are supported	ML-DSA SigGen: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612)
Signature Verification with ML-DSA	DigSig-SigVer	FIPS 204. Signature verification	Signature interfaces:The external and internal interfaces are supported External mu:Both the external and internal mu computations are supported	ML-DSA SigVer: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Generation with SLH-DSA	AsymKeyPair- KeyGen CKG	FIPS 205. Key generation		SLH-DSA KeyGen: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612)

Name	Type	Description	Properties	Algorithms
				SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614)
Signature Generation with SLH-DSA	DigSig-SigGen	FIPS 205. Signature generation	Signature interfaces: The external and internal interfaces are supported Deterministic: Both the deterministic and hedged variants are supported	SLH-DSA SigGen: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) HMAC-SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) HMAC-SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614)
Signature Verification with SLH-DSA	DigSig-SigVer	FIPS 205. Signature verification	Signature interfaces: The external and internal interfaces are supported	SLH-DSA SigVer: (A7583, A7584, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614) SHAKE-256: (A7584, A7585,

Name	Type	Description	Properties	Algorithms
				A7593, A7610, A7611, A7612) SHA2-256: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614)
Key Encapsulation with KEM-Hybrid-MLX-KEM	KEM-Hybrid KEM-Encap	FIPS 203. Key encapsulation using X25519MLKEM768 per SP 800-227	IG:IG D.S Scenario 2 CAVP Testing Option 1 Key Combiner:N/A: the approved HKDF key combiner is offered as a separate SFI/Service Non-ML-KEM:X25519, non-approved with no security claimed per IG 2.4.A	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612) X25519 shared secret computation: () SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Encapsulation with Hybrid-ML-KEM-ECDH	KEM-Hybrid KEM-Encap	FIPS 203 and SP 800-56Ar3. Key encapsulation using SecP256r1MLKEM768 and SecP384r1MLKEM1024 per SP 800-227	IG:IG D.S Scenario 2 CAVP Testing Option 1 Key Combiner:N/A: the approved HKDF key combiner is offered as a separate SFI/Service Non-ML-KEM:KAS-ECC-SSC, approved per IG D.F Scenario 2	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612) KAS-ECC-SSC Sp800-56Ar3: (A7583, A7584, A7591, A7604, A7606, A7608, A7614)

Name	Type	Description	Properties	Algorithms
				SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Decapsulation with KEM-Hybrid-MLX-KEM	KEM-Hybrid KEM-Decap	FIPS 203. Key decapsulation using X25519MLKEM768 per SP 800-227	IG:IG D.S Scenario 2 CAVP Testing Option 1 Key Combiner:N/A: the approved HKDF key combiner is offered as a separate SFI/Service Non-ML- KEM:X25519, non- approved with no security claimed per IG 2.4.A	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612) X25519 shared secret computation: () SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Decapsulation with Hybrid-ML-KEM-ECDH	KEM-Hybrid KEM-Decap	FIPS 203 and SP 800-56Ar3. Key decapsulation using SecP256r1MLKEM768 and SecP384r1MLKEM1024 per SP 800-227	IG:IG D.S Scenario 2 CAVP Testing Option 1 Key Combiner:N/A: the approved HKDF	ML-KEM EncapDecap: (A7585, A7593, A7610, A7611, A7612)

Name	Type	Description	Properties	Algorithms
			key combiner is offered as a separate SFI/Service Non-ML-KEM:KAS-ECC-SSC, approved per IG D.F Scenario 2	KAS-ECC-SSC Sp800-56Ar3: (A7583, A7584, A7591, A7604, A7606, A7608, A7614) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Generation with KEM-Hybrid-MLX-KEM	AsymKeyPair-KeyGen CKG	FIPS 203. Key generation of X25519MLKEM768 key pairs per SP 800-227		ML-KEM KeyGen: (A7585, A7593, A7610, A7611, A7612) X25519 KeyGen: () SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key Generation with Hybrid-	AsymKeyPair-KeyGen	FIPS 203 and SP 800-56Ar3. Key generation of		ML-KEM KeyGen: (A7585, A7593,

Name	Type	Description	Properties	Algorithms
ML-KEM-ECDH		SecP256r1MLKEM768 and SecP384r1MLKEM1024 key pairs per SP 800-227		A7610, A7611, A7612) ECDSA KeyGen (FIPS186-5): (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHAKE-256: (A7584, A7585, A7593, A7610, A7611, A7612) SHA3-512: (A7585, A7593, A7610, A7611, A7612, A7584) SHA3-256: (A7585, A7593, A7610, A7611, A7612, A7584) SHAKE-128: (A7584, A7585, A7593, A7610, A7611, A7612)
Key verification of ML-KEM encapsulation key	AsymKeyPair-KeyVer AsymKeyPair-PubKeyVal	FIPS 203. Encapsulation key verification		ML-KEM KeyGen: (A7585, A7593, A7610, A7611, A7612)
Signature Generation with EdDSA	DigSig-SigGen	FIPS 186-5; Signature Generation		SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) EDDSA SigGen: (A7583, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614, A7584) SHAKE-256: (A7585, A7593, A7610, A7611, A7612, A7584)

Name	Type	Description	Properties	Algorithms
Signature Verification with EdDSA	DigSig-SigVer	FIPS 186-5; Signature Verification		EDDSA SigVer: (A7583, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614, A7584) SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) SHAKE-256: (A7585, A7593, A7610, A7611, A7612, A7584)
Verification of EdDSA Public Key	AsymKeyPair-PubKeyVal	FIPS 186-5; Key verification		EDDSA KeyVer: (A7583, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614, A7584)
EdDSA Key Pair Generation	AsymKeyPair-KeyGen CKG	FIPS 186-5; Key pair generation		SHA2-512: (A7583, A7591, A7604, A7606, A7608, A7614, A7584) EDDSA KeyGen: (A7583, A7585, A7591, A7593, A7604, A7606, A7608, A7610, A7611, A7612, A7614, A7584) SHAKE-256: (A7585, A7593, A7610, A7611, A7612, A7584)
Key verification of ML-KEM decapsulation key	AsymKeyPair-KeyVer	FIPS 203. Decapsulation key verification		ML-KEM KeyGen: (A7585, A7593, A7610, A7611, A7612)
RSA key encapsulation	KTS-Encap	SP 800-56Br2; RSA key encapsulation		KTS-IFC: (A7583, A7584, A7591, A7604, A7606, A7608, A7614)

Name	Type	Description	Properties	Algorithms
RSA key un-encapsulation	KTS-Decap	SP 800-56Br2; RSA key un-encapsulation		KTS-IFC: (A7583, A7584, A7591, A7604, A7606, A7608, A7614)

Table 10: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 AES XTS

In accordance with FIPS 140-3 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. Key_1 and Key_2 shall be generated and/or established independently according to the rules for component symmetric keys from NIST SP 800-133r2, Section 6.3.

In addition, Section 4 of SP 800-38E states that 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.

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.

2.7.2 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 with SP 800-132 and FIPS 140-3 IG D.N, the following requirements shall be met:

- Derived keys shall only be used in storage applications. The MK shall not be used for other purposes. The module supports a minimum length of 112 bits for the MK or DPK.
- 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 supported by the module is 8 characters. Assuming the worst-case scenario of all digits, this results in the estimated probability of guessing the password to be at most 10^{-8} . 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, with a length of at least 128 bits (shorter salts are not supported by the module), shall be generated randomly using the SP 800-90Ar1 DRBG provided by the module.
- The iteration count shall be selected as large as possible, if the time required to generate the key using the entered password is acceptable for the users. If an iteration count less than 1000 is used, the module will indicate that invocation of the PBKDF2 service as non-approved.

2.7.3 Compliance to SP 800-56Arev3 Assurances

The module offers DH and ECDH shared secret computation services compliant to the SP 800-56Arev3 and meeting IG D.F scenario 2 path (1). To meet the required assurances listed in section 5.6 of SP 800-56Arev3, the

module shall be used together with an application that implements the “TLS protocol” and the following steps shall be performed.

1. The entity using the module must use the module's "Key pair generation" service for generating DH/ECDH ephemeral keys. This meets the assurances required by key pair owner defined in the section 5.6.2.1 of SP 800-56ARev3.
2. As part of the module's shared secret computation (SSC) service, the module internally performs the public key validation on the peer's public key passed in as input to the SSC function. This meets the public key validity assurance required by the sections 5.6.2.2.2 of SP 800-56ARev3.
3. The module does not support static keys therefore the "assurance of peer's possession of private key" is not applicable.

2.7.4 SHA-3 and SHAKE

The module implements the SHA-3 and SHAKE functions as both standalone functions and as part of higher-level algorithms (in compliance with FIPS 140-3 IG C.C) as detailed in Section 2.6 *Security Function Implementations* table with corresponding certificates.

2.7.5 Legacy Algorithms

The module utilizes the following legacy algorithms as defined in SP 800-131Arev2:

- ECDSA Signature Verification, under FIPS 186-4, allows verifying elliptic curve-based signatures with curve P-192.
- RSA Signature Verification, under FIPS 186-4, allows verifying signatures with X9.31 padding.

2.7.6 RSA Signatures

The module's RSA signature generation and verification implementations were CAVP tested with moduli sizes 2048, 3072, and 4096 bits. These moduli sizes are allowed by FIPS 140-3 IG C.F. The module supports moduli sizes larger than 4096 bits.

2.7.7 RSA Key Generation

The module's RSA key generation implementation was CAVP tested with moduli sizes 2048, 3072, 4096, 6144, and 8192 bits. These moduli sizes are allowed by FIPS 140-3 IG C.F. The module supports moduli sizes larger than 8192 bits. The number of Miller-Rabin tests is compliant with Table B.1 of FIPS 186-5.

2.7.8 Key Wrapping with AES

The module supports Key Wrapping and Unwrapping with AES in compliance with FIPS 140-3 IG D.G.

2.7.9 AES GCM IV

For TLS 1.2 and 1.3, the module offers the AES GCM implementation per Scenario 1 and 5 of the FIPS 140-3 IG C.H respectively. For TLS 1.2 the module is compliant with SP 800-52r2 Section 3.3.1 and the mechanism for IV generation is compliant with RFC 5288 and 8446. For TLS 1.3 defined in RFC8446, the module supports AES GCM cipher suites from Section 3.3.1 of SP800-52r2.

The module does not implement the TLS 1.2 or 1.3 protocol. The module's implementation of AES GCM is used together with an application that runs outside the module's cryptographic boundary. The design of the TLS

protocol implicitly ensures that the counter (the nonce_explicit part of the IV) does not exhaust the maximum number of possible values for a given session key.

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.

Alternatively, the module also implements GCM with internal IV generation per Scenario 2 of IG C.H. The internally generated IVs are always 96 bits and are generated using the approved DRBG internal to the module's boundary.

The module also provides a non-approved AES GCM encryption service which accepts arbitrary external IVs from the operator. The service can be requested by invoking the `EVP_EncryptInit_ex2` API function with a non-NULL IV value. When this is the case, the API will set a non-approved service indicator as described in Section 4.3.

2.7.10 Compliance to SP 800-56Br2 Assurances

To comply with the assurances found in Section 6.4 of SP 800-56Br2, 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 Section 4.3) must be used to generate RSA 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 key pair validation of the generated public key.

The operator must use the `EVP_PKEY_public_check()` API to perform partial public key validation of the peer public key, complying with Section 6.4.2.2 of SP 800-56Br2. The operator must also confirm the peer's possession of private key by using any method specified in Section 6.4.2.3 of SP 800-56Br2.

2.7.11 Compliance to SP 800-227

If an application uses an ephemeral key pair, the application shall ensure that the key pair is used for only one execution of key-establishment via a KEM and shall be destroyed as soon as possible after its use using the zeroization methods specified in section 9.3 of this security policy.

When using ML-KEM, the calling application shall select an ML-KEM parameter set with a security strength appropriate for the intended application, in accordance with SP 800-227 Section 4.2 (RM2).

The module shall be deployed on an appropriately secured device and operational environment that protects sensitive KEM data from unauthorized access or modification, in accordance with SP 800-227 Section 4.2, RM4.

The calling application shall ensure that the key-establishment process over the communication channel provides an application-appropriate level of integrity, in accordance with SP 800-227 Section 4.2, RM5.

When using an externally obtained ML-KEM static public encapsulation key, the operator shall ensure that the proof of possession of the private key has been established prior to invoking the ML-KEM encapsulation service. This assurance may be obtained using a trusted third party (e.g., a certificate authority) or another mechanism appropriate for the deployment, consistent with the guidance of SP 800-227 Section 4.5.

2.7.12 Compliance to FIPS 203

The module does not establish SSPs using an approved key encapsulation mechanism (KEM). However, it does offer some or all of the underlying KEM cryptographic functionality to be used by an external operator/application as part of an approved KEM.

The shared secret key output by ML-KEM can be used as a key for symmetric cryptography. If further key derivation is needed, the shared secret key shall be used with the SP 800-108 KDFs or SP 800-56C KDFs specified in section 2.6 of this security policy.

2.7.13 Compliance to FIPS 204

The module supports ML-DSA signature generation and verification using both the internal and external μ (Mu) interfaces. Per FIPS 204, the external Mu shall be provided by another cryptographic module. The module that is used to compute the Mu externally must have its SHAKE256 implementation CAVP certified.

2.8 RBG and Entropy

Cert Number	Vendor Name
E191	Chainguard

Table 11: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Chainguard CPU Time Jitter RNG Entropy Source	Non-Physical	Chainguard Image 20230214 on Amazon Linux 2023 on EC2 m7g.metal on Amazon Graviton3 AWS Graviton3; Chainguard Image 20230214 on Amazon Linux 2023 on EC2 m7i.metal-24xl on Intel Sapphire Rapids Xeon Platinum 8488C	256	Full entropy	SHA3-256 (A5446)

Table 12: Entropy Sources

The entropy source is statically compiled into the module and hence it is located within the cryptographic boundary of the module. As per the Public Use Document of entropy certificates E191, the entropy source provides full entropy of 256 bits.

In addition to the DRBG algorithms provided to the operator as random number generation services, the module internally uses two dedicated DRBG instances based on SP 800-90Ar1 to generate seeds for asymmetric key pairs and random numbers for security functions. The two instances are:

- Private DRBG: AES-256 CTR_DRBG with derivation function (256 bits of security strength). This DRBG is used to generate secret random values (e.g. during asymmetric key pair generation). It can be accessed using RAND_priv_bytes.

- Public DRBG: AES-256 CTR_DRBG with derivation function (256 bits of security strength). This DRBG is used to generate general purpose random values that do not need to remain secret (e.g. initialization vectors). It can be accessed using RAND_bytes.

All the random values needed internally in the implemented security functions are generated by one of the two aforementioned CTR DRBG instances.

2.9 Key Generation

The module implements Cryptographic Key Generation (CKG, vendor affirmed), compliant with SP 800-133r2. When random values are required, they are obtained from the SP 800-90Ar1 approved DRBG, compliant with Section 4 of SP 800-133r2. This method does not use the value V as described in Additional Comment 2 of FIPS 140-3 IG D.H.

The following methods are implemented: Safe primes key pair generation, RSA key pair generation, ECC (ECDH and ECDSA) key pair generation, EdDSA key pair generation, ML-KEM key pair generation, ML-DSA key pair generation, and SLH-DSA key pair generation.

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

Additionally, the module implements the following key derivation methods: (KDKDF, KDA OneStep, KDA TwoStep, HKDF, ANS X9.42 KDF (CVL), ANS X9.63 KDF (CVL), SSH KDF (CVL), TLS 1.2 KDF (CVL), TLS 1.3 KDF (CVL), PBKDF2

All certificates can be found in the Approved Algorithms table.

2.10 Key Establishment

The module implements key agreement, key transport, and key encapsulation methods as listed in the Security Function Implementations table in Section 2.6.

2.11 Industry Protocols

The module implements the SSH KDF (CVL) for use in the SSH protocol (RFC 4253 and RFC 6668).

GCM with internal IV generation in the approved mode is compliant with versions 1.2 and 1.3 of the TLS protocol (RFC 5288 and 8446) and shall only be used in conjunction with the TLS protocol. Additionally, the module implements the TLS 1.2 and TLS 1.3 key derivation functions for use in the TLS protocol.

For Diffie-Hellman, the module supports the use of the safe primes defined in RFC 3526 (IKE) and RFC 7919 (TLS). Note that the module only implements key pair generation, key pair verification, and shared secret computation. No other part of the IKE or TLS protocols is implemented (with the exception of the TLS 1.2 KDF (CVL) and 1.3 KDF (CVL)):

- IKE (RFC 3526): MODP-2048 (ID = 14), MODP-3072 (ID = 15), MODP-4096 (ID = 16), MODP-6144 (ID = 17), MODP-8192 (ID = 18)

- TLS (RFC 7919): ffdhe2048 (ID = 256), ffdhe3072 (ID = 257), ffdhe4096 (ID = 258), ffdhe6144 (ID = 259), ffdhe8192 (ID = 260)

For Elliptic Curve Diffie-Hellman, the module supports the NIST-defined P-224, P-256, P-384, and P-521 curves.

No parts of the SSH, TLS, or IKE protocols, other than those mentioned above, have been tested by the CAVP or 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
N/A	Data Output	API output parameters
N/A	Control Input	API function calls
N/A	Status Output	API return codes, error queue

Table 13: Ports and Interfaces

As a software-only module, the module does not have physical ports. The module does not implement a control output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 14: Roles

The module supports the Crypto Officer role only. This sole role is implicitly and always assumed by the operator of the module. No support is provided for a maintenance role. There are no concurrent operators.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Message Digest	Compute a message digest	EVP_DigestFinal returns 1	Message	Message digest	Message Digest with SHA Message Digest with SHAKE	Crypto Officer
Symmetric Encryption	Encrypt a plaintext	EVP_EncryptFinal_ex returns 1	AES Key, plaintext, IV	Ciphertext	Encryption with AES	Crypto Officer - AES Key: W,E
Symmetric Decryption	Decrypt a ciphertext	EVP_DecryptFinal_ex returns 1	AES Key, ciphertext, IV	Plaintext	Decryption with AES	Crypto Officer - AES Key: W,E
Authenticated Symmetric	Encrypt and authenticate a plaintext	GCM: OSSL_CIPHER_PARAM_AEAD_IV_GENERATED is 1; CCM: EVP_EncryptFinal_ex returns 1	AES Key, plaintext/key to be wrapped, IV	Ciphertext/wrapped key, MAC tag	Authenticated Encryption with AES	Crypto Officer - AES Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Encryption						- GCM IV: G,R,E
Authenticated Symmetric Decryption	Decrypt and authenticate a ciphertext	EVP_DecryptFinal_ex returns non-negative value	AES Key, ciphertext/wrapped key, MAC tag, IV	Plaintext or Failure	Authenticated Decryption with AES	Crypto Officer - AES Key: W,E - GCM IV: W,E
AES Message Authentication Generation	Compute a MAC tag using AES	EVP_MAC_final returns 1	AES Key, message	MAC tag	Message Authentication Generation with AES	Crypto Officer - AES Key: W,E
HMAC Message Authentication Generation	Compute a MAC tag using HMAC	OSSL_MAC_PARAM_FIPS_APPROVED_INDICATOR is 1	HMAC Key, message	MAC tag	Message Authentication Generation with HMAC	Crypto Officer - HMAC Key: W,E
KMAC Message Authentication Generation	Compute a MAC tag using KMAC	OSSL_MAC_PARAM_FIPS_APPROVED_INDICATOR is 1	KMAC Key, message	MAC tag	Message Authentication Generation with KMAC	Crypto Officer - KMAC Key: W,E
TLS KDF Key Derivation	TLS key derivation	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Pre-master secret	TLS Derived Key	Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF	Crypto Officer - Shared Secret: W,E - TLS Derived Key: G,R - TLS Master Secret: G,E,Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
KBKDF Key Derivation	Derive a key from a key-derivation key	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Key-Derivation Key	KBKDF Derived Key	Key Derivation with KBKDF	Crypto Officer - Key-Derivation Key: W,E - KBKDF Derived Key: G,R
ANS X9.42 Key Derivation	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Shared Secret	ANS X9.42 Derived Key	Key Derivation with ANS X9.42 KDF	Crypto Officer - ANS X9.42 Derived Key: G,R - Shared Secret: W,E
ANS X9.63 Key Derivation	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Shared Secret	ANS X9.63 Derived Key	Key Derivation with X9.63 KDF	Crypto Officer - Shared Secret: W,E - ANS X9.63 Derived Key: G,R
HKDF Key Derivation	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Shared Secret/Concatenated hybrid shared secret (ML-KEM and ECDH)/Concatenated hybrid shared secret (ML-	HKDF Derived key/Hybrid ML-KEM Shared Secret Key	Key Derivation with KDA HKDF	Crypto Officer - Shared Secret: W,E - HKDF Derived Key: G,R - Concatenated

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			KEM and X25519)			nated hybrid shared secret (ML-KEM and ECDH): W,E - Concatenated hybrid shared secret (ML-KEM and X25519): W,E - Hybrid ML-KEM Shared Secret Key: G,R
OneStep KDA Key Derivation	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Shared Secret	KDA OneStep Derived Key	Key Derivation with KDA OneStep	Crypto Officer - Shared Secret: W,E - KDA OneStep Derived Key: G,R
TwoStep KDA Key Derivation	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED_INDICATOR is 1	Shared Secret	KDA TwoStep Derived Key	Key Derivation with KDA TwoStep	Crypto Officer - Shared Secret: W,E - KDA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						TwoStep Derived Key: G,R
SSH KDF key derivatio n	Derive a key from a shared secret	OSSL_KDF_PARAM_FIPS_APPROVED _INDICATOR is 1	Shared Secret	SSH KDF Derived Key	Key Derivatio n with SSH KDF	Crypto Officer - Shared Secret: W,E - SSH KDF Derived Key: G,R
PBKDF Key Derivatio n	Derive a key from a passwor d	OSSL_KDF_PARAM_FIPS_APPROVED _INDICATOR is 1	Password	PBKDF Derived Key	Key Derivatio n with PBKDF2	Crypto Officer - Passwor d: W,E - PBKDF Derived Key: G,R
Random Number Generatio n	Generate random number	OSSL_RAND_PARAM_FIPS_APPROVE D_INDICATOR is 1	Number of bits	Random number	Random Number Generatio n with DRBG	Crypto Officer - Entropy Input: W,E,Z - DRBG Seed: G,E,Z - DRBG Internal State (V, Key): G,W,E - DRBG Internal State (V,

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						C): G,W,E
KAS-FFC-SSC Shared Secret Computation	Compute a shared secret	EVP_PKEY_derive returns 1	DH Private Key (owner), DH Public Key (peer)	Shared Secret	Shared Secret Computation	Crypto Officer - Shared Secret: G,R - DH Private Key: W,E - DH Public Key: W,E
KAS-ECC-SSC Shared Secret Computation	Compute a shared secret	OSSL_EXCHANGE_PARAM_FIPS_APPROVED_INDICATOR is 1	EC Private Key (owner), EC Public Key (peer)	Shared Secret	Shared Secret Computation	Crypto Officer - Shared Secret: G,R - EC Private Key: W,E - EC Public Key: W,E
RSA Digital Signature Generation	Generate a digital signature with RSA	OSSL_SIGNATURE_PARAM_FIPS_APPROVED_INDICATOR is 1 and OSSL_SIGNATURE_PARAM_FIPS_VERIFY_MESSAGE is 1	RSA Private Key, message, hash algorithm	Signature	Signature Generation with RSA	Crypto Officer - RSA Private Key: W,E
ECDSA Digital Signature Generation	Generate a digital signature with ECDSA	OSSL_SIGNATURE_PARAM_FIPS_APPROVED_INDICATOR is 1 and OSSL_SIGNATURE_PARAM_FIPS_VERIFY_MESSAGE is 1	EC Private Key, message, hash algorithm	Signature	Signature Generation with ECDSA	Crypto Officer - EC Private Key: W,E
RSA Digital	Verify a digital	OSSL_SIGNATURE_PARAM_FIPS_APPROVED_INDICATOR is 1 and	RSA Public Key, message,	Pass or Fail	Signature Verification	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Signature Verification	signature using RSA	OSSL_SIGNATURE_PARAM_FIPS_VERIFY_MESSAGE is 1	signature, hash algorithm		on with RSA Signature Verification with RSA (Legacy)	- RSA Public Key: W,E
ECDSA Digital Signature Verification	Verify a digital signature using ECDSA	OSSL_SIGNATURE_PARAM_FIPS_APPROVED_INDICATOR is 1 and OSSL_SIGNATURE_PARAM_FIPS_VERIFY_MESSAGE is 1	EC Public Key, message, signature, hash algorithm	Pass or Fail	Signature Verification with ECDSA Signature Verification with ECDSA (Legacy)	Crypto Officer - EC Public Key: W,E
RSA Key Pair Generation	Generate an RSA key pair	EVP_PKEY_keygen returns 1	Modulus bits	Module Generated RSA Private Key, Module Generated RSA Public Key	Key Pair Generation with RSA	Crypto Officer - Module Generated RSA Private Key: G,R - Module Generated RSA Public Key: G,R - Intermediate Key Generation Value: G,E,Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
ECDSA Key Pair Generation	Generate an EC key pair	OSSL_PKEY_PARAM_FIPS_APPROVED_INDICATOR is 1	Curve	Module Generated EC Private Key, Module Generated EC Public Key	Key Pair Generation with ECDSA	Crypto Officer - Module Generated EC Private Key: G,R - Module Generated EC Public Key: G,R - Intermediate Key Generation Value: G,E,Z
Safe Primes Key Pair Generation	Generate a DH key pair	EVP_PKEY_keygen returns 1	Group	Module Generated DH Private Key, Module Generated DH Public Key	Key Pair Generation with Safe Primes	Crypto Officer - Module Generated DH Private Key: G,R - Module Generated DH Public Key: G,R - Intermediate

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key Generation Value: G,E,Z
ECDSA Public Key Verification	Verify an EC public key	EVP_PKEY_check or EVP_PKEY_public_check returns 1	EC Public Key	Pass or Fail	Public Key Verification with ECDSA	Crypto Officer - EC Public Key: W,E
Safe Prime Key Pair Verification	Verify a DH key pair	EVP_PKEY_check or EVP_PKEY_public_check or EVP_PKEY_private_check returns 1	DH Private Key, DH Public Key	Pass or Fail	Key Pair Verification with Safe Primes	Crypto Officer - DH Private Key: W,E - DH Public Key: W,E
Show Version	Return the name and version information	None	None	Module name and version	None	Crypto Officer
Show Status	Return the module status	None	None	Module status	None	Crypto Officer
Self-Test	Perform the CASTs and integrity test	None	None	Pass or Fail of self-tests	None	Crypto Officer
Zeroization	Zeroize any SSP	None	An SSP	None	None	Crypto Officer - AES Key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- HMAC Key: Z - KMAC Key: Z - Key- Derivati on Key: Z - Shared Secret: Z - Passwor d: Z - PBKDF Derived Key: Z - KBKDF Derived Key: Z - ANS X9.42 Derived Key: Z - ANS X9.63 Derived Key: Z - HKDF Derived Key: Z - KDA OneStep Derived Key: Z - KDA TwoStep Derived Key: Z - TLS Derived Key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH KDF Derived Key: Z - Entropy Input: Z - DRBG Internal State (V, Key): Z - DRBG Seed: Z - DH Private Key: Z - DH Public Key: Z - EC Private Key: Z - EC Public Key: Z - RSA Private Key: Z - RSA Public Key: Z - Module Generat ed DH Private Key: Z - Module Generat ed DH Public Key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Module Generat ed EC Private Key: Z - Module Generat ed EC Public Key: Z - Module Generat ed RSA Private Key: Z - Module Generat ed RSA Public Key: Z - ML- DSA Public Key: Z - ML- DSA Private Key: Z - ML- KEM Public Key: Z - ML- KEM Private Key: Z - SLH- DSA Public

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: Z - SLH- DSA Private Key: Z - Module Generat ed ML- DSA Public Key: Z - Module Generat ed ML- DSA Private Key: Z - Module Generat ed ML- KEM Public Key: Z - Module Generat ed ML- KEM Private Key: Z - Module Generat ed SLH- DSA Public Key: Z - Module Generat

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						ed SLH- DSA Private Key: Z - ML- KEM Seed: Z - ML- DSA Seed: Z - SLH- DSA Seed: Z - ML- KEM Shared Secret: Z - EdDSA public key: Z - EdDSA private key: Z - Concate nated hybrid shared secret (ML- KEM and X25519): Z - Concate nated hybrid shared secret (ML- KEM and

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						ECDH): Z
ML-KEM Key Generation	Generate a ML-KEM key pair	EVP_PKEY_keygen returns 1	None	Module Generated ML-KEM Private Key, Module Generated ML-KEM Public Key	Key Generation with ML-KEM	Crypto Officer - Module Generated ML-KEM Public Key: G,R - Module Generated ML-KEM Private Key: G,R - ML-KEM Seed: G,E - Intermediate Key Generation Value: G,E,Z
ML-KEM Key Encapsulation	Perform encapsulation to compute an ML-KEM shared secret and ciphertext	OSSL_KEM_PARAM_FIPS_APPROVED_INDICATOR is 1	Encapsulation Key	Shared Secret, Ciphertext	Key Encapsulation with ML-KEM	Crypto Officer - ML-KEM Public Key: W,E - ML-KEM Shared

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Secret: G,R
ML-KEM Key Decapsulation	Perform decapsulation to compute an ML-KEM shared secret	EVP_PKEY_decapsulate returns 1	Decapsulation Key, Ciphertext	Shared Secret	Key Decapsulation with ML-KEM	Crypto Officer - ML-KEM Private Key: W,E - ML-KEM Shared Secret: G,R
ML-DSA Key Generation	Generate a ML-DSA key pair	EVP_PKEY_keygen returns 1	None	Module Generated ML-DSA Private Key, Module Generated ML-DSA Public Key	Key Generation with ML-DSA	Crypto Officer - Module Generated ML-DSA Public Key: G,R - Module Generated ML-DSA Private Key: G,R - ML-DSA Seed: G,E - Intermediate Key Generation

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Value: G,E,Z
ML-DSA Signature Generation (External Interface)	Generate a digital signature with ML-DSA	EVP_PKEY_sign returns 1	ML-DSA private Key, message, context string	Digital Signature	Signature Generation with ML-DSA	Crypto Officer - ML-DSA Private Key: W,E
ML-DSA Signature Generation (Internal Interface)	Generate a digital signature with ML-DSA	EVP_PKEY_sign returns 1	ML-DSA private Key, M', randomness rnd	Digital Signature	Signature Generation with ML-DSA	Crypto Officer - ML-DSA Private Key: W,E
ML-DSA Signature Verification (External Interface)	Verify a digital signature with ML-DSA	EVP_PKEY_verify returns 1	ML-DSA public key, message, signature, context string	Pass or Fail	Signature Verification with ML-DSA	Crypto Officer - ML-DSA Public Key: W,E
ML-DSA Signature Verification (Internal Interface)	Verify a digital signature with ML-DSA	EVP_PKEY_verify returns 1	ML-DSA public key, M', signature	Pass or Fail	Signature Verification with ML-DSA	Crypto Officer - ML-DSA Public Key: W,E
SLH-DSA Key Generation	Generate a SLH-DSA key pair	EVP_PKEY_keygen returns 1	None	Module Generated SLH-DSA Private Key, Module Generated SLH-DSA Public Key	Key Generation with SLH-DSA	Crypto Officer - Module Generated SLH-DSA Public Key: G,R - Module

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Generated SLH-DSA Private Key: G,R - SLH-DSA Seed: G,E - Intermediate Key Generation Value: G,E,Z
SLH-DSA Signature Generation (External Interface)	Generate a digital signature with SLH-DSA	EVP_PKEY_sign returns 1	SLH-DSA private key, message, context string	Digital Signature	Signature Generation with SLH-DSA	Crypto Officer - SLH-DSA Private Key: W,E
SLH-DSA Signature Generation (Internal Interface)	Generate a digital signature with SLH-DSA	EVP_PKEY_sign returns 1	SLH-DSA private key, M', randomness addrnd	Digital Signature	Signature Generation with SLH-DSA	Crypto Officer - SLH-DSA Private Key: W,E
SLH-DSA Signature Verification (External Interface)	Verify a digital signature with SLH-DSA	EVP_PKEY_verify returns 1	SLH-DSA public key, message, signature, context string	Pass or Fail	Signature Verification with SLH-DSA	Crypto Officer - SLH-DSA Public Key: W,E
SLH-DSA Signature Verification	Verify a digital signature	EVP_PKEY_verify returns 1	SLH-DSA public key, M', signature	Pass or Fail	Signature Verification	Crypto Officer - SLH-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
on (Internal Interface)	e with SLH- DSA				on with SLH-DSA	DSA Public Key: W,E
MLX- KEM Hybrid Key Generation	Generate an ML- KEM key pair and X25519 key pair	EVP_PKEY_keygen returns 1	None	Module Generated ML-KEM Encapsulation Key, Module Generated ML-KEM Decapsulation Key, X25519 public key, X25519 private key	Key Generation with KEM- Hybrid- MLX- KEM	Crypto Officer - ML- KEM Seed: G,E,Z - Module Generated ML- KEM Public Key: G,R - Module Generated ML- KEM Private Key: G,R - Intermediate Key Generation Value: G,E,Z
MLX- KEM Key Encapsulation	Perform hybrid ML- KEM and X25519 encapsulation to produce	OSSL_KEM_PARAM_FIPS_APPROVED _INDICATOR parameter of ML-KEM EVP_PKEY_CTX is 1	X public key, X private key, ML-KEM encapsulation key	Concatenated hybrid shared secret (ML-KEM and X25519), ciphertext	Key Encapsulation with KEM- Hybrid- MLX- KEM	Crypto Officer - ML- KEM Public Key: W,E - Concatenate

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	an ML-KEM ciphertext and concatenated hybrid shared secret.					nated hybrid shared secret (ML-KEM and X25519): G,R
MLX-KEM Key Decapsulation	Perform hybrid ML-KEM and X25519 decapsulation to produce a concatenated hybrid shared secret.	EVP_PKEY_decapsulate returns 1	X private key, ML-KEM decapsulation key, ciphertext	Concatenated hybrid shared secret (ML-KEM and X25519)	Key Decapsulation with KEM-Hybrid-MLX-KEM	Crypto Officer - ML-KEM Private Key: W,E - Concatenated hybrid shared secret (ML-KEM and X25519): G,R
ML-KEM Key Verification	Verification of encapsulation key or decapsulation key	EVP_PKEY_check or EVP_PKEY_public_check or EVP_PKEY_private_check returns 1	ML-KEM encapsulation key or decapsulation key	Success/failure	Key verification of ML-KEM encapsulation key Key verification of ML-KEM decapsulation key	Crypto Officer - ML-KEM Public Key: W,E - ML-KEM Private Key: W,E
EdDSA Digital Signature	Generate a digital signature	EVP_PKEY_sign returns 1	EdDSA private key, message	Signature	Signature Generation	Crypto Officer - EdDSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Generation	e with ECDSA				n with EdDSA	private key: W,E
EdDSA Digital Verification	Verify a digital signature with EdDSA	EVP_PKEY_verify returns 1	EdDSA public key, message, signature	Success/failure	Signature Verification with EdDSA	Crypto Officer - EdDSA public key: W,E
EdDSA Public Key Verification	Verify an EdDSA public key	EVP_PKEY_check or EVP_PKEY_public_check returns 1	EdDSA public key	Success/failure	Verification of EdDSA Public Key	Crypto Officer - EdDSA public key: W,E
EdDSA Key Pair Generation	Generate an EdDSA key pair	EVP_PKEY_keygen returns 1	Desired EdDSA curve	EdDSA public key, EdDSA private key	EdDSA Key Pair Generation	Crypto Officer - Module generated EdDSA public key: G,R - Module generated EdDSA private key: G,R - Intermediate Key Generation Value: G,E,Z
Hybrid-ML-KEM	Generate an ML-KEM	EVP_PKEY_keygen returns 1	Desired ML-KEM	ML-KEM encapsulation key, ML-	Key Generation with	Crypto Officer -

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
ECDH Key Pair Generation	key pair and an EC P-curve key pair		parameter set and P curve	KEM decapsulation key, EC public key, EC private key	Hybrid-ML-KEM-ECDH	Module Generated EC Private Key: G,R - Module Generated EC Public Key: G,R - Module Generated ML-KEM Public Key: G,R - Module Generated ML-KEM Private Key: G,R - Intermediate Key Generation Value: G,E,Z
Hybrid-ML-KEM-ECDH encapsulation	Perform hybrid ML-KEM and ECDH	OSSL_KEM_PARAM_FIPS_APPROVED_INDICATOR parameter of ML-KEM EVP_PKEY_CTX is 1	ML-KEM encapsulation key, EC public key, EC private key	Concatenated hybrid shared secret (ML-KEM and ECDH),	Key Encapsulation with Hybrid-ML-	Crypto Officer - EC Public Key: R,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	encapsulation to produce ML-KEM ciphertext and concatenated hybrid shared secret.			EC public key	KEM-ECDH	- ML-KEM Public Key: W,E - Concatenated hybrid shared secret (ML-KEM and ECDH): G,R - EC Private Key: W,E
Hybrid-ML-KEM-ECDH decapsulation	Perform hybrid ML-KEM and ECDH decapsulation to produce a concatenated hybrid shared secret.	EVP_PKEY_decapsulate returns 1	ML-KEM decapsulation key, EC public key, EC private key	Concatenated hybrid shared secret (ML-KEM and ECDH)	Key Decapsulation with Hybrid-ML-KEM-ECDH	Crypto Officer - ML-KEM Private Key: W,E - EC Private Key: W,E - EC Public Key: W,E - Concatenated hybrid shared secret (ML-KEM

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						and ECDH): G,R
RSA key encapsulation	Key encapsulation with RSA	EVP_PKEY_encrypt returns 1	RSA Public Key, Key to encapsulate	Encapsulated key	RSA key encapsulation	Crypto Officer - RSA Public Key: W,E
RSA key un-encapsulation	Key un-encapsulation with RSA	EVP_PKEY_decrypt returns 1	RSA Private Key, Key to un-encapsulate	Un-encapsulated key	RSA key un-encapsulation	Crypto Officer - RSA Private Key: W,E
RSA-based shared secret computation	KAS-IFC-SSC	EVP_PKEY_encapsulate or EVP_PKEY_decapsulate returns 1	RSA private key, RSA public key	Shared secret	Shared Secret Computation	Crypto Officer - RSA Private Key: W,E - RSA Public Key: W,E

Table 15: 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 Approved Services table and the Non-Approved Services table define the services that utilize approved and non-approved security functions in this module. For the respective tables, 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.

To interact with the module, a calling application must use the EVP API layer provided by OpenSSL. This layer will delegate the request to the FIPS provider, which will in turn perform the requested service. Additionally, this EVP API layer can be used to retrieve the approved service indicator for the module.

4.4 Non-Approved Services

Name	Description	Algorithms	Role
Encryption	AES GCM (external IV)	AES GCM encryption with external IV	CO
ML-KEM encapsulation with user-provided randomness	Key encapsulation	ML-KEM encapsulation with user-provided randomness	CO
Hybrid ML-KEM and X25519 encapsulation with user-provided randomness	Hybrid ML-KEM and X25519 encapsulation with user-provided randomness for use in ML-KEM encapsulation	ML-KEM encapsulation with user-provided randomness	CO
Non-approved Digital Signature Generation	Non-approved RSA or ECDSA signature generation	RSA SigGen with X9.31 padding or with modulus length < 2048 bits or with PSS salt length > digest length or without hashing (primitive) ECDSA SigGen with P-192 or with SHA2-512/256; ECDSA deterministic SigGen with P-192 or with SHA2-512/256; ECDSA SigGen component with P-192	CO
Non-approved Digital Signature Verification	Non-approved RSA or ECDSA signature verification	RSA SigVer with modulus length < 2048 bits or with PSS salt length > digest length or without hashing (primitive) ECDSA SigVer with SHA2-512/256; ECDSA SigVer component	CO
Non-approved Key Derivation	Non-approved PBKDF, TLS, KBKDF, SSH, One step KDF, two step KDF, HKDF, ANS X9.42 KDF, ANS X9.63 KDF	PBKDF2 with password length < 8 characters or salt length < 128 bits or output key length < 112 bits or iteration count < 1000 TLS 1.2 KDF without extended master secret or with SHA2-512/256 or with SHA-3 functions or with input secret length < 112 bits KBKDF with input key length < 112 bits SSH KDF with SHA2-512/256 or with SHA-3 functions or with input secret length < 112 bits TLS 1.3 KDF with SHA2-512 or SHA2-512/256 or SHA-3 functions or input secret length < 112 bits One step KDF, two step KDF, HKDF, ANS X9.42 KDF, ANS X9.63 KDF with input secret length < 112 bits	CO

Name	Description	Algorithms	Role
Non-approved Random Bit Generation	Non-approved HMAC_DRBG or Hash_DRBG	Hash_DRBG or HMAC_DRBG with SHA2-384 or SHA2-512/256 or SHA-3 functions or KMAC	CO
Non-approved Message Authentication Code Generation	Non-approved HMAC or KMAC	HMAC with key length < 112 bits KMAC with key length < 112 bits or tag length < 32 bits	CO
Non-approved Shared Secret Computation	Non-approved elliptic curve Diffie-Hellman	ECDH with P-192	CO
Non-approved Key Generation	Non-approved key generation	ECDSA KeyGen with P-192	CO

Table 16: Non-Approved Services

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 embedded in the fipsmodule.cnf file that was computed at build time. The key used for the HMAC calculation is contained within 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. This will perform (among others) the software integrity test.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied:

The module shall be installed as stated in Section 11. 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

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 mechanism, 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. The module does not perform persistent storage of SSPs.	Dynamic

Table 17: Storage Areas

SSPs are provided to the module by the calling application and are destroyed when released by the appropriate API function calls. The module does not perform persistent storage of SSPs.

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 18: 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: <code>EVP_CIPHER_CTX_free()</code> clears and frees symmetric cipher context, <code>EVP_MAC_CTX_free()</code> clears and frees MAC context, <code>EVP_KDF_CTX_free()</code> clears and frees KDF context, <code>EVP_RAND_CTX_free()</code> clears and frees DRBG context, <code>EVP_PKEY_free()</code> clears and frees asymmetric key pair structures	Memory occupied by SSPs is overwritten with zeroes and then it is released, which renders the SSP values irretrievable. The completion of the zeroization routine indicates that the	By calling the cipher related zeroization API

Zeroization Method	Description	Rationale	Operator Initiation
		zeroization procedure succeeded.	
Automatic	Automatically zeroized by the module when no longer needed	Memory occupied by SSPs is overwritten with zeroes, which renders the SSP values irretrievable.	N/A
Module reset	De-allocates the volatile memory used to store SSPs	Volatile memory used by the module is overwritten within nanoseconds when power is removed.	By unloading and reloading the module

Table 19: SSP Zeroization Methods

All data output is inhibited during zeroization.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES Key	Used for encryption, decryption, and message authentication	128, 192, 256 bits - 128, 192, 256 bits	Symmetric key - CSP			Encryption with AES Decryption with AES Authenticated Encryption with AES Authenticated Decryption with AES Message Authentication Generation with AES
HMAC Key	Used for hash-based message authentication	112-524288 bits - 112-256 bits	Symmetric key - CSP			Message Authentication Generation with HMAC
KMAC Key	Used for message authentication	128-1024 bits - 112-256 bits	Symmetric key - CSP			Message Authentication Generation with KMAC

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Key-Derivation Key	Used for key derivation	112-4096 bits - 112-256 bits	Symmetric key - CSP			Key Derivation with KBKDF
Shared Secret	Generated by shared secret computation and used for key derivation	224-8192 bits - 112-256 bits	Shared secret - CSP		Shared Secret Computation	Key Derivation with KDA OneStep Key Derivation with KDA TwoStep Key Derivation with KDA HKDF Key Derivation with ANS X9.42 KDF Key Derivation with X9.63 KDF Key Derivation with SSH KDF Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF
Password	Used for password-based key derivation	At least 8 characters - N/A	Password - CSP			Key Derivation with PBKDF2
PBKDF Derived Key	Generated by password-based key derivation	112-4096 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with PBKDF2		
KBKDF Derived Key	Generated by key-based key derivation	112-4096 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with KBKDF		
ANS X9.42 Derived Key	Generated by ANS X9.42 key derivation	128-4096 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with ANS X9.42 KDF		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
ANS X9.63 Derived Key	Generated by ANS X9.63 key derivation	128-4096 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with X9.63 KDF		
HKDF Derived Key	Generated by HKDF key derivation	224-8192 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with KDA HKDF		
KDA OneStep Derived Key	Generated by OneStep KDA key derivation	2048 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with KDA OneStep		
KDA TwoStep Derived Key	Generated by TwoStep KDA key derivation	2048 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with KDA TwoStep		
TLS Derived Key	Generated by TLS KDF key derivation	112-1024 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF		
SSH KDF Derived Key	Generated by SSH KDF key derivation	112-256 bits - 112-256 bits	Symmetric key - CSP	Key Derivation with SSH KDF		
Entropy Input	Used for random number generation and seeding a DRBG (compliant with IG D.L)	128-384 bits - 128-384 bits	Entropy input - CSP			Random Number Generation with DRBG
DRBG Seed	Used for random number generation (compliant with IG D.L)	Counter DRBG: 256, 320, 348 bits; Hash_DRBG: 440, 888 bits; HMAC DRBG: 320, 512, 1024 bits - Counter	Seed - CSP	Random Number Generation with DRBG		Random Number Generation with DRBG

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		DRBG: 128, 192, 256 bits; Hash_DRBG: 128, 256 bits; HMAC DRBG: 128, 256 bits				
DRBG Internal State (V, Key)	Used for random number generation (compliant with IG D.L)	Counter DRBG: 256, 320, 348 bits; HMAC DRBG: 320, 512, 1024 bits - Counter DRBG: 128, 192, 256 bits; HMAC DRBG: 128, 256 bits	Internal state - CSP	Random Number Generation with DRBG		Random Number Generation with DRBG
DRBG Internal State (V, C)	Used for random number generation (compliant with IG D.L)	880, 1776 bits - 128, 256 bits	Internal state - CSP	Random Number Generation with DRBG		Random Number Generation with DRBG
DH Private Key	Used for shared secret computation and key pair verification	2048-8192 bits - 112-200 bits	Private key - CSP			Key Pair Verification with Safe Primes Shared Secret Computation
DH Public Key	Used for shared secret computation and key pair verification	2048-8192 bits - 112-200 bits	Public key - PSP			Key Pair Verification with Safe Primes Shared Secret Computation
EC Private Key	Used for shared secret computation, digital signature generation, key pair verification	P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571 bits - 112-256 bits	Private key - CSP			Signature Generation with ECDSA Shared Secret Computation Key Encapsulation with Hybrid-ML-KEM-

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						ECDH Key Decapsulation with Hybrid- ML-KEM- ECDH
EC Public Key	Used for shared secret computation, signature verification, and key pair verification	P-192, P-224, P-256, P-384, P-521, K-163, K-233, K-283, K-409, K-571, B-163, B-233, B-283, B-409, B-571 bits - 80-256 bits	Public key - PSP			Signature Verification with ECDSA Public Key Verification with ECDSA Shared Secret Computation Key Encapsulation with Hybrid- ML-KEM- ECDH Key Decapsulation with Hybrid- ML-KEM- ECDH
RSA Private Key	Used for signature generation	2048-16384 bits - 112-256 bits	Private key - CSP			Signature Generation with RSA Shared Secret Computation RSA key un- encapsulation
RSA Public Key	Used for signature verification	2048-16384 bits - 112-256 bits	Public key - PSP			Signature Verification with RSA Shared Secret Computation RSA key encapsulation
Module Generated DH Private Key	DH private key generated by the module	2048-8192 bits - 112-200 bits	Private key - CSP	Key Pair Generation with Safe Primes		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Module Generated DH Public Key	DH public key generated by the module	2048-8192 bits - 112-200 bits	Public key - PSP	Key Pair Generation with Safe Primes		
Module Generated EC Private Key	EC private key generated by the module	P-224, P-256, P-384, P-521 - 112-256 bits	Private key - CSP	Key Pair Generation with ECDSA		
Module Generated EC Public Key	EC public key generated by the module	P-224, P-256, P-384, P-521 - 112-256 bits	Public key - PSP	Key Pair Generation with ECDSA		
Module Generated RSA Private Key	RSA private key generated by the module	2048-16384 bits - 112-256 bits	Private key - CSP	Key Pair Generation with RSA		
Module Generated RSA Public Key	RSA public key generated by the module	2048-16384 bits - 112-256 bits	Public key - PSP	Key Pair Generation with RSA		
Intermediate Key Generation Value	Used for key pair generation	224-16384 bits - 112-256 bits	Intermediate value - CSP	Key Pair Generation with ECDSA Key Pair Generation with RSA Key Pair Generation with Safe Primes Key Generation with ML-KEM Key Generation with ML-DSA Key Generation with SLH-DSA Key Generation with KEM-Hybrid-MLX-		Key Pair Generation with ECDSA Key Pair Generation with RSA Key Pair Generation with Safe Primes Key Generation with ML-KEM Key Generation with ML-DSA Key Generation with SLH-DSA Key Generation with KEM-Hybrid-MLX- Key

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
				KEM Key Generation with Hybrid- ML-KEM- ECDH EdDSA Key Pair Generation		Generation with Hybrid- ML-KEM- ECDH EdDSA Key Pair Generation
ML-DSA Public Key	Used for signature verification	ML-DSA-44: 10496 bits; ML- DSA-65: 15616 bits; ML-DSA-87: 20736 bits - ML- DSA-44: 128 bits; ML-DSA-65: 192 bits; ML-DSA-87: 256 bits	Public key - PSP			Signature Verification with SLH-DSA
ML-DSA Private Key	Used for signature generation	ML-DSA-44: 20480 bits; ML- DSA-65: 32256 bits; ML-DSA-87: 39168 bits - ML- DSA-44: 128 bits; ML-DSA-65: 192 bits; ML-DSA-87: 256 bits	Private key - CSP			Signature Generation with ML-DSA
ML-KEM Public Key	Used for key encapsulation	ML-KEM-512: 6400 bits; ML- KEM-768: 9472 bits; ML-KEM- 1024: 12544 bits - ML-KEM-512: 128 bits; ML- KEM-768: 192 bits; ML-KEM- 1024: 256 bits	Encapsulation key - PSP			Key Encapsulation with ML-KEM
ML-KEM Private Key	Used for key decapsulation	ML-KEM-512: 13056 bits; ML- KEM-768: 19200 bits; ML-KEM- 1024: 25344 bits - ML-KEM-512:	Decapsulation key - CSP			Key Decapsulation with ML-KEM

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits				
SLH-DSA Public Key	Used for signature verification	SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 256 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 384 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 512 bits - SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 128 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 192 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 256 bits	Public key - PSP			Signature Verification with SLH-DSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		SHAKE-256f: 256 bits				
SLH-DSA Private Key	Used for signature generation	SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 512 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 768 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 1024 bits - SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 128 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 192 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 256 bits	Private key - CSP			Signature Generation with SLH-DSA

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Module Generated ML-DSA Public Key	Used for signature verification	ML-DSA-44: 10496 bits; ML-DSA-65: 15616 bits; ML-DSA-87: 20736 bits - ML-DSA-44: 128 bits; ML-DSA-65: 192 bits; ML-DSA-87: 256 bits	Public key - PSP	Key Generation with ML-DSA		
Module Generated ML-DSA Private Key	Used for signature generation	ML-DSA-44: 20480 bits; ML-DSA-65: 32256 bits; ML-DSA-87: 39168 bits - ML-DSA-44: 128 bits; ML-DSA-65: 192 bits; ML-DSA-87: 256 bits	Private key - CSP	Key Generation with ML-DSA		
Module Generated ML-KEM Public Key	Used for key encapsulation	ML-KEM-512: 6400 bits; ML-KEM-768: 9472 bits; ML-KEM-1024: 12544 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Encapsulation key - PSP	Key Generation with ML-KEM		
Module Generated ML-KEM Private Key	Used for key decapsulation	ML-KEM-512: 13056 bits; ML-KEM-768: 19200 bits; ML-KEM-1024: 25344 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Decapsulation key - CSP	Key Generation with ML-KEM		
Module Generated SLH-DSA Public Key	Used for signature verification	SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-	Public key - PSP	Key Generation with SLH-DSA		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		SHAKE-128f: 256 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 384 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 512 bits - SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 128 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 192 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 256 bits				
Module Generated SLH-DSA Private Key	Used for signature generation	SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 512 bits; SLH-DSA-	Private key - CSP	Key Generation with SLH-DSA		

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
		SHA2-192s, SLH-D SHA2-192f, SLH-D SHA2-192s, SLH-D SHA2-192f, SLH-D SHA2-192f: 768 bits; SLH-D SHA2-256s, SLH-D SHA2-256f, SLH-D SHA2-256s, SLH-D SHA2-256f: 1024 bits - SLH-D SHA2-128s, SLH-D SHA2-128f, SLH-D SHA2-128s, SLH-D SHA2-128f: 128 bits; SLH-D SHA2-192s, SLH-D SHA2-192f, SLH-D SHA2-192s, SLH-D SHA2-192f: 192 bits; SLH-D SHA2-256s, SLH-D SHA2-256f, SLH-D SHA2-256s, SLH-D SHA2-256f: 256 bits				
ML-KEM Seed	Seed used for Key Generation with ML-KEM	512 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Seed - CSP	Random Number Generation with DRBG		Key Generation with ML-KEM Key Generation with KEM-Hybrid-MLX-KEM

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
ML-DSA Seed	Seed used for Key Generation with ML-DSA	256 bits - ML-DSA-44: 128 bits; ML-DSA-65: 192 bits; ML-DSA-87: 256 bits	Seed - CSP	Random Number Generation with DRBG		Key Generation with ML-DSA
SLH-DSA Seed	Seed used for Key Generation with SLH-DSA	384, 576, 768 bits - SLH-DSA-SHA2-128s, SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-128f: 128 bits; SLH-DSA-SHA2-192s, SLH-DSA-SHA2-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-192f: 192 bits; SLH-DSA-SHA2-256s, SLH-DSA-SHA2-256f, SLH-DSA-SHAKE-256s, SLH-DSA-SHAKE-256f: 256 bits	Seed - CSP	Random Number Generation with DRBG		Key Generation with SLH-DSA
ML-KEM Shared Secret	Shared secret used in ML-KEM	256 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Shared secret - CSP		Key Encapsulation with ML-KEM Key Decapsulation with ML-KEM	Key Derivation with KBKDF Key Derivation with KDA OneStep Key Derivation with KDA TwoStep Key Derivation with KDA HKDF Key Derivation with ANS X9.42 KDF Key Derivation with X9.63

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						KDF Key Derivation with SSH KDF Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF Key Derivation with PBKDF2
Concatenated hybrid shared secret (ML-KEM and X25519)	Shared secret output by KEM-Hybrid-MLX-KEM	512 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Shared secret - CSP		Key Encapsulation with KEM-Hybrid-MLX-KEM Key Decapsulation with KEM-Hybrid-MLX-KEM	Key Derivation with KDA HKDF
Concatenated hybrid shared secret (ML-KEM and ECDH)	Shared secret output by Hybrid-ML-KEM-ECDH	ML-KEM with ECDH P-256: 512 bits; ML-KEM with ECDH P-384: 640 bits - ML-KEM-512: 128 bits; ML-KEM-768: 192 bits; ML-KEM-1024: 256 bits	Shared secret - CSP		Key Encapsulation with Hybrid-ML-KEM-ECDH Key Decapsulation with Hybrid-ML-KEM-ECDH	Key Derivation with KDA HKDF
Hybrid ML-KEM Shared Secret Key	Shared secret key post combining via the approved HKDF key combiner	224-8192 bits - 112-256 bits	Shared secret key - CSP	Key Derivation with KDA HKDF		
EdDSA public key	Used in digital signature verification and key verification	Ed25519, Ed448 - 128, 224 bits	Public key - PSP			Signature Verification with EdDSA Verification of EdDSA Public Key

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
EdDSA private key	Used in digital signature generation	Ed25519, Ed448 - 128, 224 bits	Private key - CSP			Signature Generation with EdDSA
Module generated EdDSA public key	Used in digital signature verification and key verification	Ed25519, Ed448 - 128, 224 bits	Public key - PSP	EdDSA Key Pair Generation		
Module generated EdDSA private key	Used in digital signature generation	Ed25519, Ed448 - 128, 224 bits	Private key - CSP	EdDSA Key Pair Generation		
GCM IV	IV used for Authenticated encryption; Authenticated decryption	96 bits - N/A	Initialization vector - PSP	Authenticated Encryption with AES		Authenticated Encryption with AES Authenticated Decryption with AES
TLS Master Secret	TLS Master Secret used for deriving the TLS Derived Secret	384 bits - 112-256 bits	Secret - CSP	Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF		Key Derivation with TLS 1.2 KDF Key Derivation with TLS 1.3 KDF

Table 20: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	
HMAC Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	
KMAC Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Key-Derivation Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	KBKDF Derived Key:Derives
Shared Secret	API input parameters API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	DH Private Key:Generated From DH Public Key:Generated From EC Private Key:Generated From EC Public Key:Generated From
Password	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	PBKDF Derived Key:Derives
PBKDF Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Password:Derived From
KBKDF Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Key-Derivation Key:Derived From
ANS X9.42 Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From
ANS X9.63 Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From
HKDF Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From
KDA OneStep Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From
KDA TwoStep Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From
TLS Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	TLS Master Secret:Derived From
SSH KDF Derived Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Shared Secret:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Entropy Input		RAM:Plaintext	Until the DRBG has completed instantiation or the module is reset	Automatic Module reset	DRBG Seed:Generates
DRBG Seed		RAM:Plaintext	Until the DRBG has completed instantiation or the module is reset	Automatic Module reset	DRBG Internal State (V, Key):Generates DRBG Internal State (V, C):Generates Entropy Input:Generated From
DRBG Internal State (V, Key)		RAM:Plaintext	Until the DRBG has been un-instantiation, or the cipher handle has been freed, or the module is reset	Free cipher handle Automatic Module reset	DRBG Seed:Generated From
DRBG Internal State (V, C)		RAM:Plaintext	Until the DRBG has been un-instantiation, or the cipher handle has been freed, or the module is reset	Free cipher handle Automatic Module reset	DRBG Seed:Generated From
DH Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	DH Public Key:Paired With Shared Secret:Derives
DH Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	DH Private Key:Paired With Shared Secret:Generates
EC Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	EC Public Key:Paired With Shared Secret:Generates
EC Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	EC Private Key:Paired With Shared Secret:Generates
RSA Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	RSA Public Key:Paired With
RSA Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	RSA Private Key:Paired With
Module Generated DH Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated DH Public Key:Paired With Intermediate Key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Generation Value:Generated From
Module Generated DH Public Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated DH Private Key:Paired With Intermediate Key Generation Value:Generated From
Module Generated EC Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	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	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated EC Private Key:Paired With Intermediate Key Generation Value:Generated From
Module Generated RSA Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	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	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated RSA Private Key:Paired With Intermediate Key Generation Value:Generated From
Intermediate Key Generation Value		RAM:Plaintext	From service invocation to service completion	Automatic	Module Generated DH Private Key:Generates Module Generated DH Public Key:Generates Module Generated EC Private Key:Generates Module Generated EC Public Key:Generates Module Generated RSA Private Key:Generates Module Generated RSA Public Key:Generates Module generated EdDSA public key:Generates Module generated

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					EdDSA private key:Generates Module Generated SLH-DSA Public Key:Generates Module Generated SLH-DSA Private Key:Generates Module Generated ML-DSA Public Key:Generates Module Generated ML-DSA Private Key:Generates Module Generated ML-KEM Public Key:Generates Module Generated ML-KEM Private Key:Generates
ML-DSA Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-DSA Private Key:Paired With
ML-DSA Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-DSA Public Key:Paired With
ML-KEM Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-KEM Private Key:Paired With Shared Secret:Encapsulates
ML-KEM Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-KEM Public Key:Paired With Shared Secret:Decapsulates
SLH-DSA Public Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	SLH-DSA Private Key:Paired With
SLH-DSA Private Key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	SLH-DSA Public Key:Paired With
Module Generated ML-DSA Public Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated ML-DSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					ML-DSA Seed:Derived From Intermediate Key Generation Value:Generated from
Module Generated ML-DSA Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated ML-DSA Public Key:Paired With ML-DSA Seed:Derived From Intermediate Key Generation Value:Generated from
Module Generated ML-KEM Public Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated ML-KEM Private Key:Paired With ML-KEM Seed:Derived From Intermediate Key Generation Value:Generated from
Module Generated ML-KEM Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated ML-KEM Public Key:Paired With ML-KEM Seed:Derived From Intermediate Key Generation Value:Generated from
Module Generated SLH-DSA Public Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated SLH-DSA Private Key:Paired With SLH-DSA Seed:Derived From Intermediate Key Generation Value:Generated from
Module Generated SLH-DSA Private Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module Generated SLH-DSA Public Key:Paired With SLH-DSA Seed:Derived From Intermediate Key

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					Generation Value:Generated from
ML-KEM Seed	API output parameters	RAM:Plaintext	From service invocation to service completion, or until cipher handle is freed, or until the module is reset	Free cipher handle Automatic Module reset	Module Generated ML-KEM Private Key:Generates Module Generated ML-KEM Public Key:Generates
ML-DSA Seed	API output parameters	RAM:Plaintext	From service invocation to service completion, or until cipher handle is freed, or until the module is reset	Free cipher handle Automatic Module reset	Module Generated ML-DSA Private Key:Generates Module Generated ML-DSA Public Key:Generates
SLH-DSA Seed		RAM:Plaintext	From service invocation to service completion	Automatic	Module Generated SLH-DSA Private Key:Generates Module Generated SLH-DSA Public Key:Generates
ML-KEM Shared Secret	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-KEM Public Key:Wrapped by ML-KEM Private Key:Unwrapped by
Concatenated hybrid shared secret (ML-KEM and X25519)	API input parameters API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-KEM Public Key:Wrapped by ML-KEM Private Key:Unwrapped by Hybrid ML-KEM Shared Secret Key:Used to derive
Concatenated hybrid shared secret (ML-KEM and ECDH)	API input parameters API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	ML-KEM Public Key:Wrapped by ML-KEM Private Key:Unwrapped by EC Private Key:Derived from EC Public Key:Derived from Hybrid ML-KEM Shared Secret Key:Used to derive

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Hybrid ML-KEM Shared Secret Key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Concatenated hybrid shared secret (ML-KEM and X25519):Derived from Concatenated hybrid shared secret (ML-KEM and ECDH):Derived from
EdDSA public key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	EdDSA private key:Paired With
EdDSA private key	API input parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	EdDSA public key:Paired With
Module generated EdDSA public key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module generated EdDSA private key:Paired With Intermediate Key Generation Value:Generated from
Module generated EdDSA private key	API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	Module generated EdDSA public key:Paired With Intermediate Key Generation Value:Generated from
GCM IV	API input parameters API output parameters	RAM:Plaintext	Until cipher handle is freed or module is reset	Free cipher handle Module reset	
TLS Master Secret		RAM:Plaintext	From service invocation to service completion	Automatic	TLS Derived Key:Derives

Table 21: SSP Table 2

SSPs, including CSPs, are directly imported as input parameters and exported as output parameters from the module. Because these SSPs are only transiently used for a specific service, they are, by definition, exclusive between approved and non-approved services.

10 Self-Tests

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 does not return control to the calling application until the tests are completed. If any of these tests fail, the module transitions to the error state.

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A7583)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7584)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7591)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7604)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7606)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7608)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so
HMAC-SHA2-256 (A7614)	256-bit key	Message Authentication	SW/FW Integrity	Module becomes operational and services are available for use	Integrity test for fips.so

Table 22: Pre-Operational Self-Tests

The pre-operational software integrity test is performed automatically when the module is powered on, before the module transitions to the operational state.

If the software integrity test fails, the module transitions to the error state. The HMAC and SHA2-256 algorithms go through their respective CASTs before the software integrity test is performed.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB	Decrypt with 128-bit key	KAT	CAST	Module becomes operational and services are available for use	Symmetric operation	Test runs at power-on before the integrity test
AES-GCM Encryption	Encrypt with 256-bit key	KAT	CAST	Module becomes operational and services are available for use	Symmetric operation	Test runs at power-on before the integrity test
AES-GCM Decryption	Decrypt with 256-bit key	KAT	CAST	Module becomes operational and services are available for use	Symmetric operation	Test runs at power-on before the integrity test
SHA2-512	3-byte message	KAT	CAST	Module becomes operational and services are available for use	Message digest	Test runs at power-on before the integrity test
SHA3-256	4-byte message	KAT	CAST	Module becomes operational and services are available for use	Message digest	Test runs at power-on before the integrity test
Counter DRBG	128 bit keys, DF, with PR	KAT	CAST	Module becomes operational and services are available for use	Compliant with SP 800-90Ar1 including health test per section 11.3: performs instantiation, reseeding, and generation.	Test runs at power-on before the integrity test
HMAC DRBG	HMAC-SHA-256, with PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1 including health test per	Test runs at power-on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
				and services are available for use	section 11.3: performs instantiation, reseeding, and generation.	before the integrity test
KAS-ECC-SSC Sp800-56Ar3	P-256 curve	KAT	CAST	Module becomes operational and services are available for use	Shared secret computation	Test runs at power-on before the integrity test
KAS-FFC-SSC Sp800-56Ar3	ffdhe2048	KAT	CAST	Module becomes operational and services are available for use	Shared secret computation	Test runs at power-on before the integrity test
KDF SP800-108	HMAC-SHA2-256 with 128-bit key; KMAC-128 with 128-bit key	KAT	CAST	Module becomes operational and services are available for use	Key based key derivation	Test runs at power-on before the integrity test
KDA OneStep SP800-56Cr2	SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Shared secret key derivation	Test runs at power-on before the integrity test
KDF ANS 9.42	SHA-256	KAT	CAST	Module becomes operational and services are available for use	Industry-based ANS X9.42 key derivation	Test runs at power-on before the integrity test
KDF ANS 9.63	SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Industry-based ANS X9.63 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	SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Industry-based TLS v1.2 KDF key derivation	Test runs at power-on before the integrity test
TLS v1.3 KDF	SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Industry-based TLS v1.3 KDF key derivation	Test runs at power-on before the integrity test
PBKDF	HMAC-SHA2-256 with 200-bit derived key length, 24-character password, 2 iterations, 288-bit salt	KAT	CAST	Module becomes operational and services are available for use	Password-based key derivation	Test runs at power-on before the integrity test
KDA HKDF SP800-56Cr2	SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Shared secret key derivation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-5)	P-224 with SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-5)	P-224 with SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Digital signature verification	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-5)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Digital signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
EDDSA SigGen (FIPS186-5)	Ed25519; Ed4488	KAT	CAST	Module becomes operational and services are available for use	Digital signature generation	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-5)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Digital signature verification	Test runs at power-on before the integrity test
EDDSA SigVer (FIPS186-5)	Ed25519; Ed4488	KAT	CAST	Module becomes operational and services are available for use	Digital signature verification	Test runs at power-on before the integrity test
ECDSA KeyGen (FIPS186-5)	SHA2-512	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
EDDSA KeyGen (FIPS186-5)	N/A	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA KeyGen (FIPS186-5)	Section 6.4.1.1 of SP800-56Br2	PCT	PCT	Successful key pair generation	Encryption & decryption	Key pair generation
Safe Primes Key Generation	Section 5.6.2.1.4 of SP800-56Arev3	PCT	PCT	Successful key pair generation	SP 800-56Arev3, 5.6.2.1.4	Key pair generation
Hash DRBG	SHA2-256, with PR	KAT	CAST	Module becomes operational and services are available for use	Compliant with SP 800-90Ar1 including health test per section 11.3: performs instantiation, reseeding, and generation.	Test runs at power-on before the integrity test
ML-KEM KeyGen PCT	N/A	PCT	PCT	Successful key pair generation	Key encapsulation & decapsulation	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ML-KEM Encapsulation	ML-KEM-512, encap KAT	KAT	CAST	Module becomes operational and services are available for use	Key encapsulation	Test runs at power-on before the integrity test
ML-KEM Decapsulation	ML-KEM-512, decap KAT: implicit rejection path and non-rejection path	KAT	CAST	Module becomes operational and services are available for use	Key decapsulation	Test runs at power-on before the integrity test
ML-DSA KeyGen PCT	N/A	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
ML-DSA SigGen	ML-DSA-65 (internal interface, covering all rejection paths for ML-DSA-65 per Note 23 of IG 10.3.A)	KAT	CAST	Module becomes operational and services are available for use	Signature generation	Test runs at power-on before the integrity test
ML-DSA SigVer	ML-DSA-65 (internal interface)	KAT	CAST	Module becomes operational and services are available for use	Signature verification	Test runs at power-on before the integrity test
SLH-DSA KeyGen PCT	N/A	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
SLH-DSA SigGen with SHA2	SLH-DSA-SHA2-128f, external signature interface	KAT	CAST	Module becomes operational and services are available for use	Signature generation	Test runs at power-on before the integrity test
SLH-DSA SigGen with SHAKE	SLH-DSA-SHAKE-128f, external signature interface	KAT	CAST	Module becomes operational and services are available for use	Signature generation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SLH-DSA SigVer with SHA2	SLH-DSA-SHA2-128f, external signature interface	KAT	CAST	Module becomes operational and services are available for use	Signature verification	Test runs at power-on before the integrity test
SLH-DSA SigVer with SHAKE	SLH-DSA-SHAKE-128f, external signature interface	KAT	CAST	Module becomes operational and services are available for use	Signature verification	Test runs at power-on before the integrity test
Deterministic ECDSA SigGen (FIPS186-5)	P-224, SHA2-256	KAT	CAST	Module becomes operational and services are available for use	Signature generation	Test runs at power-on before the integrity test
ML-KEM KeyGen KAT	ML-KEM-512	KAT	CAST	Module becomes operational and services are available for use	Key generation	Test runs at power-on before the integrity test
ML-DSA KeyGen KAT	ML-DSA-65	KAT	CAST	Module becomes operational and services are available for use	Key generation	Test runs at power-on before the integrity test
SLH-DSA KeyGen KAT	SLH-DSA-SHA2-128f	KAT	CAST	Module becomes operational and services are available for use	Key generation	Test runs at power-on before the integrity test
Entropy Source - RCT and APT start-up test	1024 samples. Repetition count test according to Section 4.4.1 and Adaptive proportion test according to Section 4.4.2 of SP 800-90B	RCT and APT	CAST	Module becomes operational and services are available for use	Entropy source start-up test	Entropy source initialization

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Entropy Source - RCT and APT continuous test	Repetition count test according to Section 4.4.1 and Adaptive proportion test according to Section 4.4.2 of SP 800-90B	RCT and APT	CAST	Entropy source is operational	Entropy source continuous test	Continuously when the entropy source is accessed

Table 23: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A7583)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7584)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7591)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7604)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7606)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7608)	Message Authentication	SW/FW Integrity	On demand	Manually
HMAC-SHA2-256 (A7614)	Message Authentication	SW/FW Integrity	On demand	Manually

Table 24: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-ECB	KAT	CAST	On Demand	Manually
AES-GCM Encryption	KAT	CAST	On Demand	Manually
AES-GCM Decryption	KAT	CAST	On Demand	Manually
SHA2-512	KAT	CAST	On Demand	Manually
SHA3-256	KAT	CAST	On Demand	Manually
Counter DRBG	KAT	CAST	On Demand	Manually
HMAC DRBG	KAT	CAST	On Demand	Manually
KAS-ECC-SSC Sp800-56Ar3	KAT	CAST	On Demand	Manually
KAS-FFC-SSC Sp800-56Ar3	KAT	CAST	On Demand	Manually
KDF SP800-108	KAT	CAST	On Demand	Manually
KDA OneStep SP800-56Cr2	KAT	CAST	On Demand	Manually
KDF ANS 9.42	KAT	CAST	On Demand	Manually
KDF ANS 9.63	KAT	CAST	On Demand	Manually
TLS v1.2 KDF RFC7627	KAT	CAST	On Demand	Manually
TLS v1.3 KDF	KAT	CAST	On Demand	Manually
PBKDF	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KDA HKDF SP800-56Cr2	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-5)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-5)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-5)	KAT	CAST	On Demand	Manually
EDDSA SigGen (FIPS186-5)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-5)	KAT	CAST	On Demand	Manually
EDDSA SigVer (FIPS186-5)	KAT	CAST	On Demand	Manually
ECDSA KeyGen (FIPS186-5)	PCT	PCT	On Demand	Manually
EDDSA KeyGen (FIPS186-5)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-5)	PCT	PCT	On Demand	Manually
Safe Primes Key Generation	PCT	PCT	On Demand	Manually
Hash DRBG	KAT	CAST	On Demand	Manually
ML-KEM KeyGen PCT	PCT	PCT	On Demand	Manually
ML-KEM Encapsulation	KAT	CAST	On Demand	Manually
ML-KEM Decapsulation	KAT	CAST	On Demand	Manually
ML-DSA KeyGen PCT	PCT	PCT	On Demand	Manually
ML-DSA SigGen	KAT	CAST	On Demand	Manually
ML-DSA SigVer	KAT	CAST	On Demand	Manually
SLH-DSA KeyGen PCT	PCT	PCT	On Demand	Manually
SLH-DSA SigGen with SHA2	KAT	CAST	On Demand	Manually
SLH-DSA SigGen with SHAKE	KAT	CAST	On Demand	Manually
SLH-DSA SigVer with SHA2	KAT	CAST	On Demand	Manually
SLH-DSA SigVer with SHAKE	KAT	CAST	On Demand	Manually
Deterministic ECDSA SigGen (FIPS186-5)	KAT	CAST	On Demand	Manually
ML-KEM KeyGen KAT	KAT	CAST	On Demand	Manually
ML-DSA KeyGen KAT	KAT	CAST	On Demand	Manually
SLH-DSA KeyGen KAT	KAT	CAST	On Demand	Manually
Entropy Source - RCT and APT start-up test	RCT and APT	CAST	On Demand	Manually
Entropy Source - RCT and APT continuous test	RCT and APT	CAST	On Demand	Manually

Table 25: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Power-up error	An error occurred during the self-tests executed on power-up	Software integrity test failure	Re-initialization of the module	Module will not load

Name	Description	Conditions	Recovery Method	Indicator
		CAST failure		
Run-time error	An error occurred during a PCT or an entropy source health test failure occurred	PCT failure or entropy source health test failure	Re-initialization of the module	Cryptographic functionality and data output is blocked

Table 26: Error States

In any error state, the output interface is inhibited, and the module cannot perform cryptographic operations.

10.5 Operator Initiation of Self-Tests

The software integrity test and cryptographic algorithm self-tests can be invoked on demand by unloading and subsequently re-initializing the module. The PCTs can be invoked on demand by requesting the key pair generation service.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

No installation, initialization, or startup steps are required as the module is pre-built into the images that the vendor provides.

11.2 Administrator Guidance

To verify that the module is prebuilt into a Chainguard image, the Crypto Officer must review the image specification (e.g. Chainguard Console SBOM tab, SPDX image attestation, apk installed package database, syft output, AWS Inspector) to ensure it contains the package “openssl-fips-provider-3.6.0” version “3.6.0-r4”.

The Crypto Officer must verify the name and version of the module. This is done by retrieving the parameters `OSSL_PROV_PARAM_NAME` and `OSSL_PROV_PARAM_BUILDINFO` from the module. Since there can be several providers for OpenSSL, and since only the “fips” provider is the module, the Crypto Officer must ensure that the “fips” provider of OpenSSL is queried when retrieving these parameters and when requesting any other service.

`OSSL_PROV_PARAM_NAME` must have the value: “Chainguard FIPS Provider for OpenSSL”.

`OSSL_PROV_PARAM_BUILDINFO` must have the value: “3.6.0-r4”

The Approved and non-Approved modes of operation are specified in section 2.4. The administrative functions are specified in the Approved Services table. All the logical interfaces are specified in section 3.1. The requirements and restrictions that shall be considered when operating the module in approved mode are specified in section 2.7 and section 6.

11.3 Non-Administrator Guidance

There is no non-administrator guidance.

11.4 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 package can be uninstalled from the system.

12 Mitigation of Other Attacks

12.1 Attack List

Certain cryptographic subroutines and algorithms are vulnerable to timing analysis. The module mitigates this vulnerability by using constant-time implementations. This includes, but is not limited to:

- Big number operations: computing GCDs, modular inversion, multiplication, division, and modular exponentiation (using Montgomery multiplication).
- Elliptic curve point arithmetic: addition and multiplication (using the Montgomery ladder).
- Vector-based AES implementations.

12.2 Mitigation Effectiveness

RSA, ECDSA, ECDH, and DH employ blinding techniques to further impede timing and power analysis.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
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
CPACF	CP Assist for Cryptographic Functions
CSP	Critical Security Parameter
CTR	Counter
CTS	Ciphertext Stealing
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
KBKDF	Key-based Key Derivation Function
KMAC	KECCAK Message Authentication Code
KW	Key Wrap
KWP	Key Wrap with Padding
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OAEP	Optimal Asymmetric Encryption Padding
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PCT	Pair-wise Consistency Test
PBKDF2	Password-based Key Derivation Function v2
PKCS	Public-Key Cryptography Standards
PSS	Probabilistic Signature Scheme
RSADP	RSA Decryption Primitive
RSAEP	RSA Encryption Primitive
RSA	Rivest, Shamir, Addleman
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 B. References

- ANS X9.42-2001** **Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography**
2001
<https://webstore.ansi.org/standards/ascx9/ansix9422001>
- ANS X9.63-2001** **Public Key Cryptography for the Financial Services Industry, Key Agreement and Key Transport Using Elliptic Curve Cryptography**
2001
<https://webstore.ansi.org/standards/ascx9/ansix9632001>
- 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 [09-02-2025]** **Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program**
<https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-ig-announcements>
- FIPS 180-4** **Secure Hash Standard (SHS)**
March 2012
<https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf>
- FIPS 186-5** **Digital Signature Standard (DSS)**
February 2023
<https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-5.pdf>
- FIPS 197** **Advanced Encryption Standard**
November 2001
<https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>
- FIPS 198-1** **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf
- FIPS 202** **SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions**
August 2015
<https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf>
- PKCS#1** **Public Key Cryptography Standards (PKCS) #1: RSA Cryptography**
Specifications Version 2.1
February 2003
<http://www.ietf.org/rfc/rfc3447.txt>
- RFC 3526** **More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)**
May 2003
<https://www.ietf.org/rfc/rfc3526.txt>

RFC 5288	AES Galois Counter Mode (GCM) Cipher Suites for TLS August 2008 https://www.ietf.org/rfc/rfc5288.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://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a-add.pdf
SP 800-38A Addendum	Recommendation for Block Cipher Modes of Operation: Three Variants of Ciphertext Stealing for CBC Mode October 2010 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a-add.pdf
SP 800-38B	Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication May 2005 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38B.pdf
SP 800-38C	Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality May 2004 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf
SP 800-38D	Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC November 2007 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf
SP 800-38E	Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38e.pdf
SP 800-38F	Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping December 2012 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf
SP 800-52r2	Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations August 2019 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r2.pdf

SP 800-56Ar3	Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography April 2018 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Ar3.pdf
SP 800-56Cr1	Recommendation for Key-Derivation Methods in Key-Establishment Schemes August 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Cr1.pdf
SP 800-56Cr2	Recommendation for Key-Derivation Methods in Key-Establishment Schemes August 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-56Cr2.pdf
SP 800-90Ar1	Recommendation for Random Number Generation Using Deterministic Random Bit Generators June 2015 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
SP 800-90B	Recommendation for the Entropy Sources Used for Random Bit Generation January 2018 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf
SP 800-108r1	NIST Special Publication 800-108 - Recommendation for Key Derivation Using Pseudorandom Functions August 2022 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-108r1.pdf
SP 800-132	Recommendation for Password-Based Key Derivation - Part 1: Storage Applications December 2010 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-132.pdf
SP 800-133r2	Recommendation for Cryptographic Key Generation June 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf
SP 800-135r1	Recommendation for Existing Application-Specific Key Derivation Functions December 2011 https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-135r1.pdf
SP 800-140Br1	CMVP Security Policy Requirements November 2023 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140Br1.pdf