Canonical Ltd.

Canonical Ltd. Ubuntu 22.04 Libgcrypt Cryptographic Module

version 1.9.4-3ubuntu3+Fips1.2

FIPS 140-3 Non-Proprietary Security Policy

Document version 1.1

Last updated: 2024-08-30

Prepared by:

atsec information security corporation

4516 Seton Center Parkway, Suite 250

Austin, TX 78759

www.atsec.com

Prepared for:

Canonical Ltd.

110 Southwark Street, Blue Fin Building, 5th Floor

London, SE1 0SU

www.canonical.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for version 1.9.4-3ubuntu3+Fips1.2 of the Canonical Ltd. Ubuntu 22.04 Libgcrypt Cryptographic Module. It has a one-to-one mapping to the [SP 800-140B] standard, starting with Section B.2.1 named “General” that maps to Section 1 in this document and ending with Section B.2.12 named “Mitigation of Other Attacks” that maps to Section 12 in this document.

1.2 Security Levels

Section	Security Level
1	1
2	1
3	1
4	1
5	1
6	1
7	N/A
8	N/A
9	1
10	1
11	1
12	1
	1

Table 1: Security Levels

1.3 Additional Information [O]

Not applicable.

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The module is software library implementing general purpose cryptographic algorithms. The module provides cryptographic services to applications running in the user space of the underlying operating system through an application program interface (API).

The module is implemented as a set of shared library / binary file; as shown in Figure 1. The shared library file constitutes the cryptographic boundary.

Module Type: Software

Module Embodiment: Multi-Chip standalone

Module Characteristics:

Cryptographic Boundary:

The Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets) table in Section 2.2 lists the software components of the cryptographic module, which defines its cryptographic boundary.

Tested Operational Environment's Physical Perimeter (TOEPP) [O]:

The physical perimeter is comprised in a General Purpose Computer.

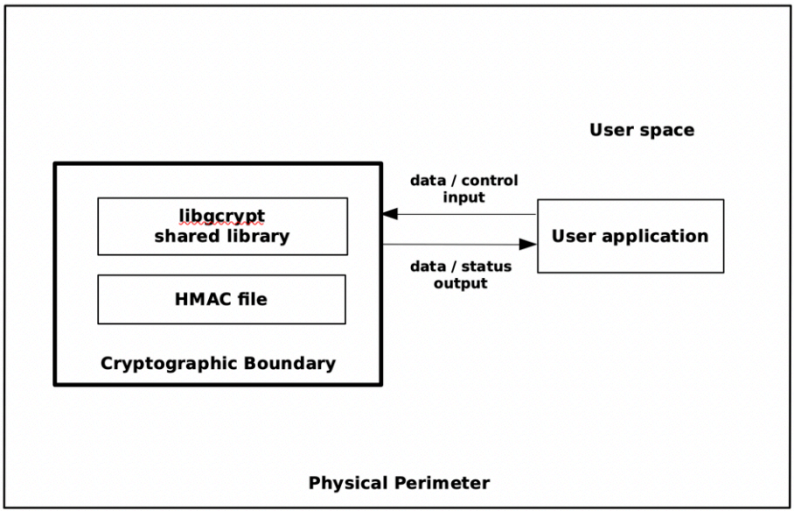

Figure 1: Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
libgcrypt.so.20.3.4	1.9.4-3ubuntu3+Fips1.2	N/A	HMAC-SHA2-256
.libgcrypt.so.20.hmac	1.9.4-3ubuntu3+Fips1.2	N/A	HMAC-SHA2-256

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

Tested Module Identification – Hybrid Disjoint Hardware:

N/A for this module.

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Ubuntu 22.04 LTS 64-bit	Supermicro SYS-1019P-WTR	Intel Xeon Gold 6226	AESNI, SSSE3, SHLD	N/A	1.9.4-3ubuntu3+Fips1.2
Ubuntu 22.04 LTS 64-bit	Amazon Web Services (AWS) c6g.metal	AWS Graviton2	NEON	N/A	1.9.4-3ubuntu3+Fips1.2
Ubuntu 22.04 LTS 64-bit	IBM z15	IBM z15	CPACF	N/A	1.9.4-3ubuntu3+Fips1.2

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Ubuntu 22.04 LTS 64-bit	Supermicro SYS-1019P-WTR	Intel Xeon Gold 6226	No	N/A	1.9.4-3ubuntu3+Fips1.2
Ubuntu 22.04 LTS 64-bit	Amazon Web Services (AWS) c6g.metal	AWS Graviton2	No	N/A	1.9.4-3ubuntu3+Fips1.2
Ubuntu 22.04 LTS 64-bit	IBM z15	IBM z15	No	N/A	1.9.4-3ubuntu3+Fips1.2

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

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

Operating System	Hardware Platform
Ubuntu Core 22	Supermicro SYS-1019P-WTR
Ubuntu Core 22	Amazon Web Services (AWS) c6g.metal

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

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

2.3 Excluded Components

There are no excluded components from the module.

2.4 Modes of Operation

Modes List and Description:

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

Table 5: Modes List and Description

Mode Change Instructions and Status [O]:

When the module starts up successfully, after passing all the pre-operational self-test, the module is operating in the approved mode of operation by default and can only be transitioned into the non-approved mode by calling one of the non-approved services listed in the Non-Approved Services table. See Section 4 for the details on service indicator provided by the module that identifies when an approved service is called.

Degraded Mode Description [O]:

There is no degraded mode of operation implemented within the module.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A3699	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
SHA-1	A3699	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
AES-CBC	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A3700	Key Length - 128, 192, 256	SP 800-38C

Algorithm	CAVP Cert	Properties	Reference
AES-CFB128	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A3700	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-OFB	A3700	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A3700	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E
Counter DRBG	A3700	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A3700	Secret Generation Mode: Testing Candidate Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3700	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3700	Curves: P-224, P-256, P-384, P-521	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigVer (FIPS186-4)	A3700	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A3700	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC DRBG	A3700	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-224	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-256	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA3-512	A3700	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
PBKDF	A3700	Password; derived key with 112-512 bits key size and 112-256 bits key strength	SP 800-132
RSA KeyGen (FIPS186-4)	A3700	Key Generation Mode: B.3.3 Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigGen (FIPS186-4)	A3700	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A3700	Private Key Format: standard Public Exponent Mode: random	FIPS 186-4
RSA SigVer (FIPS186-4)	A3700	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA-1	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-384	A3700	- Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/224	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA3-224	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-256	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-384	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-512	A3700	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202

Algorithm	CAVP Cert	Properties	Reference
SHAKE-128	A3700	Message Length - Message Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A3700	Message Length - Message Length: 16-65536 Increment 8	FIPS 202
AES-CBC	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A3701	Key Length - 128, 192, 256	SP 800-38C
AES-CFB128	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A3701	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-OFB	A3701	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A3701	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E
Counter DRBG	A3701	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyGen (FIPS186-4)	A3701	Secret Generation Mode: Testing Candidate Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3701	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3701	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3701	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A3701	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC DRBG	A3701	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-512/256	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-224	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-256	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-512	A3701	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
PBKDF	A3701	Password; derived key with 112-512 bits key size and 112-256 bits key strength	SP 800-132
RSA KeyGen (FIPS186-4)	A3701	Key Generation Mode: B.3.3 Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigGen (FIPS186-4)	A3701	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A3701	Private Key Format: standard Public Exponent Mode: random	FIPS 186-4
RSA SigVer (FIPS186-4)	A3701	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
SHA-1	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A3701	- Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/224	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA3-224	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202

Algorithm	CAVP Cert	Properties	Reference
SHA3-256	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-384	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-512	A3701	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHAKE-128	A3701	Message Length - Message Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A3701	Message Length - Message Length: 16-65536 Increment 8	FIPS 202
HMAC-SHA-1	A3702	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
SHA-1	A3702	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
AES-CBC	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A3703	Key Length - 128, 192, 256	SP 800-38C
AES-CFB128	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Algorithm	CAVP Cert	Properties	Reference
AES-CMAC	A3703	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-OFB	A3703	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A3703	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E
Counter DRBG	A3703	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A3703	Secret Generation Mode: Testing Candidate Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3703	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3703	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3703	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A3703	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1

Algorithm	CAVP Cert	Properties	Reference
HMAC DRBG	A3703	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A3703	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
PBKDF	A3703	Password; derived key with 112-512 bits key size and 112-256 bits key strength	SP 800-132
RSA KeyGen (FIPS186-4)	A3703	Key Generation Mode: B.3.3 Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigGen (FIPS186-4)	A3703	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A3703	Private Key Format: standard	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
		Public Exponent Mode: random	
RSA SigVer (FIPS186-4)	A3703	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA-1	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/224	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A3703	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CCM	A3704	Key Length - 128, 192, 256	SP 800-38C
AES-CFB128	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB8	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A3704	Direction - Generation, Verification Key Length - 128, 192, 256	SP 800-38B
AES-CTR	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-KW	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38F
AES-OFB	A3704	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-XTS Testing Revision 2.0	A3704	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38E
Counter DRBG	A3704	Prediction Resistance - No, Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - Yes	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A3704	Secret Generation Mode: Testing Candidate Curves: P-224, P-256, P-384, P-521	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
ECDSA KeyVer (FIPS186-4)	A3704	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3704	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3704	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A3704	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC DRBG	A3704	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-224	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A3704	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
PBKDF	A3704	Password; derived key with 112-512 bits key size and 112-256 bits key strength	SP 800-132
RSA KeyGen (FIPS186-4)	A3704	Key Generation Mode: B.3.3 Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigGen (FIPS186-4)	A3704	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A3704	Private Key Format: standard Public Exponent Mode: random	FIPS 186-4
RSA SigVer (FIPS186-4)	A3704	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA-1	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A3704	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
		Large Message Sizes - 1, 2, 4, 8	
SHA2-512	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/224	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A3704	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
ECDSA KeyGen (FIPS186-4)	A3705	Secret Generation Mode: Testing Candidate Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A3705	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A3705	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A3705	Curves: P-224, P-256, P-384, P-521	FIPS 186-4
Hash DRBG	A3705	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC DRBG	A3705	Prediction Resistance - No, Yes Mode - SHA-1, SHA2-256, SHA2-512	SP 800-90A Rev. 1
HMAC-SHA-1	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-224	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-224	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-256	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
HMAC-SHA3-512	A3705	Key Length - Key Length: 112-524288 Increment 8	FIPS 198-1
PBKDF	A3705	Password; derived key with 112-512 bits key size and 112-256 bits key strength	SP 800-132
RSA KeyGen (FIPS186-4)	A3705	Key Generation Mode: B.3.3 Modulo - 2048, 3072, 4096	FIPS 186-4

Algorithm	CAVP Cert	Properties	Reference
RSA SigGen (FIPS186-4)	A3705	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA Signature Primitive (CVL)	A3705	Private Key Format: standard Public Exponent Mode: random	FIPS 186-4
RSA SigVer (FIPS186-4)	A3705	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Signature Type - PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
SHA-1	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-224	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-256	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-384	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4

Algorithm	CAVP Cert	Properties	Reference
SHA2-512/224	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA2-512/256	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 180-4
SHA3-224	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-256	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-384	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHA3-512	A3705	Message Length - Message Length: 0-65536 Increment 8 Large Message Sizes - 1, 2, 4, 8	FIPS 202
SHAKE-128	A3705	Message Length - Message Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A3705	Message Length - Message Length: 16-65536 Increment 8	FIPS 202

Table 6: Approved Algorithms

The table above lists all implemented modes of operation for every security function employed for approved services by the module.

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
Cooperative Key Generation (CKG)	Key Type:Asymmetric RSA:2048, 3072, 4096 bits (112, 128, 149 bits) ECDSA:P-224, P-256, P-384, P-521 (112, 128, 192, 256 bits)	N/A	FIPS 186-4, SP800-133r2 Section 4 example 1

Table 7: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

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

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

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

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
MD5	Message digest
ECDH	Shared secret computation
RSA	Encryption primitives; Decryption primitives; Signature verification primitives
RSA with non-approved public key flags	Key generation; Signature generation; Signature verification
ECDSA	Signature generation primitives
ECDSA with non-approved public key flags	Key generation; Signature generation; Signature verification

Name	Use and Function
AES-GCM	Authenticated symmetric encryption; Authenticated symmetric decryption
AES-OCB	Authenticated symmetric encryption; Authenticated symmetric decryption
AES-EAX	Authenticated symmetric encryption; Authenticated symmetric decryption
AES-SIV	Authenticated symmetric encryption; Authenticated symmetric decryption

Table 8: Non-Approved, Not Allowed Algorithms

The table above lists non-approved algorithms that are not allowed in the approved mode of operation. These algorithms are used by the non-approved services listed in the Non-Approved Services table.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Message authentication code	MAC	Message authentication code	AES-CMAC:128, 192, 256-bit keys with 128-256 bits key strength HMAC-SHA-1:112-524288 bit key size with 112 -256 bits key strength HMAC-SHA2-224:112-524288 bit key size with 112-256 bits key strength HMAC-SHA2-256:112-524288 bit key size with 112-256 bits key strength	AES-CMAC AES-CMAC AES-CMAC AES-CMAC AES-CMAC AES-CMAC HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA-1 HMAC-SHA2-224 HMAC-SHA2-224 HMAC-SHA2-224

Name	Type	Description	Properties	Algorithms
			HMAC-SHA2-384:112-524288 bit key size with 112-256 bits key strength HMAC-SHA2-512:112-524288 bit key size with 112-256 bits key strength HMAC-SHA2-512/224:112-524288 bit key size with 112-256 bits key strength HMAC-SHA2-512/256:112-524288 bit key size with 112-256 bits key strength HMAC-SHA3-224:112-524288 bit key size with 112-256 bits key strength HMAC-SHA3-256:112-524288 bit key size with 112-256 bits key strength HMAC-SHA3-384:112-524288 bit key size with 112-256 bits key strength HMAC-SHA3-512:112-524288 bit key size with 112-256 bits key strength	HMAC-SHA2-224 HMAC-SHA2-224 HMAC-SHA2-224 HMAC-SHA2-256 HMAC-SHA2-256 HMAC-SHA2-256 HMAC-SHA2-256 HMAC-SHA2-256 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-384 HMAC-SHA2-384 HMAC-SHA2-384 HMAC-SHA2-384 HMAC-SHA2-384 HMAC-SHA2-512 HMAC-SHA2-512 HMAC-SHA2-512 HMAC-SHA2-512 HMAC-SHA2-512 HMAC-SHA2-

Name	Type	Description	Properties	Algorithms
				512/224 HMAC-SHA2-512/224 HMAC-SHA2-512/224 HMAC-SHA2-512/224 HMAC-SHA2-512/224 HMAC-SHA2-512/224 HMAC-SHA2-512/256 HMAC-SHA2-512/256 HMAC-SHA2-512/256 HMAC-SHA2-512/256 HMAC-SHA2-512/256 HMAC-SHA2-512/256 HMAC-SHA3-224 HMAC-SHA3-224 HMAC-SHA3-224 HMAC-SHA3-224 HMAC-SHA3-256 HMAC-SHA3-256 HMAC-SHA3-256 HMAC-SHA3-256 HMAC-SHA3-384 HMAC-SHA3-384

Name	Type	Description	Properties	Algorithms
				HMAC-SHA3-384 HMAC-SHA3-384 HMAC-SHA3-512 HMAC-SHA3-512 HMAC-SHA3-512 HMAC-SHA3-512
Message digest	SHA XOF	Message digest	SHA-1:N/A SHA2-224:N/A SHA2-256:N/A SHA2-384:N/A SHA2-512:N/A SHA2-512/224:N/A SHA2-512/256:N/A SHA3-224:N/A SHA3-256:N/A SHA3-384:N/A SHA3-512:N/A SHAKE-128:N/A SHAKE-256:N/A	SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 SHA-1 SHA2-224 SHA2-224 SHA2-224 SHA2-224 SHA2-224 SHA2-224 SHA2-256 SHA2-256 SHA2-256 SHA2-256 SHA2-256 SHA2-256 SHA2-384 SHA2-384 SHA2-384 SHA2-384 SHA2-384 SHA2-384 SHA2-512 SHA2-512 SHA2-512 SHA2-512 SHA2-512

Name	Type	Description	Properties	Algorithms
				SHA2-512 SHA2-512/224 SHA2-512/224 SHA2-512/224 SHA2-512/224 SHA2-512/224 SHA2-512/224 SHA2-512/256 SHA2-512/256 SHA2-512/256 SHA2-512/256 SHA2-512/256 SHA2-512/256 SHA3-224 SHA3-224 SHA3-224 SHA3-224 SHA3-256 SHA3-256 SHA3-256 SHA3-256 SHA3-384 SHA3-384 SHA3-384 SHA3-384 SHA3-512 SHA3-512 SHA3-512 SHA3-512 SHAKE-128 SHAKE-128 SHAKE-128 SHAKE-128 SHAKE-256 SHAKE-256 SHAKE-256 SHAKE-256
Symmetric encryption	BC-UnAuth	Symmetric encryption	AES-CBC:128, 192, 256-bit keys with 128-256 bits key strength AES-CFB128:128,	AES-CBC AES-CBC AES-CBC AES-CBC AES-CBC

Name	Type	Description	Properties	Algorithms
			192, 256-bit keys with 128-256 bits key strength AES-CFB8:128, 192, 256-bit keys with 128-256 bits key strength AES-CTR:128, 192, 256-bit keys with 128-256 bits key strength AES-ECB:128, 192, 256-bit keys with 128-256 bits key strength AES-OFB:128, 192, 256-bit keys with 128-256 bits key strength	AES-CBC AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CTR AES-CTR AES-CTR AES-CTR AES-CTR AES-CTR AES-ECB AES-ECB AES-ECB AES-ECB AES-ECB AES-ECB AES-ECB AES-OFB AES-OFB AES-OFB AES-OFB AES-OFB AES-OFB
Symmetric decryption	BC-UnAuth	Symmetric decryption	AES-CBC:128, 192, 256-bit keys with 128-256 bits key strength AES-CFB128:128, 192, 256-bit keys with 128-256 bits key strength AES-CFB8:128, 192, 256-bit keys with 128-256 bits	AES-CBC AES-CBC AES-CBC AES-CBC AES-CBC AES-CBC AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128 AES-CFB128

Name	Type	Description	Properties	Algorithms
			key strength AES-CTR:128, 192, 256-bit keys with 128-256 bits key strength AES-ECB:128, 192, 256-bit keys with 128-256 bits key strength AES-OFB:128, 192, 256-bit keys with 128-256 bits key strength	AES-CFB128 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CFB8 AES-CTR AES-CTR AES-CTR AES-CTR AES-CTR AES-CTR AES-ECB AES-ECB AES-ECB AES-ECB AES-ECB AES-ECB AES-OFB AES-OFB AES-OFB AES-OFB AES-OFB AES-OFB AES-OFB
Authenticated symmetric encryption	BC-Auth	Authenticated symmetric encryption	AES-CCM:128, 192, 256-bit keys with 128-256 bits key strength	AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Authenticated symmetric decryption	BC-Auth	Authenticated symmetric decryption	AES-CCM:128, 192, 256-bit keys with 128-256 bits key strength	AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Key wrapping	KTS-Wrap	SP800-38C and SP800-	AES-KW:128, 192, 256-bit keys with	AES-KW AES-KW

Name	Type	Description	Properties	Algorithms
		38F. KTS (Key wrapping) per IG D.G	128-256 bits key strength AES-CCM:128, 192, 256-bit keys with 128-256 bits key strength	AES-KW AES-KW AES-KW AES-KW AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Key unwrapping	KTS-Wrap	SP800-38C and SP800-38F. KTS (Key unwrapping) per IG D.G	AES-KW:128, 192, 256-bit keys with 128-256 bits key strength AES-CCM:128, 192, 256-bit keys with 128-256 bits key strength	AES-KW AES-KW AES-KW AES-KW AES-KW AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM AES-CCM
Symmetric encryption (for data storage)	BC-UnAuth	Symmetric encryption (for data storage)	AES-XTS Testing Revision 2.0:128, 256-bit keys with 128-256 bits key strength	AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0

Name	Type	Description	Properties	Algorithms
Symmetric decryption (for data storage)	BC-UnAuth	Symmetric decryption (for data storage)	AES-XTS Testing Revision 2.0:128, 256-bit keys with 128-256 bits key strength	AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0 AES-XTS Testing Revision 2.0
Random number generation	DRBG	Random number generation	Counter DRBG:128, 192, 256-bit keys with 128-256 bits key strength HMAC DRBG:112-256 bits key strength Hash DRBG:N/A	Counter DRBG Counter DRBG Counter DRBG Counter DRBG Counter DRBG Counter DRBG HMAC DRBG HMAC DRBG HMAC DRBG HMAC DRBG HMAC DRBG HMAC DRBG Hash DRBG Hash DRBG Hash DRBG Hash DRBG

Name	Type	Description	Properties	Algorithms
				Hash DRBG Hash DRBG
Key pair generation	AsymKeyPair-KeyGen CKG	Key pair generation	ECDSA KeyGen (FIPS186-4):P-224, P-256, P-384, P-521 with 112-256 bits strength RSA KeyGen (FIPS186-4):2048, 3072, 4096 with 112-149 bits strength	ECDSA KeyGen (FIPS186-4) ECDSA KeyGen (FIPS186-4) ECDSA KeyGen (FIPS186-4) ECDSA KeyGen (FIPS186-4) ECDSA KeyGen (FIPS186-4) ECDSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4) RSA KeyGen (FIPS186-4)
Public key verification	AsymKeyPair-KeyVer	Public key verification	ECDSA KeyVer (FIPS186-4):P-224, P-256, P-384, P-521 with 112-256 bits strength	ECDSA KeyVer (FIPS186-4) ECDSA KeyVer (FIPS186-4) ECDSA KeyVer

Name	Type	Description	Properties	Algorithms
				(FIPS186-4) ECDSA KeyVer (FIPS186-4) ECDSA KeyVer (FIPS186-4) ECDSA KeyVer (FIPS186-4)
Digital signature generation	DigSig-SigGen	Digital signature generation	ECDSA SigGen (FIPS186-4):P-224, P-256, P-384, P-521 with 112-256 bits strength RSA SigGen (FIPS186-4):2048, 3072, 4096 with 112-149 bits strength	ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) ECDSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4) RSA SigGen (FIPS186-4)
Digital signature verification	DigSig-SigVer	Digital signature verification	ECDSA SigVer (FIPS186-4):P-224, P-256, P-384, P-521 with 112-256 bits strength	ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4)

Name	Type	Description	Properties	Algorithms
			RSA SigVer (FIPS186-4):2048, 3072, 4096 with 112-149 bits strength	ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) ECDSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4) RSA SigVer (FIPS186-4)
Key derivation	PBKDF	Key derivation	PBKDF:Password; derived key with 112-512 bits key size and 112-256 bits key strength	PBKDF PBKDF PBKDF PBKDF PBKDF PBKDF
Digital signature generation primitive	DigSig-SigGen	Digital signature generation primitive	RSA Signature Primitive:2048 with 112 bits strength	RSA Signature Primitive RSA Signature Primitive RSA Signature Primitive RSA Signature Primitive RSA Signature Primitive RSA Signature Primitive

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 AES XTS

The AES algorithm in XTS mode can be only used for the cryptographic protection of data on storage devices, as specified in [SP800-38E]. The AES-XTS shall not be used for other purposes, such as the encryption of data in transit. In addition, the length of a single data unit encrypted with the XTS-AES shall not exceed 2^{20} AES blocks, that is 16MB of data.

To meet the requirement stated in IG C.I, the module implements a check that ensures, before performing any cryptographic operation, that the two AES keys used in AES XTS mode are not identical.

2.7.2 Key Derivation using SP800-132 PBKDF

The module provides password-based key derivation (PBKDF), compliant with SP800-132. The module supports option 1a from Section 5.4 of [SP800-132], in which the Master Key (MK) or a segment of it is used directly as the Data Protection Key (DPK).

In accordance with [SP800-132] and FIPS 140-3 IG D.N, the following requirements shall be met.

- Derived keys shall only be used in storage applications. The Master Key (MK) shall not be used for other purposes. The module accepts a minimum length of 112 bits for the MK or DPK.
- A portion of the salt, with a length of at least 128 bits, shall be generated randomly using the SP800-90A DRBG.
- The iteration count shall be selected as large as possible, as long as the time required to generate the key using the entered password is acceptable for the users. The module only allows minimum iteration count to be 1000.
- Passwords or passphrases, used as an input for the PBKDF, shall not be used as cryptographic keys.
- The minimum length of the password or passphrase accepted by the module is 14 characters. The probability of guessing the value, assuming a worst-case scenario of all digits, is estimated to be at most 10^{-14}. Combined with the minimum iteration count, this provides an acceptable trade-off between user experience and security against brute-force attacks.

The calling application shall also observe the rest of the requirements and recommendations specified in [SP800-132].

2.8 RBG and Entropy

Cert Number	Vendor Name
E60	Canonical Ltd.

Table 10: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Userspace CPU Time Jitter RNG Entropy Source (version 2.2.0)	Non-Physical	Ubuntu 22.04 LTS 64-bit on Supermicro SYS-1019P-WTR with Intel Xeon Gold 6226; Ubuntu 22.04 LTS 64-bit on Amazon Web Services (AWS) c6g.metal with AWS Graviton2; Ubuntu 22.04 LTS 64-bit on IBM z15 with IBM z15	64	Full entropy	AES-256-CTR-DRBG (A3814)

Table 11: Entropy Sources

The module provides an SP800-90A-compliant Deterministic Random Bit Generator (DRBG) for creation of key components of asymmetric keys, and random number generation.

The seeding (and automatic reseeding) of the DRBG is done with getrandom().

The DRBG supports the Hash_DRBG, HMAC_DRBG and CTR_DRBG mechanisms. The DRBG is initialized during module initialization; the module loads by default the DRBG using the HMAC_DRBG mechanism with SHA-256 and without prediction resistance. A different DRBG mechanism can be chosen by invoking the gcry_control(GCRYCTL_DRBG_REINIT) function.

The module uses an [SP800-90B]-compliant entropy source specified in the table above. This entropy source is located within the module's physical perimeter, but outside of the module's cryptographic boundary. This is in compliance with IG 9.3.A Resolution 1(b). The module obtains 384 bits to seed the DRBG, and 256 bits to reseed it.

The module performs the DRBG health tests as defined in Section 11.3 of [SP800-90A].

2.9 Key Generation

The module provides an [SP800-90Arev1]-compliant Deterministic Random Bit Generator (DRBG) for the creation of key components of asymmetric keys and random number generation.

The Cryptographic Key Generation (CKG) methods implemented in the module for Approved services in approved mode are compliant with Section 4 example 1 of [SP800-133rev2].

For generating RSA and ECDSA keys, the module implements asymmetric key generation services compliant with [FIPS186-4]. A seed (i.e., the random value) used in asymmetric key generation is directly obtained from the [SP800-90Arev1] DRBG.

Additionally, the module implements the PBKDF2 key derivation method compliant with option 1a of SP 800-132. This implementation shall only be used to derive keys for use in storage applications.

2.10 Key Establishment

The module provides the following key transport mechanisms:

- AES Key wrapping using AES-KW.
- AES Key wrapping using AES-CCM.

According to Table 2: Comparable strengths in [SP 800-57rev5], the key sizes of AES provide the following security strength in approved mode of operation:

- AES key wrapping in KW mode provides between 128 and 256 bits of encryption strength.
- AES key wrapping using AES-CCM provides between 128 and 256 bits of encryption strength.

2.11 Industry Protocols

The module does not support any industry protocols listed within the publication of SP 800-135rev1. Therefore, this section is not applicable.

2.12 Additional Information [O]

Not applicable.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	API input parameters for data.
N/A	Data Output	API output parameters for data.
N/A	Control Input	API function calls, API input parameters for control input, /proc/sys/crypto/fips_enabled control file.
N/A	Status Output	API return codes, API output parameters for status output.

Table 12: Ports and Interfaces

As a software-only module, the module does not have physical ports. The operator can only interact with the module through the API provided by the module. Thus, the physical ports are interpreted to be the physical ports of the hardware platform on which the module runs.

All data output via data output interface is inhibited when the module is performing pre-operational test or zeroization or when the module enters error state.

3.2 Trusted Channel Specification [O]

Not applicable.

3.3 Control Interface Not Inhibited [O]

Not applicable.

3.4 Additional Information [O]

Not applicable.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

The module does not support authentication.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	CO	None

Table 13: Roles

The module supports the Crypto Office role only. This sole role is implicitly assumed by the operator of the module when performing a service.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Symmetric encryption	Perform AES encryption	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key, Plaintext	Ciphertext	Symmetric encryption Symmetric encryption (for data storage)	Crypto Officer - AES key: W,E
Symmetric	Perform AES	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key,	Plaintext	Symmetric decryption	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
decryption	decryption		Ciphertext		Symmetric decryption (for data storage)	- AES key: W,E
Authenticated symmetric encryption	Perform authenticated AES encryption	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key, IV, Plaintext	Ciphertext, MAC tag	Authenticated symmetric encryption	Crypto Officer - AES key: W,E
Authenticated symmetric decryption	Perform authenticated AES decryption	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key, Ciphertext, MAC tag	Plaintext or failure	Authenticated symmetric decryption	Crypto Officer - AES key: W,E
RSA key generation	Generate RSA key pairs	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) return GPG_ERR_NO_ERROR	Key size	RSA private key, RSA public key	Key pair generation	Crypto Officer - RSA private key: G,E - RSA public key: G,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
ECDSA key generation	Generate ECDSA key pairs	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) return GPG_ERR_NO_ERROR	Key size	ECDSA private key, ECDSA public key	Key pair generation	Crypto Officer - ECDSA private key: G,E - ECDSA public key: G,E
RSA Digital signature generation	RSA signature generation without pre-computed hash	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) and gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MD) return GPG_ERR_NO_ERROR	RSA private key, Message, Hash algorithm	Signature	Digital signature generation	Crypto Officer - RSA private key: W,E
RSA Digital signature generation (with pre-compu	RSA signature primitive	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) returns GPG_ERR_NO_ERROR	RSA private key, Message digest	Signature	Digital signature generation primitive	Crypto Officer - RSA private key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
ted hash)						
ECDSA Digital signature generation	ECDSA signature generation	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) and gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MD) return GPG_ERR_NO_ERROR	ECDSA private key, Message, Hash algorithm	Signature	Digital signature generation	Crypto Officer - ECDSA private key: W,E
RSA Digital signature verification	RSA signature verification	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) and gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MD) return GPG_ERR_NO_ERROR	RSA public key, Signature, Hash algorithm	Signature verification result (verified/fail)	Digital signature verification	Crypto Officer - RSA public key: W,E
ECDSA Digital signature verification	ECDSA signature verification	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS) and gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MD) return GPG_ERR_NO_ERROR	ECDSA public key, Signature, Hash algorithm	Signature verification result (verified/fail)	Digital signature verification	Crypto Officer - ECDSA public key: W,E
Public key verification	Verify ECDSA public key	gcry_mpi_ec_curve_point() returns GPG_ERR_NO_ERROR	ECDSA public key	Key verification result (verifi	Public key verification	Crypto Officer -

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
				ed/fail)		ECDSA public key: W,E
Random number generation	Generate random bitstrings	gcry_randomize(), gcry_random_bytes(), gcry_random_bytes_secure() return GPG_ERR_NO_ERROR			Random number generation	Crypto Officer - Entropy input: W,E - DRBG seed: G,W,E - DRBG internal state: (V value, C value): G,W,E - DRBG internal state: (V value, key): G,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Message digest	Compute SHA hashes	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MD) returns GPG_ERR_NO_ERROR	Message	Message digest	Message digest	Crypto Officer
Hash-based Message authentication code	Compute HMAC	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MAC) returns GPG_ERR_NO_ERROR	HMAC key, Message	MAC tag	Message authentication code	Crypto Officer - HMAC key: W,E
AES-based Message authentication code	Compute AES-based CMAC	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_MAC) returns GPG_ERR_NO_ERROR	AES key, Message	MAC tag	Message authentication code	Crypto Officer - AES key: W,E
Key wrapping	Perform AES-based key wrapping	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key, Key to be wrapped	Wrapped key	Key wrapping	Crypto Officer - AES key: W,E
Key unwrapping	Perform AES-based key unwrapping	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER) returns GPG_ERR_NO_ERROR	AES key, Key to be unwrapped	Unwrapped key	Key unwrapping	Crypto Officer - AES key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Key derivation	Perform key derivation	gcry_control(GCRYCTL_FIPS_SERVICE_INDICATOR_KDF) returns GPG_ERR_NO_ERROR	Password or passphrase	Derived key	Key derivation	Crypto Officer - Password or passphrase: W,E - Derived key: G,R
Show status	Show module status	N/A	None	Module status	None	Crypto Officer
Zeroization	Zeroize SSPs	N/A	None	None	None	Crypto Officer - AES key: Z - HMAC key: Z - RSA public key: Z - RSA private key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - ECDSA public key: Z - ECDSA private key: Z - Password or passphrase: Z - Derived key: Z - Entropy input: Z - DRBG internal state: (V value, C value): Z -

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						DRBG internal state: (V value, key): Z - DRBG seed: Z
Self-tests	Perform self-tests	N/A	None	Result of self-test (pass/fail)	None	Crypto Officer
Show module name and version	Show module name and version	N/A	None	Module's name and version	None	Crypto Officer

Table 14: Approved Services

The table above lists the approved services. For each service, the table lists the associated cryptographic algorithm(s), the role to perform the service, the cryptographic keys or CSPs involved, and their access type(s). The following convention is used to specify access rights to a CSP:

- **G** = **Generate**: The module generates or derives the SSP.
- **R** = **Read**: The SSP is read from the module (e.g., the SSP is output).
- **W** = **Write**: The SSP is updated, imported, or written to the module.
- **E** = **Execute**: The module uses the SSP in performing a cryptographic operation.
- **Z** = **Zeroize**: The module zeroizes the SSP.

- **N/A**: the calling application does not access any CSP or key during its operation.

The details of the approved cryptographic algorithms including the CAVP certificate numbers can be found in the Approved Services table. In order to check whether it utilizes an approved security function or not, the operator is responsible to invoke the gcry_control() API along with dedicated controls in the form of API input parameters.

The module implements the following controls depending on the requested service:

1. GCRYCTL_FIPS_SERVICE_INDICATOR_CIPHER - For symmetric algorithms.
2. GCRYCTL_FIPS_SERVICE_INDICATOR_KDF - For KDF functions.
3. GCRYCTL_FIPS_SERVICE_INDICATOR_PK_FLAGS - For asymmetric functions.[1]
4. GCRYCTL_FIPS_SERVICE_INDICATOR_MD - For digest functions.
5. GCRYCTL_FIPS_SERVICE_INDICATOR_MAC - For MAC functions.
6. GCRYCTL_FIPS_SERVICE_INDICATOR_FUNCTION - For non-approved public key functions.

In addition to that, for some of the below-mentioned services, the approved service indicator corresponds to the successful completion of the service. This affects the following APIs:

1. gcry_randomize, gcry_random_bytes, gcry_random_bytes_secure – API for *Random number generation* service.
2. gcry_mpi_ec_curve_point – API for *Public key verification* service.

For all approved services, GPG_ERR_NO_ERROR (i.e., “0”) return code indicates the service is approved. In case the dedicated functions are used in conjunction with the APIs representing the requested services, the operator is responsible to check that all of the called functions return GPG_ERR_NO_ERROR (i.e., “0”). For all non-approved services, "non-zero" return code indicates the service is not approved.

4.4 Non-Approved Services

Name	Description	Algorithms	Role
MD5	Message digest	MD5	CO
ECDH	Shared secret computation	ECDH	CO

[1] The list of public key flags allowed in approved mode of operation is described in 0.

Name	Description	Algorithms	Role
RSA	Encryption primitives; Decryption primitives; Signature verification primitives	RSA	CO
RSA with non-approved public key flags	Key generation; Signature generation; Signature verification	RSA with non-approved public key flags	CO
ECDSA	Signature verification primitives	ECDSA	CO
ECDSA with non-approved public key flags	Key generation; Signature generation; Signature verification	ECDSA with non-approved public key flags	CO
AES-GCM	Authenticated symmetric encryption; Authenticated symmetric decryption	AES-GCM	CO
AES-OCB	Authenticated symmetric encryption; Authenticated symmetric decryption	AES-OCB	CO
AES-EAX	Authenticated symmetric encryption; Authenticated symmetric decryption	AES-EAX	CO
AES-SIV	Authenticated symmetric encryption; Authenticated symmetric decryption	AES-SIV	CO

Table 15: Non-Approved Services

The table above lists the non-approved services. The details of the non-approved cryptographic algorithms available in non-approved mode can be found in the Non-Approved, Not Allowed Algorithms table.

4.5 External Software/Firmware Loaded

Not applicable.

4.6 Bypass Actions and Status [O]

Not applicable.

4.7 Cryptographic Output Actions and Status [O]

Not applicable.

4.8 Additional Information [O]

Not applicable.

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module is verified comparing the HMAC-SHA2-256 value calculated at run time with the HMAC-SHA2-256 value stored in the .hmac file that was computed at build time for each software component of the module. If the HMAC values do not match, the test fails and the module enters the Error state.

5.2 Initiate on Demand

Integrity tests are performed as part of the Pre-Operational Self-Tests.

The module provides the Self-Test service to perform self-tests on demand which includes the pre-operational self-test (i.e., integrity test) and cryptographic algorithm self-tests (CASTs). This service can be invoked relying on the gcry_control(GCRYCTL_SELFTEST) API function call or by powering-off and reloading the module. During the execution of the on-demand self-tests, services are not available, and no data output or input is possible. In addition, the integrity tests cannot be modified.

In order to verify whether the self-tests have succeeded and the module is in the Operational state, the calling application may invoke the gcry_control(GCRYCTL_OPERATIONAL_P). The function will return TRUE if the module is in the operational state, FALSE if the module is in the Error state.

5.3 Open-Source Parameters [O]

Not applicable.

5.4 Additional Information [O]

Not applicable.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied [O]:

The module shall be installed as stated in Section 11. 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 [O]

The application that requests cryptographic services is the single user of the module, even when the application is serving multiple clients. In the approved mode of operation, the ptrace(2) system call, the debugger (gdb(1)), and strace(1) shall be not used.

6.3 Additional Information [O]

The module does not support multiple concurrent operators.

7 Physical Security

7.1 Mechanisms and Actions Required [O]

N/A for this module.

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

7.2 User Placed Tamper Seals [O]

Number: Not applicable.

Placement: Not applicable.

Surface Preparation: Not applicable.

Operator Responsible for Securing Unused Seals: Not applicable.

Part Numbers: Not applicable.

7.3 Filler Panels [O]

Not applicable.

7.4 Fault Induction Mitigation [O]

Not applicable.

7.5 EFP/EFT Information [O]

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature			
HighTemperature			
LowVoltage			
HighVoltage			

Table 16: EFP/EFT Information

Not applicable.

7.6 Hardness Testing Temperature Ranges [O]

Temperature Type	Temperature
LowTemperature	
HighTemperature	

Table 17: Hardness Testing Temperatures

Not applicable.

7.7 Additional Information [O]

Not applicable.

8 Non-Invasive Security

8.1 Mitigation Techniques [O]

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

8.2 Effectiveness [O]

Not applicable.

8.3 Additional Information [O]

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 18: Storage Areas

The module does not perform persistent storage of SSPs. The SSPs are temporarily stored in the RAM in plaintext form. SSPs are provided to the module by the calling process and are destroyed when released by the appropriate zeroization function calls.

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 19: SSP Input-Output Methods

The module does not support manual SSP entry or intermediate SSP generation output. The SSPs are provided to the module via API input parameters in plaintext form and output via API output parameters in plaintext form within the physical perimeter of the operational environment. This is allowed by [FIPS140-3_IG] 9.5.A, according to the “CM Software to/from App via TOEPP Path” entry on the Key Establishment Table.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Wipe and free memory block allocated	Zeroizes the SSPs contained within the cipher handle	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 zeroization procedure succeeded.	By calling the cipher related zeroization API: gcry_cipher_close() and gcry_free() clears and frees symmetric ciphers context, gcry_mac_close() and gcry_free() clears and frees HMAC context, gcry_sexp_release() and gcry_mpi_release() and gcry_free() clears and frees RSA key structure, gcry_sexp_release() and gcry_mpi_release() and gcry_ctx_release() and gcry_mpi_point_release() and gcry_free() clears and frees ECDSA key structures, gcry_free() clears PBKDF context, gcry_ctrl(GCRYCTL_TERM_SECMEM) clears DRBG contexts
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 20: SSP Zeroization Methods

The memory occupied by SSPs is allocated by regular memory allocation operating system calls. The application that is acting as the CO is responsible for calling the appropriate zeroization functions provided in the module's API and listed in the table above. Calling gcry_free(), which will zeroize the SSPs and also invoke the corresponding API functions listed above to zeroize SSPs. The zeroization functions overwrite the memory occupied by SSPs with “zeros” and deallocate the memory with the regular memory deallocation operating system call. In case of abnormal

termination, or swap in/out of a physical memory page of a process, the keys in physical memory are overwritten by the Linux kernel before the physical memory is allocated to another process. The completion of a zeroization routine(s) will indicate that a zeroization procedure succeeded. 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 symmetric encryption, symmetric decryption, authenticated symmetric encryption, authenticated symmetric decryption, message authentication, key wrapping and unwrapping	128, 192, 256 - 128, 192, 256	Symmetric key - CSP			Symmetric encryption Symmetric decryption Symmetric encryption (for data storage) Symmetric decryption (for data storage) Authenticated symmetric encryption Authenticated symmetric decryption Key wrapping Key unwrapping Message authentication code
HMAC key	Used for hash-based message	112-256 - 112-256	Symmetric key - CSP			Message authentication code

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	authentication					
RSA public key	Used for digital signature verification	2048, 3072, 4096 - 112, 128, 149	Public key - PSP	Key pair generation		Digital signature verification
RSA private key	Used for digital signature generation	2048, 3072, 4096 - 112, 128, 149	Private key - CSP	Key pair generation		Digital signature generation
ECDSA public key	Used for digital signature verification, and public key verification	P-224, P-256, P-384, P-521 - 112, 128, 192, 256	Public key - PSP	Key pair generation		Digital signature verification Public key verification
ECDSA private key	Used for digital signature generation and public key verification	P-224, P-256, P-384, P-521 - 112, 128, 192, 256	Private key - CSP	Key pair generation		Digital signature generation Public key verification
Password or passphrase	Used for key derivation	At least 14 characters - N/A	Password - CSP			Key derivation
Derived key	Used for key derivation	112-256 - 112-256	Symmetric key - CSP			Key derivation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Entropy input	Used for random number generation (compliant with IG D.L)	256 - 256	Entropy Input - CSP			Random number generation
DRBG internal state: (V value, C value)	Used for random number generation (compliant with IG D.L)	112-256 - 112-256	Internal state - CSP	Random number generation		Random number generation
DRBG internal state: (V value, key)	Used for random number generation (compliant with IG D.L)	112-256 - 112-256	Internal state - CSP	Random number generation		Random number generation
DRBG seed	Used for random number generation (compliant with IG D.L)	256 - 256	Seed - CSP	Random number generation		Random number generation

Table 21: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES key	API input parameters	RAM:Plaintext	From service invocation to service	Wipe and free memory block allocated	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
			completion	Module Reset	
HMAC key	API input parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	
RSA public key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	RSA private key:Paired With DRBG internal state: (V value, C value):Generated From DRBG internal state: (V value, key):Generated From
RSA private key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	RSA public key:Paired With DRBG internal state: (V value, C value):Generated From DRBG internal state: (V value, key):Generated From
ECDSA public key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	ECDSA private key:Paired With DRBG internal state: (V value, C value):Generated From DRBG internal state: (V value,

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					key):Generate From
ECDSA private key	API input parameters API output parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	ECDSA public key:Paired With DRBG internal state: (V value, C value):Generated From DRBG internal state: (V value, key):Generated From
Password or passphrase	API input parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	Derived key:Derives
Derived key	API output parameters	RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	Password or passphrase:Derived From
Entropy input		RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	DRBG seed:Generates

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
DRBG internal state: (V value, C value)		RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	DRBG seed:Generated From
DRBG internal state: (V value, key)		RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	DRBG seed:Generated From
DRBG seed		RAM:Plaintext	From service invocation to service completion	Wipe and free memory block allocated Module Reset	Entropy input:Generated From DRBG internal state: (V value, C value):Generates DRBG internal state: (V value, key):Generates

Table 22: SSP Table 2

The tables above summarize the Sensitive Security Parameters (SSPs) that are used by the cryptographic services implemented in the module.

9.5 Transitions [O]

The SHA-1 algorithm, as implemented by the module, will be non-approved for all purposes starting January 1, 2030.

The RSA algorithm as implemented by the module conforms to FIPS 186-4, which has been superseded by FIPS 186-5. FIPS 186-4 has been withdrawn since February 3, 2024.

9.6 Additional Information [O]

Not applicable.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A3700)	Key size: 232 bits	Message authentication	SW/FW Integrity	Module is operational	Integrity test for libgcrypt.so.20.3.4
HMAC-SHA2-256 (A3701)	Key size: 232 bits	Message authentication	SW/FW Integrity	Module is operational	Integrity test for libgcrypt.so.20.3.4
HMAC-SHA2-256 (A3703)	Key size: 232 bits	Message authentication	SW/FW Integrity	Module is operational	Integrity test for libgcrypt.so.20.3.4
HMAC-SHA2-256 (A3704)	Key size: 232 bits	Message authentication	SW/FW Integrity	Module is operational	Integrity test for libgcrypt.so.20.3.4
HMAC-SHA2-256 (A3705)	Key size: 232 bits	Message authentication	SW/FW Integrity	Module is operational	Integrity test for libgcrypt.so.20.3.4

Table 23: Pre-Operational Self-Tests

The module performs the integrity test using HMAC-SHA-256. Further details of the integrity test are provided in Section 5.1.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A3700)	Encrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
AES-ECB (A3701)	Encrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A3703)	Encrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A3704)	Encrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A3700)	Decrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A3701)	Decrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-ECB (A3703)	Decrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-ECB (A3704)	Decrypt with 128, 192, 256 bit keys	KAT	CAST	Module becomes operational	Symmetric operation	Test runs at power-on before the integrity test
AES-CMAC (A3700)	128, 192, 256 bit keys, MAC generation, encrypt	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
AES-CMAC (A3701)	128, 192, 256 bit keys, MAC generation, encrypt	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
AES-CMAC (A3703)	128, 192, 256 bit keys, MAC generation, encrypt	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
AES-CMAC (A3704)	128, 192, 256 bit keys, MAC	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	generation, encrypt					integrity test
Counter DRBG (A3700)	AES with 128-bit key with DF, with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Counter DRBG (A3701)	AES with 128-bit key with DF, with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Counter DRBG (A3703)	AES with 128-bit key with DF, with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Counter DRBG (A3704)	AES with 128-bit key with DF, with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3700)	SHA-1 without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Hash DRBG (A3701)	SHA-1 without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3703)	SHA-1 without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3704)	SHA-1 without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3705)	SHA-1 without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3700)	SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3701)	SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
Hash DRBG (A3703)	SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3704)	SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
Hash DRBG (A3705)	SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
HMAC DRBG (A3700)	HMAC-SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
HMAC DRBG (A3701)	HMAC-SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC DRBG (A3703)	HMAC-SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
HMAC DRBG (A3704)	HMAC-SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
HMAC DRBG (A3705)	HMAC-SHA2-256 with and without PR	KAT	CAST	Module becomes operational	Compliant with SP 800-90Ar1	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-4) (A3700)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-4) (A3701)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-4) (A3703)	PKCS#1 v1.5 with 2048 bit	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	key and SHA2-256					integrity test
RSA SigGen (FIPS186-4) (A3704)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigGen (FIPS186-4) (A3705)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-4) (A3700)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-4) (A3701)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-4) (A3703)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigVer (FIPS186-4) (A3704)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
RSA SigVer (FIPS186-4) (A3705)	PKCS#1 v1.5 with 2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-4) (A3700)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-4) (A3701)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-4) (A3703)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigGen (FIPS186-4) (A3704)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
ECDSA SigGen (FIPS186-4) (A3705)	P-256 with SHA-256	KAT	CAST	Module becomes operational	Digital signature generation	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A3700)	P-256 with SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A3701)	P-256 with SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A3703)	P-256 with SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
ECDSA SigVer (FIPS186-4) (A3704)	P-256 with SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA SigVer (FIPS186-4) (A3705)	P-256 with SHA2-256	KAT	CAST	Module becomes operational	Digital signature verification	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3699)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3700)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3701)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3702)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3703)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
HMAC-SHA-1 (A3704)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA-1 (A3705)	SHA-1	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A3700)	SHA2-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A3701)	SHA2-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A3703)	SHA2-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-224 (A3704)	SHA2-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-224 (A3705)	SHA2-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A3700)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A3701)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A3703)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-256 (A3704)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
HMAC-SHA2-256 (A3705)	SHA2-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A3700)	SHA2-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A3701)	SHA2-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A3703)	SHA2-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-384 (A3704)	SHA2-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA2-384 (A3705)	SHA2-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A3700)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A3701)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A3703)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A3704)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA2-512 (A3705)	SHA2-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
HMAC-SHA3-224 (A3700)	SHA3-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A3701)	SHA3-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-224 (A3705)	SHA3-224	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A3700)	SHA3-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-256 (A3701)	SHA3-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
HMAC-SHA3-256 (A3705)	SHA3-256	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A3700)	SHA3-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A3701)	SHA3-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-384 (A3705)	SHA3-384	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A3700)	SHA3-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
HMAC-SHA3-512 (A3701)	SHA3-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						integrity test
HMAC-SHA3-512 (A3705)	SHA3-512	KAT	CAST	Module becomes operational	Message authentication	Test runs at power-on before the integrity test
PBKDF (A3700)	SHA-1 password length 24 characters, master key length of 200 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3701)	SHA-1 password length 24 characters, master key length of 200 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3703)	SHA-1 password length 24 characters,	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	master key length of 200 bits, iteration count of 4096, and salt length of 288 bits					integrity test
PBKDF (A3704)	SHA-1 password length 24 characters, master key length of 200 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3705)	SHA-1 password length 24 characters, master key length of 200 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3700)	SHA-256 password length 24 characters, master key	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	length of 320 bits, iteration count of 4096, and salt length of 288 bits					integrity test
PBKDF (A3701)	SHA-256 password length 24 characters, master key length of 320 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3703)	SHA-256 password length 24 characters, master key length of 320 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
PBKDF (A3704)	SHA-256 password length 24 characters, master key length of	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	320 bits, iteration count of 4096, and salt length of 288 bits					
PBKDF (A3705)	SHA-256 password length 24 characters, master key length of 320 bits, iteration count of 4096, and salt length of 288 bits	KAT	CAST	Module becomes operational	Password-based key derivation	Test runs at power-on before the integrity test
RSA KeyGen (FIPS186-4) (A3700)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA KeyGen (FIPS186-4) (A3701)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA KeyGen (FIPS186-4) (A3703)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA KeyGen (FIPS186-4) (A3704)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
RSA KeyGen (FIPS186-4) (A3705)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
ECDSA KeyGen (FIPS186-4) (A3700)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
ECDSA KeyGen (FIPS186-4) (A3701)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
ECDSA KeyGen (FIPS186-4) (A3703)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation
ECDSA KeyGen (FIPS186-4) (A3704)	Signature generation and verification	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
	with SHA2-256					
ECDSA KeyGen (FIPS186-4) (A3705)	Signature generation and verification with SHA2-256	PCT	PCT	Successful key pair generation	Signature generation & verification	Key pair generation

Table 24: Conditional Self-Tests

The table above lists the CASTs and PCTs performed by the module. All CASTs performed are in the form of the Known Answer Tests (KATs) and are run prior to performing the integrity test. The details of the integrity test are provided in Section 5.1.

The KAT includes comparison of the calculated output with the expected known answer, hard coded as part of the test vectors used in the test. If the values do not match, the KAT fails.

The module also performs the Pair-wise Consistency Tests (PCT) shown in the table above. If at least one of the tests fails, the module returns an error code and enters the Error state. When the module is in the Error state, no data is output, and cryptographic operations are not allowed.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A3700)	Message authentication	SW/FW Integrity	Whenever the module is powered on	Upon every power on
HMAC-SHA2-256 (A3701)	Message authentication	SW/FW Integrity	Whenever the module is powered on	Upon every power on

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A3703)	Message authentication	SW/FW Integrity	Whenever the module is powered on	Upon every power on
HMAC-SHA2-256 (A3704)	Message authentication	SW/FW Integrity	Whenever the module is powered on	Upon every power on
HMAC-SHA2-256 (A3705)	Message authentication	SW/FW Integrity	Whenever the module is powered on	Upon every power on

Table 25: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-ECB (A3700)	KAT	CAST	On Demand	Manually
AES-ECB (A3701)	KAT	CAST	On Demand	Manually
AES-ECB (A3703)	KAT	CAST	On Demand	Manually
AES-ECB (A3704)	KAT	CAST	On Demand	Manually
AES-ECB (A3700)	KAT	CAST	On Demand	Manually
AES-ECB (A3701)	KAT	CAST	On Demand	Manually
AES-ECB (A3703)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-ECB (A3704)	KAT	CAST	On Demand	Manually
AES-CMAC (A3700)	KAT	CAST	On Demand	Manually
AES-CMAC (A3701)	KAT	CAST	On Demand	Manually
AES-CMAC (A3703)	KAT	CAST	On Demand	Manually
AES-CMAC (A3704)	KAT	CAST	On Demand	Manually
Counter DRBG (A3700)	KAT	CAST	On Demand	Manually
Counter DRBG (A3701)	KAT	CAST	On Demand	Manually
Counter DRBG (A3703)	KAT	CAST	On Demand	Manually
Counter DRBG (A3704)	KAT	CAST	On Demand	Manually
Hash DRBG (A3700)	KAT	CAST	On Demand	Manually
Hash DRBG (A3701)	KAT	CAST	On Demand	Manually
Hash DRBG (A3703)	KAT	CAST	On Demand	Manually
Hash DRBG (A3704)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Hash DRBG (A3705)	KAT	CAST	On Demand	Manually
Hash DRBG (A3700)	KAT	CAST	On Demand	Manually
Hash DRBG (A3701)	KAT	CAST	On Demand	Manually
Hash DRBG (A3703)	KAT	CAST	On Demand	Manually
Hash DRBG (A3704)	KAT	CAST	On Demand	Manually
Hash DRBG (A3705)	KAT	CAST	On Demand	Manually
HMAC DRBG (A3700)	KAT	CAST	On Demand	Manually
HMAC DRBG (A3701)	KAT	CAST	On Demand	Manually
HMAC DRBG (A3703)	KAT	CAST	On Demand	Manually
HMAC DRBG (A3704)	KAT	CAST	On Demand	Manually
HMAC DRBG (A3705)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3700)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigGen (FIPS186-4) (A3701)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3703)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3704)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3705)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3700)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3701)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3703)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3704)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3705)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3700)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA SigGen (FIPS186-4) (A3701)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3703)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3704)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3705)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3700)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3701)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3703)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3704)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3699)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA-1 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3702)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3703)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3704)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA2-224 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA2-224 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA2-224 (A3703)	KAT	CAST	On Demand	Manually
HMAC-SHA2-224 (A3704)	KAT	CAST	On Demand	Manually
HMAC-SHA2-224 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3701)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A3703)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3704)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3703)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3704)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3703)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3704)	KAT	CAST	On Demand	Manually
HMAC-SHA2-512 (A3705)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA3-224 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA3-224 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA3-224 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA3-256 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA3-256 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA3-256 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA3-384 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA3-384 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA3-384 (A3705)	KAT	CAST	On Demand	Manually
HMAC-SHA3-512 (A3700)	KAT	CAST	On Demand	Manually
HMAC-SHA3-512 (A3701)	KAT	CAST	On Demand	Manually
HMAC-SHA3-512 (A3705)	KAT	CAST	On Demand	Manually
PBKDF (A3700)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
PBKDF (A3701)	KAT	CAST	On Demand	Manually
PBKDF (A3703)	KAT	CAST	On Demand	Manually
PBKDF (A3704)	KAT	CAST	On Demand	Manually
PBKDF (A3705)	KAT	CAST	On Demand	Manually
PBKDF (A3700)	KAT	CAST	On Demand	Manually
PBKDF (A3701)	KAT	CAST	On Demand	Manually
PBKDF (A3703)	KAT	CAST	On Demand	Manually
PBKDF (A3704)	KAT	CAST	On Demand	Manually
PBKDF (A3705)	KAT	CAST	On Demand	Manually
RSA KeyGen (FIPS186-4) (A3700)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-4) (A3701)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-4) (A3703)	PCT	PCT	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA KeyGen (FIPS186-4) (A3704)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-4) (A3705)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3700)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3701)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3703)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3704)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3705)	PCT	PCT	On Demand	Manually

Table 26: Conditional Periodic Information

The module does not implement periodic self-tests.

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error state	The module immediately stops	Software integrity test failure	Restart of the module	An error message related to the

Name	Description	Conditions	Recovery Method	Indicator
	functioning due to a self-test failure	CAST failure PCT failure		cause of the failure
Fatal Error state	The module immediately halts all cryptographic operations and transits to shutdown	Random numbers are requested in the Error state Cipher operations are requested on a deallocated handle	Restart of the module	The module is aborted and is not available for use

Table 27: Error States

When the module fails any pre-operational self-test or conditional test, the module will return an error code to indicate the error and will enter the Error state. Any further cryptographic operation is inhibited. The calling application can obtain the module state by calling the gcry_control(GCRYCTL_OPERATIONAL_P) API function. The function returns FALSE if the module is in the Error state, TRUE if the module is in the Operational state. In the Error state, all data output is inhibited, and no cryptographic operation is allowed. The error can be recovered by a restart (i.e., powering off and powering on) of the module.

The table above shows the error codes and their corresponding condition.

10.5 Operator Initiation of Self-Tests [O]

On-Demand self-tests can be performed by powering-off and reloading the module, which cause the module to run the pre-operational tests again. Invoking the gcry_control(GCRYCTL_SELFTEST) API function will run the pre-operational self-test and CASTs on demand. Information on the execution of these tests are detailed in Section 5.2.

10.6 Additional Information [O]

Not applicable.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Once the operating environment is configured following the instructions provided below, the Crypto Officer can install the Ubuntu packages containing the module listed in Section 11.2 using the Advanced Package Tool (APT) with the following command line:

```
$ sudo apt-get install libcgrypt20 libgcrypt20-hmac
```

All the Ubuntu packages are associated with hashes for integrity check. The integrity of the Ubuntu package is automatically verified by the packing tool during the installation of the module. The Crypto Officer shall not install the package if the integrity fails.

Once the Debian packages have been installed, the operator needs to check the output of the gcry_version() API, which should show the following name and version:

Canonical Ltd. Ubuntu 22.04 Libgcrypt Cryptographic Module 1.9.4-3ubuntu3+Fips1.2

FIPS configuration can be enabled automatically via the Ubuntu Advantage tool after attaching your subscription.

(1) To install the tool type the following commands:

```
$ sudo apt update
$ sudo apt install ubuntu-advantage-tools
```

(2) To activate the Ubuntu Pro subscription run:

```
$ sudo pro attach <your_pro_token>
```

(3) To enable approved mode run:

```
$ sudo pro enable fips
```

(4) To verify that approved mode is enabled run:

```
$ sudo pro status
```

The pro client will install the necessary packages for the approved mode, including the kernel and the bootloader. After this step you MUST reboot to put the system into approved mode. The reboot will boot into FIPS supported kernel and create the /proc/sys/crypto/fips_enabled entry which tells the FIPS certified modules to run in approved mode. If you do not reboot after installing and configuring the bootloader, approved mode is not yet enabled.

11.2 Administrator Guidance

The binaries of the module are contained in the Ubuntu packages for delivery. The Crypto Officer shall follow this Security Policy (Section 11.1) to configure the operational environment and install the module to be operated as a FIPS 140-3 validated module.

The following Ubuntu packages contain the FIPS validated module:

- x86_64 processors:
  - libgcrypt20_1.9.4-3ubuntu3+Fips1.2_amd64.deb
    libgcrypt20-hmac-1.9.4-3ubuntu3+Fips1.2_amd64.deb
- s390x processors:
  - libgcrypt20_1.9.4-3ubuntu3+Fips1.2_s390x.deb
    libgcrypt20-hmac-1.9.4-3ubuntu3+Fips1.2_s390x.deb
- ARM64 processors:
  - libgcrypt20_1.9.4-3ubuntu3+Fips1.2_arm64.deb
    libgcrypt20-hmac-1.9.4-3ubuntu3+Fips1.2_arm64.deb

The libgcrypt20-doc_1.9.4-3ubuntu3+Fips1.2.deb and libgcrypt20-doc_1.9.4-3ubuntu3.deb Ubuntu packages contain the man pages for the module.

11.3 Non-Administrator Guidance

There is no non-administrator guidance.

11.4 Design and Rules [O]

The user must not call malloc/free to create/release space for keys, let libgcrypt manage space for keys, which will ensure that the key memory is overwritten before it is released. gcry_control(GCRYCTL_TERM_SECMEM) needs to be called before the process is terminated.

11.5 Maintenance Requirements [O]

There are no maintenance requirements.

11.6 End of Life [O]

For secure sanitization of the cryptographic module, the module must first be powered off, which will zeroize all keys and CSPs in volatile memory. Then, for actual deprecation, the module shall be upgraded to a newer version that is FIPS 140-3 validated.

The module does not possess persistent storage of SSPs, so further sanitization steps are not required.

11.7 Additional Information [O]

Not applicable.

12 Mitigation of Other Attacks

12.1 Attack List [O]

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

12.2 Mitigation Effectiveness [O]

By default, the module uses the following blinding technique: instead of using the RSA decryption directly, a blinded value $y = x\, r^e \ mod\ n$ is decrypted and the unblinded value $x' = y'\, r^{-1} \ mod\ n$ returned.

12.3 Guidance and Constraints [O]

The blinding value r is a random value with the size of the modulus n.

12.4 Additional Information [O]

Not applicable.

Appendix A. Approved Public Key Flags

Below are listed the approved public key flags for an input s-expression:

curve	*d*	*data*	*e*	*ecdsa*	*flags*
genkey	*hash*	*hash-algo*	*n*	*nbits*	*pkcs1*
private-key	*pss*	*public-key*	*q*	*r*	*raw*
rsa	*rsa-use-e*	*s*	*salt-length*	*sig-val*	*value*

Appendix B. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DF	Derivation Function
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECDSA	Elliptic Curve Digital Signature Algorithm
FIPS	Federal Information Processing Standards Publication
FSM	Finite State Model
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
KAS	Key Agreement Schema
KAT	Known Answer Test
KW	AES Key Wrap
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PR	Prediction Resistance
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm

SHS	Secure Hash Standard
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

Appendix C. References

FIPS140-3 **FIPS PUB 140-3 - Security Requirements For Cryptographic Modules**
March 2019
https://doi.org/10.6028/NIST.FIPS.140-3

FIPS140-3_IG **Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program**
September 2020
https://csrc.nist.gov/Projects/cryptographic-module-validation-program/fips-140-3-ig-announcements

FIPS180-4 **Secure Hash Standard (SHS)**
March 2012
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

FIPS186-4 **Digital Signature Standard (DSS)**
July 2013
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

FIPS197 **Advanced Encryption Standard**
November 2001
https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

FIPS198-1 **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

FIPS202 **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
https://www.ietf.org/rfc/rfc3447.txt

SP800-38A **NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques**
December 2001
https://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

SP800-38B	**NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication** May 2005 https://csrc.nist.gov/publications/detail/sp/800-38b/final
SP800-38C	**NIST Special Publication 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
SP800-38E	**NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices** January 2010 https://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf
SP800-38F	**NIST Special Publication 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
SP800-57	**NIST Special Publication 800-57 Part 1 Revision 5 - Recommendation for Key Management Part 1: General** May 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf
SP800-90A	**NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators** June 2015 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf
SP800-90B	**NIST Special Publication 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

SP800-108	**NIST Special Publication 800-108 - Recommendation for Key Derivation Using Pseudorandom Functions (Revised)** October 2009 https://csrc.nist.gov/publications/nistpubs/800-108/sp800-108.pdf
SP800-132	**NIST Special Publication 800-132 - Recommendation for Password-Based Key Derivation - Part 1: Storage Applications** December 2010 https://csrc.nist.gov/publications/nistpubs/800-132/nist-sp800-132.pdf
SP800-133	**NIST Special Publication 800-133 - Recommendation for Cryptographic Key Generation** June 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-133r2.pdf
SP800-140B	**NIST Special Publication 800-140B - CMVP Security Policy Requirements** March 2020 https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140B.pdf