SafeLogic, Inc.

CryptoComply Native Go Module

FIPS 140-3 Non-Proprietary Security Policy

Software Versions 1.0.0

Document Version 1.0

May 27, 2026

SafeLogic, Inc.
8300 Boone Blvd., Suite 500
Vienna, VA 22182
www.safelogic.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document provides a non-proprietary FIPS 140-3 Security Policy for CryptoComply Native Go Module.

SafeLogic, Inc.'s CryptoComply Native Go Module is designed to provide FIPS 140-3 validated cryptographic functionality and is available for licensing. For more information, visit www.safelogic.com/cryptocomply.

1.1.1 About FIPS 140

Federal Information Processing Standards Publication 140-3, Security Requirements for Cryptographic Modules, (FIPS 140-3) specifies the latest requirements for cryptographic modules utilized to protect sensitive but unclassified information. The National Institute of Standards and Technology (NIST) and Canadian Centre for Cyber Security (CCCS) collaborate to run the Cryptographic Module Validation Program (CMVP), which assesses conformance to FIPS 140. NIST (through NVLAP) accredits independent testing labs to perform FIPS 140 testing. The CMVP reviews and validates modules tested against FIPS 140 criteria. *Validated* is the term given to a module that has successfully gone through this FIPS 140 validation process. Validated modules receive a validation certificate that is posted on the CMVP's website.

More information is available on the CMVP website at:
https://csrc.nist.gov/projects/cryptographic-module-validation-program.

1.1.2 About this Document

This non-proprietary cryptographic module Security Policy for CryptoComply Native Go Module from SafeLogic, Inc. (SafeLogic) provides an overview of the product and a high-level description of how it meets the security requirements of FIPS 140-3. This document includes details on the module's cryptographic capabilities, services, sensitive security parameters, and self-tests. This Security Policy also includes guidance on operating the module while maintaining compliance with FIPS 140-3.

CryptoComply Native Go Module may also be referred to as the "module" in this document.

1.1.3 External Resources

The SafeLogic website (www.safelogic.com) contains information on SafeLogic services and products. The CMVP website maintains all FIPS 140 certificates for SafeLogic's FIPS 140 validations. These certificates also include SafeLogic contact information.

1.1.4 Notices

This document may be freely reproduced and distributed, but only in its entirety and without modification.

1.2 Security Levels

The following table lists the module's level of validation for each area in FIPS 140-3.

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	N/A
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	1
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

CryptoComply Native Go Module is a standards-based cryptographic module. It delivers core cryptographic functions to the Go standard library and other Go applications. The module features robust algorithm support, including quantum-resistant and classical algorithms.

Module Type: Software

Module Embodiment: Multi-Chip Standalone

Module Characteristics:

Cryptographic Boundary:

The module's cryptographic boundary is delimited by the module's components, as well as the instantiation of the cryptographic module saved in memory and executed by the processor. The executable files that constitute the cryptographic module are listed in Security Policy Section 2.2 - Tested and Vendor Affirmed Module Version and Identification. Additionally, the module's integrity value is included inside the boundary.

Refer to the block diagram in Figure 1 for additional detail.

Tested Operational Environment's Physical Perimeter (TOEPP):

As a software cryptographic module, the module operates within the Tested Operational Environment's Physical Perimeter (TOEPP). The TOEPP consists of the Operating System (OS) and the physical perimeter of the General Purpose Computer (GPC). This TOEPP comprises the Operational Environment (OE) that the module operates in, the module itself, and all other applications that operate within the OE, including the host application for the module. Where available, Processor Algorithm Acceleration (PAA) functions are provided by the processor in the Operational Environment. Entropy is provided as specified in Security Policy Section 2.8 - RBG and Entropy).

Refer to the block diagram in Figure 1 for additional detail.

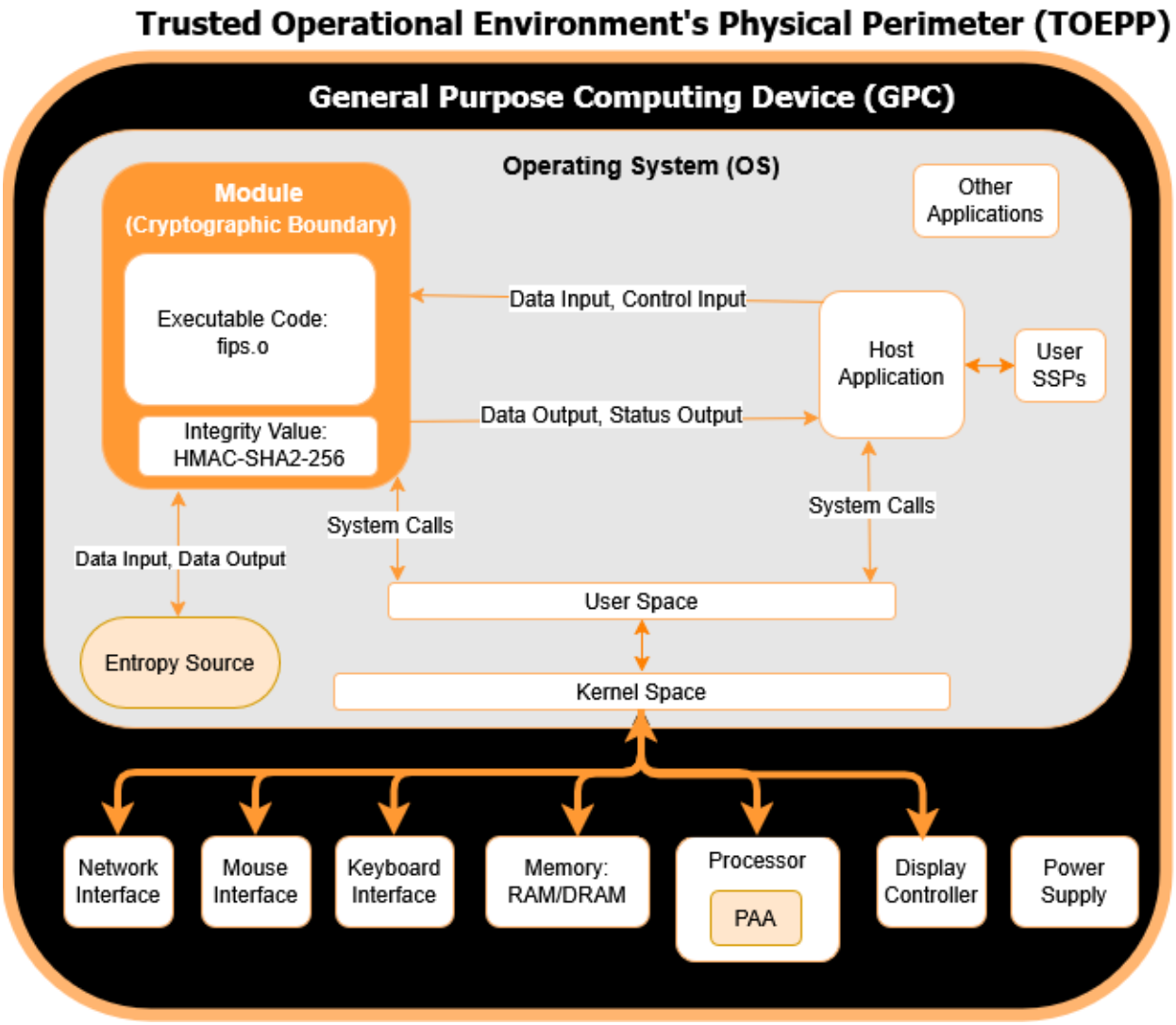

Figure 1: Block Diagram

The module's block diagram depicts the cryptographic boundary, TOEPP, and the components of each. Additionally, it depicts the data flow between these components. The module's logical interfaces are defined by its API. These interfaces are used by the host application to interact with the module. All input to the module occurs through the data input interface or control input interface. All output from the module occurs through the data output interface or status output interface. Refer also to Security Policy Section 3 - Cryptographic Module Interfaces and Section 9.2 - SSP Input-Output Methods.

The module executes within the operating environments specified in Security Policy Section 2.2 - Tested and Vendor Affirmed Module Version and Identification.

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
fips.o (darwin/arm64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (freebsd/amd64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/amd64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/arm)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/arm64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/mips64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/ppc64le)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (linux/s390x)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (windows/amd64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	HMAC-SHA2-256
fips.o (windows/arm64)	v1.0.0	Binary file compiled from the source in the v1.0.0-c2097c7c.zip file	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:

The module operates in a modifiable operational environment under the FIPS 140-3 definitions. The module operates on a general purpose computer (GPC) running a general purpose operating system (GPOS).

The module was tested in the following operating environments.

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Red Hat Enterprise Linux 9	Dell PowerEdge R660	Intel Xeon Silver 4410Y	Yes	N/A	v1.0.0
Red Hat Enterprise Linux 9	Dell PowerEdge R660	Intel Xeon Silver 4410Y	No	N/A	v1.0.0
Red Hat Enterprise Linux 9	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	Yes	N/A	v1.0.0
Red Hat Enterprise Linux 9	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	No	N/A	v1.0.0
Red Hat Enterprise Linux 9	IBM z16 3931-A01	IBM Z System z16	Yes	PR/SM Driver Level 51 with Bundle Level H29	v1.0.0
Red Hat Enterprise Linux 9	IBM z16 3931-A01	IBM Z System z16	No	PR/SM Driver Level 51 with Bundle Level H29	v1.0.0
Red Hat Enterprise Linux 9	IBM 9080-HEX	IBM Power10	Yes	PowerVM FW1040.00 with VIOS 3.1.3.00	v1.0.0
Red Hat Enterprise Linux 9	IBM 9080-HEX	IBM Power10	No	PowerVM FW1040.00 with VIOS 3.1.3.00	v1.0.0
Alpine Linux 3.20 image	Dell PowerEdge R660	Intel Xeon Silver 4410Y	Yes	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0
Alpine Linux 3.20 image	Dell PowerEdge R660	Intel Xeon Silver 4410Y	No	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0
Alpine Linux 3.20 image	Dell PowerEdge R6615	AMD EPYC 9454P	Yes	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Alpine Linux 3.20 image	Dell PowerEdge R6615	AMD EPYC 9454P	No	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0
Alpine Linux 3.20 image	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	Yes	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0
Alpine Linux 3.20 image	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	No	Podman 4 on Red Hat Enterprise Linux 9	v1.0.0
Amazon Linux 2023	Amazon EC2 r8g.metal-24xl	AWS Graviton4	Yes	N/A	v1.0.0
Amazon Linux 2023	Amazon EC2 r8g.metal-24xl	AWS Graviton4	No	N/A	v1.0.0
Amazon Linux 2023	Amazon EC2 c6i.metal	Intel Xeon Platinum 8375C	Yes	N/A	v1.0.0
Amazon Linux 2023	Amazon EC2 c6i.metal	Intel Xeon Platinum 8375C	No	N/A	v1.0.0
Oracle Linux 9	Dell PowerEdge R660	Intel Xeon Silver 4410Y	Yes	N/A	v1.0.0
Oracle Linux 9	Dell PowerEdge R660	Intel Xeon Silver 4410Y	No	N/A	v1.0.0
Oracle Linux 9	Dell PowerEdge R6615	AMD EPYC 9454P	Yes	N/A	v1.0.0
Oracle Linux 9	Dell PowerEdge R6615	AMD EPYC 9454P	No	N/A	v1.0.0
Oracle Linux 9	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	Yes	N/A	v1.0.0
Oracle Linux 9	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	No	N/A	v1.0.0
Linux 5.4	Ubiquiti EdgeRouter 4	Marvell OCTEON III CN7130	No	N/A	v1.0.0
Linux 5.4	CZ.NIC Turris Omnia	Marvell ARMADA-385	No	N/A	v1.0.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Google Prodimage with Linux 5.10	APIF-824	AMD EPYC 7B12	Yes	N/A	v1.0.0
Google Prodimage with Linux 5.10	APIF-824	AMD EPYC 7B12	No	N/A	v1.0.0
Google Prodimage with Linux 5.10	APIF-738	Intel Xeon Platinum 8273CL	Yes	N/A	v1.0.0
Google Prodimage with Linux 5.10	APIF-738	Intel Xeon Platinum 8273CL	No	N/A	v1.0.0
Google Prodimage with Linux 5.10	APIF-091	ARM Neoverse-N1	Yes	N/A	v1.0.0
Google Prodimage with Linux 5.10	APIF-091	ARM Neoverse-N1	No	N/A	v1.0.0
Apple macOS 15	MacBook Air (M2, 2022)	Apple M2	Yes	N/A	v1.0.0
Apple macOS 15	MacBook Air (M2, 2022)	Apple M2	No	N/A	v1.0.0
FreeBSD 14	Lenovo ThinkCentre M73	Intel Core i3-4130T	Yes	N/A	v1.0.0
FreeBSD 14	Lenovo ThinkCentre M73	Intel Core i3-4130T	No	N/A	v1.0.0
Microsoft Windows Server 2022	Lenovo ThinkCentre M73	Intel Core i3-4130T	Yes	N/A	v1.0.0
Microsoft Windows Server 2022	Lenovo ThinkCentre M73	Intel Core i3-4130T	No	N/A	v1.0.0
Microsoft Windows 11	Surface Laptop 7th Edition	Qualcomm Snapdragon X Plus	Yes	N/A	v1.0.0
Microsoft Windows 11	Surface Laptop 7th Edition	Qualcomm Snapdragon X Plus	No	N/A	v1.0.0

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
SUSE Linux Enterprise Server 15 SP6	Dell PowerEdge R660	Intel Xeon Silver 4410Y	Yes	N/A	v1.0.0
SUSE Linux Enterprise Server 15 SP6	Dell PowerEdge R660	Intel Xeon Silver 4410Y	No	N/A	v1.0.0
SUSE Linux Enterprise Server 15 SP6	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	Yes	N/A	v1.0.0
SUSE Linux Enterprise Server 15 SP6	ASRock Rack ALTRAD8UD-1L2T	Ampere Altra Q64-22	No	N/A	v1.0.0

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

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

Operating System	Hardware Platform
Linux 3.10+	x86-64 architecture
Linux 3.10+	ARMv8 architecture
Linux 3.10+	ARMv9 architecture
Linux 3.10+	ARMv7 architecture
Linux 4.x	x86-64 architecture
Linux 4.x	ARMv8 architecture
Linux 4.x	ARMv9 architecture
Linux 4.x	ARMv7 architecture
Linux 5.x	x86-64 architecture
Linux 5.x	ARMv8 architecture
Linux 5.x	ARMv9 architecture
Linux 5.x	ARMv7 architecture

Operating System	Hardware Platform
Linux 6.x	x86-64 architecture
Linux 6.x	ARMv8 architecture
Linux 6.x	ARMv9 architecture
Linux 6.x	ARMv7 architecture
Linux 7.x	x86-64 architecture
Linux 7.x	ARMv8 architecture
Linux 7.x	ARMv9 architecture
Linux 7.x	ARMv7 architecture
Apple macOS 11	Apple M series
Apple macOS 12	Apple M series
Apple macOS 13	Apple M series
Apple macOS 14	Apple M series
Apple macOS 15	Apple M series
Apple macOS 26	Apple M series
FreeBSD 12	x86-64 architecture
FreeBSD 13	x86-64 architecture
FreeBSD 14	x86-64 architecture
FreeBSD 15	x86-64 architecture
Microsoft Windows 10	x86-64 architecture
Microsoft Windows 11	x86-64 architecture

Operating System	Hardware Platform
Microsoft Windows 11	ARMv8 architecture
Microsoft Windows 11	ARMv9 architecture
Microsoft Windows Server 2016	x86-64 architecture
Microsoft Windows Server 2019	x86-64 architecture
Microsoft Windows Server 2022	x86-64 architecture
Microsoft Windows Server 2025	x86-64 architecture
Microsoft Windows Server 2025	ARMv8 architecture
Microsoft Windows Server 2025	ARMv9 architecture
Red Hat Enterprise Linux 10	x86-64 architecture
Red Hat Enterprise Linux 10	ARMv8 architecture
Red Hat Enterprise Linux 10	ARMv9 architecture
Red Hat Enterprise Linux 10	IBM Z System z16

Operating System	Hardware Platform
Red Hat Enterprise Linux 10	IBM Power10
Oracle Linux 7 with UEK6	Oracle X series servers
Oracle Linux 7 with UEK6	Oracle E series servers
Oracle Linux 7 with UEK6	Oracle A series servers
Oracle Linux 7 with UEK6	Marvell OCTEON TX2 CN93XX
Oracle Linux 7 with UEK6	AMD Pensando DSC2-2Q200
Oracle Linux 7 with UEK6	Marvell OCTEON III CN73XX
Oracle Linux 7 with UEK6	ARM Cortex-A7
Oracle Linux 7 with UEK6	ARM Cortex-A9
Oracle Linux 8 with UEK7	Oracle X series servers
Oracle Linux 8 with UEK7	Oracle E series servers
Oracle Linux 8 with UEK7	Oracle A series servers
Oracle Linux 9 with UEK7	Oracle X series servers
Oracle Linux 9 with UEK7	Oracle E series servers
Oracle Linux 9 with UEK7	Oracle A series servers

Operating System	Hardware Platform
Oracle Linux 9 with UEK7	Marvell OCTEON TX2 CN93XX
Oracle Linux 9 with UEK7	AMD Pensando DSC2-2Q200
Oracle Linux 9 with UEK7	Marvell OCTEON III CN73XX
Windows Server 2022	Dell XR4510c with Intel Xeon D-2776NT
Windows Server 2025	Dell XR4510c with Intel Xeon D-2776NT
Azure Local 22H2	Dell XR4510c with Intel Xeon D-2776NT
Red Hat Enterprise Linux 8.10 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Red Hat Enterprise Linux 8.8 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Red Hat Enterprise Linux 8.5 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Red Hat Enterprise Linux 9.5 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Rocky Linux 8.5 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Ubuntu Linux 22.04 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT

Operating System	Hardware Platform
Ubuntu Linux 24.04 on Hyper-V	Dell XR4510c with Intel Xeon D-2776NT
Amazon Linux 2023	x86-64 architecture
Amazon Linux 2023	ARMv8 architecture
Amazon Linux 2023	ARMv9 architecture
Amazon Linux 2023	AWS Graviton2
Amazon Linux 2023	AWS Graviton3
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	N1000
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	N3000
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	N3001
HPE ANW ClearPass	VMware ESXi 7.0 To latest 8.x

Operating System	Hardware Platform
Policy Manager (CPPM) 6.14.0 or later	
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	Microsoft HyperV 2019R2 and goes to 2024 releases
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	Linux KVM Ubuntu 22.04
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	Nutanix AHV
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	HPE VM Morpheus
HPE ANW ClearPass Policy Manager (CPPM)	Amazon Web Services

Operating System	Hardware Platform
6.14.0 or later	
HPE ANW ClearPass Policy Manager (CPPM) 6.14.0 or later	Microsoft Azure
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	4100i
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	5420
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6000
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6100

Operating System	Hardware Platform
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6200F
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6200M
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6300
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6300L
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	6400
HPE ANW CX Switch Operating System (AOS-	8100

Operating System	Hardware Platform
CX) 10.18.1000 or later	
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8360
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8320
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8325
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8325H
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8325P

Operating System	Hardware Platform
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	8400
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	9300
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	9300S
HPE ANW CX Switch Operating System (AOS-CX)10.18.1000 or later	10000
HPE ANW CX Switch Operating System (AOS-CX) 10.18.1000 or later	10040
Alpine Linux v3.16 (or later)	AMD64

Operating System	Hardware Platform
Alpine Linux v3.16 (or later)	ARM64
Debian GNU/Linux 12 (or later)	AMD64
Debian GNU/Linux 12 (or later)	ARM64
Debian GNU/Linux 13 (trixie) (or later)	AMD64
Debian GNU/Linux 13 (trixie) (or later)	ARM64
Ubuntu Linux 22.04	AMD64
Ubuntu Linux 22.04	ARM64
Ubuntu Linux 24.04	AMD64
Ubuntu Linux 24.04	ARM64
HPE Alletra Storage ArcusOS Release 10.6 (or later) on Debian GNU/Linux 12 (or later)	HPE Alletra Storage MP B10000

Operating System	Hardware Platform
Ubuntu Pro 22.04 LTS x86_64	Generic Hardware Platform
Ubuntu Pro 22.04 LTS arm64	Generic Hardware Platform
Ubuntu Pro 24.04 LTS x86_64	Generic Hardware Platform
Ubuntu Pro 24.04 LTS arm64	Generic Hardware Platform
Bottlerocket x86_64 FIPS	Generic Hardware Platform
Bottlerocket arm64 FIPS	Generic Hardware Platform
Amazon Linux 2023 x86_64	Generic Hardware Platform
Amazon Linux 2023 arm64	Generic Hardware Platform
Rocky 8.10	Generic/ Standard ProLiants
Rocky 9.7	Generic/ Standard ProLiants
Rocky 9.8	Generic/ Standard ProLiants
Rocky 9.7	Synergy Composer 2
Rocky 9.8	Synergy Composer 2
Rocky 9.7	Synergy Composer 2 Plus
Rocky 9.8	Synergy Composer 2 Plus
Windows 2019	Generic/Standard ProLiants

Operating System	Hardware Platform
Windows 2022	Generic/Standard ProLiants
Windows 2025	Generic/Standard ProLiants
Azure Stack HCI 23H2	Generic/Standard ProLiants
Azure Local 24H2	Generic/Standard ProLiants
Red Hat Enterprise Linux 9 Server	Generic/Standard ProLiants
Red Hat Enterprise Linux 10	Generic/Standard ProLiants
SUSE Linux Enterprise Server 15	Generic/Standard ProLiants
SUSE Linux Enterprise Server 16	Generic/Standard ProLiants
Linux Kernel 6.12.62 and later	All Gen 13 and later generation of Rack Servers
Alpine Linux v3.16 (or later)	HPE Alletra Storage MP X10000
Debian GNU/Linux 12 (or later)	HPE Alletra Storage MP X10000
Red Hat Enterprise Linux 8.10 or later	HPE Alletra Storage MP X10000

Operating System	Hardware Platform
Debian GNU/Linux 13 (or later)	HPE Alletra Storage MP X10000
HPE StoreOnce Software 5.2.0 or later	HPE StoreOnce
Ubuntu 22.04/24.04	VM (amd64 based CPU)
Linux 5.15.167	AP36/AP36M, AP37, AP66/AP66D, AP17, AP27/AP27E
Linux 5.15.167	AP47/AP47D/AP47E
Junos 21.4R3-S4	EX2300/EX3400
Junos 22.4R2-S1	EX4000/EX4100
Junos 23.1R2	EX4300-MP/EX4400/EX4650/EX92xx/EX10k/QFX5110/QFX5120/QFX5200/QFX5210/MX204/MX301/MX304
Junos Evolved 23.2R1	QFX5130/QFX5220/QFX5700
Debian 11+	Mist-Edge or VM (amd64 based CPU)
Security Director Cloud 26.2.1 or later on Host OS Ubuntu 22.04 LTS and Container OS Ubuntu 24.04 LTS	VM (VMWare/KVM/Azure…)
Security Director On-	VM (VMWare/KVM/Azure…)

Operating System	Hardware Platform
Prem 26.2.1 or later on Host OS Ubuntu 20.04.6 LTS and Container OS Ubuntu 24.04 LTS	
FreeBSD 12+ based Junos	SRX1500/SRX1600/SRX4100/SRX4200/SRX4300/SRX4600/SRX4700/SRX2300
Junos EVO	SRX400

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

Porting guidance is defined in the FIPS 140-3 CMVP Management Manual Section 7.9. FIPS 140-3 validation compliance can be maintained when the following requirements are met:

- No source code modifications are required to recompile the module and port it to another operating environment
- The module is operating on any general-purpose platform/processor that supports the specified operating system as listed on the validation entry or another compatible operating system.

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

Not applicable.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	State in which the module is performing approved cryptographic services	Approved	ServiceIndicator() returns true

Mode Name	Description	Type	Status Indicator
Non-Approved Mode	State in which the module is performing non-approved cryptographic services	Non-Approved	ServiceIndicator() returns false

Table 5: Modes List and Description

The module provides both approved and non-approved modes of operation.

Mode Change Instructions and Status:

The approved mode of operation is entered automatically upon invocation of an approved service and exited automatically upon invocation of a non-approved service. For more details, refer to Security Policy Section 4.3 - Approved Services and Section 4.4 - Non-Approved Services.

2.5 Algorithms

2.5.1 Approved Algorithms

Approved Algorithms:

The module implements the following approved algorithms that have been tested by the Cryptographic Algorithm Validation Program (CAVP).

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A6650	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CMAC	A6650	Direction - Generation, Verification Key Length - 128, 256 MAC Length - MAC Length: 8-128 Increment 8 Message Length - Message Length: 0-524288 Increment 8	SP 800-38B
AES-CTR	A6650	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - Yes Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
AES-GCM	A6650	Direction - Decrypt, Encrypt IV Generation - External, Internal Key Length - 128, 192, 256 Tag Length - 104, 112, 120, 128, 96 IV Length - IV Length: 96	SP 800-38D

Algorithm	CAVP Cert	Properties	Reference
		Payload Length - Payload Length: 0-65536 Increment 8 AAD Length - AAD Length: 0-65536 Increment 8 IV Generation Mode - 8.2.2	
Counter DRBG	A6650	Prediction Resistance - No Supports Reseed - Yes Mode - AES-256 Derivation Function Enabled - No Additional Input - Additional Input: 384 Entropy Input - Entropy Input: 384 Nonce - Nonce: 0 Personalization String Length - Personalization String Length: 0 Returned Bits - 128	SP 800-90A Rev. 1
cSHAKE-128	A6650	Message Length - Message Length: 0-65536 Increment 8 Output Length - Output Length: 16-65536 Increment 8 Hex Customization - No	SP 800-185
cSHAKE-256	A6650	Message Length - Message Length: 0-65536 Increment 8 Output Length - Output Length: 16-65536 Increment 8 Hex Customization - No	SP 800-185
Deterministic ECDSA SigGen (FIPS186-5)	A6650	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512	FIPS 186-5
ECDSA KeyGen (FIPS186-5)	A6650	Curve - P-224, P-256, P-384, P-521 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-5)	A6650	Curve - P-224, P-256, P-384, P-521	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A6650	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A6650	Curve - P-224, P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
EDDSA KeyGen	A6650	Curve - ED-25519	FIPS 186-5
EDDSA KeyVer	A6650	Curve - ED-25519	FIPS 186-5
EDDSA SigGen	A6650	Curve - ED-25519 Context Length - Context Length: 0-255 Increment 1 PreHash - Yes Pure - Yes	FIPS 186-5
EDDSA SigVer	A6650	Curve - ED-25519 PreHash - Yes Pure - Yes	FIPS 186-5
HMAC DRBG	A6650	Prediction Resistance - No Supports Reseed - No Mode - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512 Entropy Input - Entropy Input: 192, Entropy Input: 256 Nonce - Nonce: 128, Nonce: 96 Personalization String Length - Personalization String Length: 192, Personalization String Length: 256 Additional Input - Additional Input: 0 Returned Bits - 224, 256, 384, 512	SP 800-90A Rev. 1
HMAC-SHA2-224	A6650	MAC - MAC: 32-224 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-256	A6650	MAC - MAC: 32-256 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-384	A6650	MAC - MAC: 32-384 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512	A6650	MAC - MAC: 32-512 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/224	A6650	MAC - MAC: 32-224 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA2-512/256	A6650	MAC - MAC: 32-256 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA3-224	A6650	MAC - MAC: 32-224 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA3-256	A6650	MAC - MAC: 32-256 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA3-384	A6650	MAC - MAC: 32-384 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
HMAC-SHA3-512	A6650	MAC - MAC: 32-512 Increment 8 Key Length - Key Length: 8-524288 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A6650	Domain Parameter Generation Methods - P-224, P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder staticUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDA HKDF Sp800-56Cr1	A6650	Fixed Info Pattern - uPartyInfo||vPartyInfo Fixed Info Encoding - concatenation Derived Key Length - 2048 Shared Secret Length - Shared Secret Length: 224-65336 Increment 8 HMAC Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512	SP 800-56C Rev. 2
KDA OneStepNoCounter SP800-56Cr2	A6650	Auxiliary Function Methods - Auxiliary Function Name - HMAC-SHA2-224 MAC Salting Methods - default, random Key Length - 224 Fixed Info Pattern - uPartyInfo||vPartyInfo Fixed Info Encoding - concatenation Key Length - Key Length: 224-65336 Increment 8	SP 800-56C Rev. 2
KDF SP800-108	A6650	KDF Mode - Counter, Feedback MAC Mode - CMAC-AES128, CMAC-AES192, CMAC-AES256, HMAC-SHA2-224, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, HMAC-SHA2-512/224, HMAC-SHA2-512/256, HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 Supported Lengths - Supported Lengths: 256, Supported Lengths: 8-4096 Increment 8 Fixed Data Order - After Fixed Data, Before Fixed Data Counter Length - 16, 8	SP 800-108 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
		Supports Empty IV - No, Yes Custom Key In Length - 0 Requires Empty IV - Yes	
KDF SSH (CVL)	A6650	Cipher - AES-128, AES-192, AES-256 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	SP 800-135 Rev. 1
KTS-IFC	A6650	IUT ID - C0FFEE Modulo - 2048, 3072, 4096 Key Generation Methods - rsakpg1-basic Fixed Public Exponent - 010001 Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Transport Method - Hash Algorithms - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512 Supports Null Associated Data - Yes Associated Data Encoding - concatenation Key Length - 1024	SP 800-56B Rev. 2
ML-KEM EncapDecap	A6650	Parameter Sets - ML-KEM-1024, ML-KEM-768 Functions - Decapsulation, Encapsulation	FIPS 203
ML-KEM KeyGen	A6650	Parameter Sets - ML-KEM-1024, ML-KEM-768	FIPS 203
PBKDF	A6650	Iteration Count - Iteration Count: 1-10000 Increment 1 HMAC Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512 Password Length - Password Length: 8-64 Increment 1 Salt Length - Salt Length: 128-512 Increment 8 Key Data Length - Key Data Length: 112-4096 Increment 8	SP 800-132
RSA KeyGen (FIPS186-5)	A6650	Key Generation Mode - probable Modulo - 2048, 3072, 4096 Primality Tests - 2powSecStr Fixed Public Exponent - 010001 Info Generated By Server - Yes Private Key Format - standard Public Exponent Mode - fixed	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
RSA SigGen (FIPS186-5)	A6650	Hash Pair - Hash Algorithm - SHA2-224 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1	FIPS 186-5
RSA SigVer (FIPS186-5)	A6650	Hash Pair - Hash Algorithm - SHA2-224 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1 Fixed Public Exponent - 010001 Public Exponent Mode - fixed	FIPS 186-5
SHA2-224	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA2-256	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA2-384	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA2-512	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA2-512/224	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA2-512/256	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 180-4
SHA3-224	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 202
SHA3-256	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 202
SHA3-384	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 202
SHA3-512	A6650	Message Length - Message Length: 0-65528 Increment 8	FIPS 202
SHAKE-128	A6650	Supports Bit-Oriented Messages - No Supports Empty Message - Yes Supports Bit-Oriented Output - No Output Length - Output Length: 16-65536 Increment 8	FIPS 202
SHAKE-256	A6650	Supports Bit-Oriented Messages - No Supports Empty Message - Yes Supports Bit-Oriented Output - No Output Length - Output Length: 16-65536 Increment 8	FIPS 202
TLS v1.2 KDF RFC7627 (CVL)	A6650	Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Key Block Length - Key Block Length: 1024	SP 800-135 Rev. 1

Algorithm	CAVP Cert	Properties	Reference
TLS v1.3 KDF (CVL)	A6650	HMAC Algorithm - SHA2-256, SHA2-384 KDF Running Modes - DHE, PSK, PSK-DHE	SP 800-135 Rev. 1

Table 6: Approved Algorithms

2.5.2 Vendor-Affirmed Algorithms

Vendor-Affirmed Algorithms:

The module implements the following vendor affirmed algorithms that are approved for use in an approved mode.

Name	Properties	Implementation	Reference
CKG	Key Type:Symmetric and Asymmetric	Go Cryptographic Module	SP 800-133rev2 Section 4, Section 6.3

Table 7: Vendor-Affirmed Algorithms

2.5.3 Non-Approved, Allowed Algorithms

Non-Approved, Allowed Algorithms:

The module does not implement any non-approved algorithms that are allowed for use in an approved mode.

N/A for this module.

2.5.4 Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Allowed Algorithms with No Security Claimed:

The module does not implement any non-approved algorithms with no security claimed for use in an approved mode.

N/A for this module.

2.5.5 Non-Approved, Not Allowed Algorithms

Non-Approved, Not Allowed Algorithms:

The module implements the following non-approved, not allowed algorithms.

Name	Use and Function
AES ECB	Using Block.Encrypt, Block.Decrypt for encryption/decryption

Name	Use and Function
AES GCM	Using arbitrary nonces for encryption
GHASH	MAC generation
Ed25519ctx	Signature generation/verification
HKDF	Using secrets shorter than 112 bits for key derivation; using a non-approved hash function for key derivation
HMAC	Using keys shorter than 112 bits for MAC generation; using a non-approved hash function for MAC generation
PBKDF	Using random salts shorter than 128 bits for key derivation; using a non-approved hash function for key derivation
TLS KDF	Using a non-approved hash function for key derivation
SSH KDF	Using a non-approved hash function for key derivation
RSA	Using key sizes smaller than 2048 bits or with odd bit length for key generation/verification, signature generation/verification, key encapsulation/decapsulation; using a non-approved hash function for signature generation/verification, key encapsulation/decapsulation; using public exponent smaller than 65537 for signature generation/verification, key encapsulation/decapsulation; primitive operation without padding; using non-approved DRBG for signature generation, key decapsulation; using primes of different lengths for signature generation, key decapsulation; using PSS salt longer than hash output for signature generation/verification
ECDH	Using non-approved DRBG for shared secret computation
ECDSA	Using non-approved DRBG for key/signature generation; using a non-approved hash function for signature generation/verification

Table 8: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Security function implementations (SFIs) are defined by the table below. The module is a software library, therefore the SFIs map directly to the module's services. Refer also to Security Policy Section 4.3 - Approved Services for a description of the module's services and SSP access.

Name	Type	Description	Properties	Algorithms
Data Encryption	BC-Auth BC-UnAuth	Symmetric Encryption	Publication:FIPS 197	AES-CBC: (A6650) AES-CTR: (A6650) AES-GCM: (A6650)

Name	Type	Description	Properties	Algorithms
Key Derivation Function	KAS-135KDF KAS-56CKDF PBKDF	Key Derivation Function	Publication 1:SP80056Cr1 Publication 2:SP80056Cr2 Publication 3:SP800-108	KDA HKDF Sp800-56Cr1: (A6650) KDA OneStepNoCounter SP800-56Cr2: (A6650) KDF SP800-108: (A6650) KDF SSH: (A6650) PBKDF: (A6650) TLS v1.2 KDF RFC7627: (A6650) TLS v1.3 KDF: (A6650)
Deterministic Random Bit Generation	DRBG	Deterministic Random Bit Generation	Publication:SP800-90Ar1	Counter DRBG: (A6650) HMAC DRBG: (A6650)
Digital Signature	DigSig-SigGen DigSig-SigVer	RSA, ECDSA SigGen/SigVer, EDDSA SigGen/SigVer	Publication:FIPS186-5	Deterministic ECDSA SigGen (FIPS186-5): (A6650) ECDSA SigGen (FIPS186-5): (A6650) ECDSA SigVer (FIPS186-5): (A6650) EDDSA SigGen: (A6650) EDDSA SigVer: (A6650) RSA SigGen (FIPS186-5): (A6650) RSA SigVer (FIPS186-5): (A6650)
Message Authentication	MAC	Generate or verify data integrity	Publication 1:SP800-38B	AES-CMAC: (A6650) HMAC-SHA2-224: (A6650) HMAC-SHA2-256:

Name	Type	Description	Properties	Algorithms
			Publication 2:FIPS 198-1	(A6650) HMAC-SHA2-384: (A6650) HMAC-SHA2-512: (A6650) HMAC-SHA2-512/224: (A6650) HMAC-SHA2-512/256: (A6650) HMAC-SHA3-224: (A6650) HMAC-SHA3-256: (A6650) HMAC-SHA3-384: (A6650) HMAC-SHA3-512: (A6650)
Key Agreement ECC	KAS-SSC	Perform key agreement primitives on behalf of the calling process (does not establish keys into the module)	Security Strength:Key establishment methodology provides between 112 and 256 bits of encryption strength. Publication : NIST SP 800-56Arev3 IG : D.F - Scenario 2, Path (1)	KAS-ECC-SSC Sp800-56Ar3: (A6650)
Message digest	SHA	Hashing	Publication:FIPS 180-4	SHA2-224: (A6650) SHA2-256: (A6650) SHA2-384: (A6650) SHA2-512: (A6650) SHA2-512/224: (A6650) SHA2-512/256: (A6650) SHA3-224: (A6650) SHA3-256: (A6650) SHA3-384: (A6650) SHA3-512: (A6650)

Name	Type	Description	Properties	Algorithms
Key Generation	AsymKeyPair-DomPar AsymKeyPair-KeyGen AsymKeyPair-KeyVer AsymKeyPair-PubKeyVal	Key Generation (asymmetric)	Publication 1:NIST SP 800-133rev2 - Section 4 Publication 2:NIST SP 800-133rev2 - Section 6.2.2 Publication 3:NIST FIPS 203	ECDSA KeyGen (FIPS186-5): (A6650) ECDSA KeyVer (FIPS186-5): (A6650) EDDSA KeyGen: (A6650) EDDSA KeyVer: (A6650) KAS-ECC-SSC Sp800-56Ar3: (A6650) RSA KeyGen (FIPS186-5): (A6650) CKG: () Key Type: Symmetric ML-KEM KeyGen: (A6650)
Key Encapsulation	KTS-Encap	Key Encapsulation	Security Strength:Key establishment methodology provides between 112 and 256 bits of encryption strength.	KTS-IFC: (A6650) ML-KEM EncapDecap: (A6650)
Data Decryption	BC-Auth BC-UnAuth	Symmetric Encryption	Publication:FIPS 197	AES-CBC: (A6650) AES-CTR: (A6650) AES-GCM: (A6650)
Extendable Output Function	XOF	Output message with variable length	Publication:FIPS 202	cSHAKE-128: (A6650) cSHAKE-256: (A6650) SHAKE-128: (A6650) SHAKE-256: (A6650)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 HMAC key length (IG C.M Conformance)

Per FIPS 140-3 IG C.M, key lengths below 112 bits are only allowed for legacy use (i.e. verification only). The module does not allow these. However, HKDF uses HMAC key lengths of less than 112 bits for the salt, and PBKDF2 uses them for the password, which are allowed to be shorter.

2.7.2 PBKDF (IG D.N Conformance)

Per FIPS 140-3 IG D.N, keys derived from passwords can only be used in storage applications.

As indicated under CAVP certificate #A6650, the PBKDF function supports passwords from 8 to 64 bytes and iteration counts from 1 to 10,000. SP 800-132 Section 5.2 recommends a minimum iteration count of 1,000. Operators should select an appropriate password length and iteration count for their use case, bearing in mind that both should be as large as is feasible for the application.

2.7.3 TLS 1.2 KDF (IG D.Q Conformance)

As indicated under CAVP certificate #A6650, the module supports TLS 1.2 KDF per RFC 7627, i.e. using the extended master secret.

2.7.4 Component Algorithm Validations (IG 2.4.B Conformance)

As indicated under CAVP certificate #A6650, the module supports the following components of approved algorithms, which are documented as CVLs.

KDFs from SP 800-135r1 are supported per IG 2.4.B resolution 6 for the following protocol KDFs: TLS v1.2 KDF RFC7627, and KDF SSH. Operator guidance: these SP 800-135r1 KDFs shall only be used in the context of the specified protocols (e.g. the TLS v1.2 KDF shall only be used within the context of the TLS 1.2 protocol).

The TLS 1.3 KDF from Section 7.1 of RFC 8446 is supported per IG 2.4.B resolution 7. Operator guidance: the TLS 1.3 KDF (CVL) shall only be used within the context of the TLS 1.3 protocol.

2.8 RBG and Entropy

Entropy Certificates:

N/A for this module.

Entropy Sources:

N/A for this module.

The module's entropy source is located within the TOEPP, but outside the cryptographic boundary of the module. The module passively receives entropy based on request by the calling applications and exercises no control over the amount or quality of the obtained entropy. The module aligns with IG 9.3.A, scenario 2b, therefore the module's certificate includes the caveat "No assurance of the minimum strength of generated SSPs (e.g., keys)."

2.9 Key Generation

When generating asymmetric keys, the module uses the output of CTR_DRBG to generate random numbers and keying material, per the guidance in NIST SP 800-133rev2, Section 5. ECDSA SigGen additionally generates the nonce according to FIPS 186-5, Appendix A.3.2 using a per-invocation instance of HMAC DRBG seeded with 256, 384, or 528 bits (for P-256, P-384, and P-521 respectively, per SP 800-90Ar1, Section 8.6.7) of entropy input from the global DRBG according to the IETF draft-irtf-cfrg-det-sigs-with-noise-03 document. This is analogous to Deterministic ECDSA specified in FIPS 186-5.

2.10 Key Establishment

The module implements the following approved key agreement method:

- KAS-ECC-SSC – NIST SP 800-56A Rev. 3 (FIPS 140-3 IG D.F, Scenario 2, Path (1))

While the module implements the protocol-specific KDFs in support of key agreement operations, the module only offers cryptographic services at the API level to calling applications and does not implement full key agreement within the module boundary.

The module implements the following key transport method:

- KTS-IFC – NIST SP 800-56B Rev. 2 (FIPS 140-3 IG D.G, Key Encapsulation/Decapsulation)

The module implements the following KEM method:

- ML-KEM – NIST FIPS 203 (FIPS 140-3 IG D.S, Key Encapsulation Methods)

2.11 Industry Protocols (IG D.C)

The module implements approved cryptography to support the following industry-specific protocols:

- SSH
- TLS v1.2
- TLS v1.3

These KDFs are implemented per SP 800-135r1 (Recommendation for Existing Application-Specific Key Derivation Functions) and RFC 8446. These KDFs have been validated by the CAVP and received CVL certificate #A6650.

The module only offers cryptographic services to calling applications and does not contain full implementations of industry protocols.

No parts of the SSH or TLS protocol, other than the approved cryptographic algorithms and the KDFs, have been tested by the CAVP and CMVP.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

As a software cryptographic toolkit, all of the module's interfaces are defined at the Software/Firmware Module Interface (SFMI). The logical interfaces are mapped to specific calls via a well-defined API, with which the operator interacts. The API provides functions that may be called by a host application (refer to Security Policy Section 4.3 - Approved Services).

Physical Port	Logical Interface(s)	Data That Passes
N/A	Data Input	The module accepts data input through the input arguments of the API functions.
N/A	Data Output	The module produces data output through the parameters of the API functions.
N/A	Control Input	The module accepts control input through the input arguments of the API functions used to control the module.
N/A	Status Output	The module produces status output through the return values from function calls and error messages.

Table 10: Ports and Interfaces

3.2 Additional Information

All interfaces are logically separated by the module's API.

The data output interface is inhibited when the module is performing self-tests, performing zeroisation, or while in an error state.

4 Roles, Services, and Authentication

4.1 Authentication Methods

N/A for this module.

The module does not implement an authentication mechanism. The operator role of Crypto Officer is assumed implicitly.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Crypto Officer	Role	Crypto Officer	None

Table 11: Roles

Crypto Officer (CO) is the only role supported by the module. The module does not support a User role or a Maintenance role. The Crypto Officer role is assumed implicitly by the operator when calling the module's services.

4.3 Approved Services

The following table describes the services the module provides and the access to SSPs by each service. Additional details on each service are available in the module's user guidance documentation.

SSP Access is divided into the following access types:

- Generate: The module generates or derives the SSP.
- Read: The SSP is read from the module (e.g. the SSP is output).
- Write: The SSP is updated, imported, or written to the module.
- Execute: The module uses the SSP in performing a cryptographic operation.
- Zeroise: The module zeroises the SSP.

For approved services, `fips140.ServiceIndicator()` will return true.

The operator can obtain the identifier output and versioning output for the module using the method specified in Security Policy Section 11.1 - Installation, Initialization, and Startup Procedures. This output shows the module version (v1.0.0) and module identifier (c2097c7c), which correspond to this validation record.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Data Encryption, Decryption	Encrypt or decrypt data	ServiceIndicator() returns true	Parameters, plaintext or ciphertext, key	Status, ciphertext or plaintext	Data Encryption Data Decryption	Crypto Officer - AES Key: W,E - AES GCM IV: W,E
Key Derivation Function	Perform key derivation using a key derivation function	ServiceIndicator() returns true	Parameters, key/password	Status, derived key	Key Derivation Function	Crypto Officer - KDF Derived Key: G,R - TLS Pre-Master Secret: G,R - TLS Master Secret: G,R - PBKDF Password: W,E - PBKDF Derived Key: G,R
Deterministic Random Bit Generation	Generate random numbers with SP800-90A Rev 1	ServiceIndicator() returns true	Parameters, entropy	Status, random number	Deterministic Random Bit Generation	Crypto Officer - Entropy Input: W,E,Z - DRBG 'V' Value: W,E - DRBG 'K' Value: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- DRBG Seed: G,E,Z - HMAC DRBG Entropy: W,E
Digital Signature	Generate or verify RSA or ECDSA digital signatures	ServiceIndicator() returns true	Parameters, RSA / ECDSA keys, message	Status, digital signature	Digital Signature	Crypto Officer - RSA Public Key: W,E - RSA Private Key: W,E - ECDSA Public Key: W,E - ECDSA Private Key: W,E
Message Authentication	Generate or verify data integrity	ServiceIndicator() returns true	Parameters, message, key	Status, message authentication code	Message Authentication	Crypto Officer - AES GCM Key: W,E - AES CMAC Key: W,E - HMAC Key: W,E
Key Agreement	Perform key agreement primitives on behalf of the calling process (does not	ServiceIndicator() returns true	Parameters, ECDH keys	Status, shared secret	Key Agreement ECC	Crypto Officer - ECDH Peer Public Key: W,E - ECDH Private Key: W,E - ECDH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	establish keys into the module)					Shared Secret: G,R
Key Generation	Generate and verify an asymmetric keypair	ServiceIndicator() returns true	Parameters, ECDH/ECDSA/EDDSA/ RSA keys	Status, keypair	Key Generation	Crypto Officer - RSA Public Key: G,R - RSA Private Key: G,R - ECDSA Public Key: G,R - ECDSA Private Key: G,R - EDDSA Private Key: G,R - EDDSA Public Key: G,R - ECDH Public Key: G,R - ECDH Private Key: G,R
Key Transport	Transport CSPs	ServiceIndicator() returns true	Parameters, plaintext or ciphertext key, transport key(s)	Status, plaintext or ciphertext key	Key Encapsulation	Crypto Officer - Key Transport Key: W,E
Message Digest	Generate a message digest	ServiceIndicator() returns true	Parameters, Message	Status, Digest of the message	Message digest	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Zeroize	Zeroize all SSPs	N/A	N/A	N/A	None	Crypto Officer - AES Key: Z - AES GCM IV: Z - AES GCM Key: Z - AES CMAC Key: Z - HMAC Key: Z - Entropy Input: Z - DRBG 'K' Value: Z - DRBG 'V' Value: Z - DRBG Seed: Z - RSA Public Key: Z - RSA Private Key: Z - ECDSA Public Key: Z - ECDSA Private Key: Z - ECDH Public Key: Z - ECDH Private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Key: Z - ECDH Shared Secret: Z - Key Transport Key: Z - KDF Secret: Z - KDF Derived Key: Z - TLS Pre-Master Secret: Z - TLS Master Secret: Z - PBKDF Password: Z - PBKDF Derived Key: Z - EDDSA Private Key: Z - EDDSA Public Key: Z - ECDH Peer Public Key: Z - HMAC DRBG Entropy: Z
Self-tests	Perform the	ServiceIndicator() returns true	N/A	Status	None	Crypto Officer

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	module self-tests					
Show Status	Show status of the module	ServiceIndicator() returns true	N/A	Status	None	Crypto Officer
Show module's versioning information	Show module's versioning information	Module version	N/A	Status	None	Crypto Officer
Extendable Output Function	Output message with desired length	ServiceIndicator() returns true	Parameters, message, desired length	Status, message with desired length	Extendable Output Function	Crypto Officer

Table 12: Approved Services

4.4 Non-Approved Services

The module provides the following non-approved services.

Upon invocation of any non-approved services, `fips140.ServiceIndicator()` will return false until reset, and the module will exit the approved mode of operation.

Name	Description	Algorithms	Role
Data Encryption, Decryption	Encrypt or decrypt data	AES ECB	Crypto Officer
Data Encryption	Encrypt data	AES GCM	Crypto Officer
Message Authentication	Generate MAC	GHASH HMAC	Crypto Officer
Digital Signature	Generate or verify digital signatures	Ed25519ctx RSA ECDSA	Crypto Officer

Name	Description	Algorithms	Role
Key Derivation Function	Perform key derivation using a key derivation function	HKDF PBKDF TLS KDF SSH KDF	Crypto Officer
Key Agreement	Perform key agreement primitives on behalf of the calling process	ECDH	Crypto Officer

Table 13: Non-Approved Services

4.5 External Software/Firmware Loaded

Not applicable for this module.

5 Software/Firmware Security

5.1 Integrity Techniques

As specified in the Tested Module Identification table in Security Policy Section 2.2 - Tested and Vendor Affirmed Module Version and Identification, the module implements integrity techniques for all executable code sets. The integrity technique used by the module is HMAC-SHA-256. The integrity technique has received CAVP certificate #A6650. The integrity technique is implemented by the module itself.

The module applies HMAC-SHA2-256 to all components of the `fips.o` binary and compares it to the expected value embedded into the binary.

The module's pre-operational integrity check is performed automatically at module power-up.

5.2 Initiate on Demand

The module integrity check can be performed on demand by the module operator by rebooting the host platform or by reinitializing the module.

5.3 Open-Source Parameters

The source code of the module is part of the latest release of the open-source Go Programming Language project distributed under a BSD-style license.

The module can be built using the Go toolchain which is also part of the Go Programming Language distribution. The version of the Go toolchain used to build the module should be the same as the one of the Go Programming Language distribution which contains the `lib/fips140/v1.0.0-c2097c7c.zip` file.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied:

Supported operational environments are indicated in Security Policy Section 2.2 - Tested and Vendor Affirmed Module Version and Identification. Refer also to that section for vendor affirmed operating environment porting guidance.

Operating systems provide process/memory isolation to prevent unauthorized access to CSPs and uncontrolled modifications of SSPs while the module is in process space. The module does not spawn processes or persistently store SSPs.

6.2 Configuration Settings and Restrictions

The module must be installed, and the correct installation confirmed, as described in Security Policy Section 11.1 - Installation, Initialization, and Startup Procedures.

7 Physical Security

The requirements of this section are not applicable to the module. The module is a software module and does not implement any physical security mechanisms.

8 Non-Invasive Security

The requirements of this section are not applicable to the module.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	System Memory	Dynamic
External	Calling Application	Dynamic

Table 14: Storage Areas

The table above lists the SSP storage areas implemented by the module. The module does not implement persistent storage of SSPs, and only stores keys temporarily in RAM, within the process space of the module.

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
SSP Input	External	RAM	Plaintext	Manual	Electronic	
SSP Output	RAM	External	Plaintext	Manual	Electronic	

Table 15: SSP Input-Output Methods

The information in the table above aligns with IG 9.5.A. IG 9.5.A indicates that SSPs established by a software cryptographic module to or from a general purpose application that operates outside the module's boundary but within the TOEPP are classified as Manually Distributed using Electronic Entry.

The module does not support any other methods of SSP input or output. Specifically, the module does not support Automated Distribution, Wireless Distribution, or Direct Entry.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Reboot Host	Reboot	Memory is zeroized upon reboot	Operator Initiated (Procedural)
Module Unload	Module Unload	Module unloaded from memory	Operator Initiated (Procedural)

Table 16: SSP Zeroization Methods

The module does not persistently store keys; therefore, all keys are held in the module's process space in volatile memory and are effectively zeroised upon reboot of the host platform.

9.4 SSPs

The following two tables define the module's Sensitive Security Parameters (SSPs). Access to SSPs is defined under Security Policy Section 4.3 - Approved Services.

9.4.1 SSPs (Table 1 of 2)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES Key	AES Key (CBC, CTR, GCM)	128,192, 256 - 128,192, 256	Symmetric Key - CSP	Deterministic Random Bit Generation		Data Encryption Data Decryption
AES GCM IV	AES-GCM Initialization Vector	96 bits -	Key Component - CSP	Deterministic Random Bit Generation		Data Encryption Data Decryption
AES GCM Key	AES-GCM Key	128,192, 256 - 128,192, 256	Symmetric Key - CSP	Key Derivation Function		Data Encryption Data Decryption
AES CMAC Key	CMAC Key	128, 192, 256 - 128, 192, 256	Symmetric Key - CSP	Deterministic Random Bit Generation		Data Encryption Data Decryption
HMAC Key	Key used for HMAC Operations	>= 112 bits - >= 112 bits	HMAC Key - CSP	Deterministic Random Bit Generation		Message Authentication
Entropy Input	Externally generated entropy used to seed the DRBG	512 bits - 512 bits	Entropy - CSP			Deterministic Random Bit Generation
DRBG 'K' Value	CTR-DRBG Internal Value	256 bits - 256 bits	DRBG Internal State - CSP			Deterministic Random Bit Generation

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
DRBG 'V' Value	CTR-DRBG Internal Value	128 bits - 128 bits	DRBG Internal State - CSP			Deterministic Random Bit Generation
DRBG Seed	DRBG Internal State Value	384 bits - 384 bits	DRBG Internal State - CSP			Deterministic Random Bit Generation
HMAC DRBG Entropy	Input to HMAC DRBG	256 - 528 bits - 256 - 528 bits	Entropy - CSP	Deterministic Random Bit Generation		Deterministic Random Bit Generation
RSA Public Key	Public Key used for RSA operations	>= 2048 bits - >= 112 bits	RSA Keypair - PSP	Deterministic Random Bit Generation		Digital Signature
RSA Private Key	Private Key used for RSA operations	>= 2048 bits - >= 112 bits	RSA Keypair - CSP	Deterministic Random Bit Generation		Digital Signature
ECDSA Public Key	Public Key used for EC operations	224 to 521 bits - 112 to 256 bits	EC Keypair - PSP	Deterministic Random Bit Generation		Digital Signature
ECDSA Private Key	Private Key used for EC operations	224 to 521 bits - 112 to 256 bits	EC Keypair - CSP	Deterministic Random Bit Generation		Digital Signature
ECDH Public Key	ECDH Public Key	224 to 521 bits - 112 to 256 bits	ECDH Keypair - PSP	Deterministic Random Bit Generation		Key Agreement ECC
ECDH Private Key	ECDH Private Key	224 to 521 bits - 112 to 256 bits	ECDH Keypair - CSP	Deterministic Random Bit Generation		Key Agreement ECC
ECDH Shared Secret	ECDH Shared Secret	224 to 521 bits - 112 to 256 bits	ECDH Shared Secret - CSP		Key Agreement ECC	Key Derivation Function

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
EDDSA Public Key	EDDSA Public Key	255 bits - 128 bits	EDDSA Keypair - PSP	Deterministic Random Bit Generation		Digital Signature
EDDSA Private Key	EDDSA Private Key	253 bits - 128 bits	EDDSA Keypair - CSP	Deterministic Random Bit Generation		Digital Signature
Key Transport Key	Key Transport Key	>= 2048 bits - >= 112 bits	Key Transport - CSP	Key Derivation Function		Key Encapsulation
KDF Secret	Secret used for KDF operations	>= 112 bits - >= 112 bits	Keying material - CSP		Key Agreement ECC	Key Derivation Function
KDF Derived Key	Key resulting from the module's SP 800-135rev1 KDFs	>= 112 bits - >= 112 bits	KDF Key - CSP	Key Derivation Function		
TLS Pre-Master Secret	Shared Secret used for TLS session establishment	112 to 256 bits - 112 to 256 bits	KDF Secret - CSP		Key Agreement ECC	Key Derivation Function
TLS Master Secret	Master Secret used for TLS session	112 to 256 bits - 112 to 256 bits	KDF Secret - CSP	Key Derivation Function		Data Encryption
PBKDF Password	Password used for password based key derivation	- - -	KDF Key - CSP	Deterministic Random Bit Generation		Key Derivation Function
PBKDF Derived Key	Key resulting from the module's PBKDF	>= 112 bits - >= 112 bits	Symmetric Key - CSP	Key Derivation Function		Key Derivation Function
ECDH Peer Public Key	ECDH Peer Public Key	224 to 521 bits - 112 - 256 bits	Asymmetric - PSP		Key Agreement ECC	KAS-ECC-SSC Sp800-56Ar3 (A6650)

Table 17: SSP Table 1

9.4.2 SSPs (Table 2 of 2)

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
AES GCM IV	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
AES GCM Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	AES GCM IV:Derived From
AES CMAC Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
HMAC Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
Entropy Input	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
DRBG 'K' Value	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	DRBG 'V' Value:Used With
DRBG 'V' Value	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	DRBG 'K' Value:Used With
DRBG Seed	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
HMAC DRBG Entropy	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
RSA Public Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	RSA Private Key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
RSA Private Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	RSA Public Key:Paired With
ECDSA Public Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDSA Private Key:Paired With
ECDSA Private Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDSA Public Key:Paired With
ECDH Public Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDH Private Key:Paired With
ECDH Private Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDH Public Key:Paired With
ECDH Shared Secret	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDH Public Key:Used With ECDH Private Key:Used With
EDDSA Public Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDSA Private Key:Paired With
EDDSA Private Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	ECDSA Public Key:Paired With
Key Transport Key	SSP Input SSP Output	RAM:Encrypted External:Encrypted	Until Reboot	Reboot Host Module Unload	
KDF Secret	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	
KDF Derived Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	KDF Secret:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLS Pre-Master Secret	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	TLS Master Secret:Used With
TLS Master Secret	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	TLS Pre-Master Secret:Derived From
PBKDF Password	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	PBKDF Salt:Used With
PBKDF Derived Key	SSP Input SSP Output	RAM:Plaintext External:Plaintext	Until Reboot	Reboot Host Module Unload	PBKDF Password:Derived From PBKDF Salt:Derived From
ECDH Peer Public Key	SSP Input SSP Output	RAM:Plaintext	Until Reboot	Reboot Host Module Unload	ECDH Private Key:Used With

Table 18: SSP Table 2

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-256 (A6650)	HMAC-SHA2-256	KAT	SW/FW Integrity	status output	Verify the integrity of the module

Table 19: Pre-Operational Self-Tests

The module only performs one pre-operational self-test, which is the software/firmware integrity test. The module does not implement any other pre-operational self-tests, including pre-operational self-tests for bypass or critical functions, because the module does not implement corresponding functions.

The pre-operational self-tests are always performed on module instantiation and the HMAC-SHA2-256 CAST is performed before the integrity test, and therefore before the first operational use of the algorithm. If the self-tests fail, the module terminates.

10.2 Conditional Self-Tests

The module mainly performs two types of conditional self-tests, which are Cryptographic Algorithm Self-Tests (CASTs) and Pairwise Consistency Tests (PCTs). The module does not implement any other conditional self-tests, including conditional self-tests for critical functions, software/firmware loading, manual entry, or bypass, because the module does not implement corresponding functions.

Conditional CASTs are performed at module initialization, after the module integrity test, and before the first operational use of the algorithms. Pairwise consistency tests are performed conditionally upon generation of an asymmetric keypair.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KAS-ECC-SSC Sp800-56Ar3 PCT	P-256	PCT	PCT	status output	KAS-ECC-SSC P-256, ECDH public key computation	Private key generation
ECDSA KeyGen (FIPS186-5) (A6650)	P-256, SHA2-512	PCT	PCT	status output	ECDSA sign and verify PCT	ECDSA private key generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
EDDSA KeyGen (A6650)	Ed25519	PCT	PCT	status output	ECDSA sign and verify PCT	EDDSA private key generation
ML-KEM KeyGen (A6650)	ML-KEM-768, ML-KEM-1024	PCT	PCT	status output	ML-KEM encap and decap PCT	ML-KEM-768/ML-KEM-1024 encapsulation key generation
RSA KeyGen (FIPS186-5) (A6650)	2048	PCT	PCT	status output	RSA sign and verify PCT	RSA private key generation
KAS-ECC-SSC Sp800-56Ar3 (A6650)	P256	KAT	CAST	status output	KAS-ECC-SSC P-256, ECDH secret key computation	Secret key computation
ECDSA SigGen (FIPS186-5) (A6650)	P256, SHA2-512	KAT	CAST	status output	ECDSA P-256 SHA2-512 sign and verify	Signature generation and signature verification
Deterministic ECDSA SigGen (FIPS186-5) (A6650)	P256, SHA2-512	KAT	CAST	status output	DetECDSA P-256 SHA2-512 sign	DetECDSA/ECDSA sign with HMAC-DRBG
EDDSA SigGen (A6650)	Ed25519, HashEd25519	KAT	CAST	status output	Ed25519 sign and verify	Signature generation and signature verification
RSA SigGen (FIPS186-5) (A6650)	2048 bits	KAT	CAST	status output	RSASSA-PKCS-v1.5 2048-bit sign and verify	Signature generation and signature verification
AES-CBC (A6650)	Key size: 128 bits	KAT	CAST	status output	AES-CBC encrypt and decrypt	First time calling AES-CBC or AES-CTR
KDF SP800-108 (A6650)	Counter KDF AES-CMAC Key size: 128 bits	KAT	CAST	status output	AES-128 CMAC Counter KDF	First time calling AES-GCM or AES-CMAC or Counter KDF
Counter DRBG (A6650)	Entropy input: 384 bits	KAT	CAST	status output	ctrDRBG AES-256 KAT	First time calling Counter DRBG

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
cSHAKE-128 (A6650)	Security strength: 128 bits	KAT	CAST	status output	cSHAKE128 KAT	First time calling SHA3
HMAC DRBG (A6650)	hash: SHA2-512	KAT	CAST	status output	HMAC_DRBG SHA2-512 (Init, Reseed, Generate)	First time calling HMAC-DRBG API
TLS v1.3 KDF (A6650)	Extract: SHA2-256, Expand: SHA2-256	KAT	CAST	status output	TLS 1.3 KDF SHA2-256 KAT	First time calling TLS 1.3
TLS v1.2 KDF RFC7627 (A6650)	hash: SHA2-256	KAT	CAST	status output	TLS 1.2 RFC 7627 KDF SHA2-256 KAT	First time calling TLS 1.2
SHA2-512 (A6650)	Output size: 512 bits	KAT	CAST	status output	SHA2-512 KAT	First time calling SHA2-512
SHA2-256 (A6650)	Output size: 256 bits	KAT	CAST	status output	SHA2-256 KAT	First time calling SHA2-256
ML-KEM EncapDecap (A6650)	ML-KEM-768	KAT	CAST	status output	ML-KEM-768 KAT	First time calling ML-KEM
HMAC-SHA2-256 (A6650)	HMAC-SHA2-256	KAT	CAST	status output	HMAC-SHA2-256	First self-test that runs before module integrity check
HKDF Extract and Expand	HKDF Extract and Expand	KAT	CAST	status output	HKDF Extract and Expand	First time calling KDA HKDF, KDA OneStepNoCounter, or Feedback KDF
PBKDF (A6650)	SHA2-256	KAT	CAST	status output	HKDF2 KAT	First time calling PBKDF2

Table 20: Conditional Self-Tests

10.3 Periodic Self-Test Information

The module can perform the periodic pre-operational, and conditional self-tests procedurally by power cycling the host platform.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A6650)	KAT	SW/FW Integrity	User initiated module reinitialization	User initiated module reinitialization

Table 21: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 PCT	PCT	PCT	User initiated module reinitialization	User initiated module reinitialization
ECDSA KeyGen (FIPS186-5) (A6650)	PCT	PCT	User initiated module reinitialization	User initiated module reinitialization
EDDSA KeyGen (A6650)	PCT	PCT	User initiated module reinitialization	User initiated module reinitialization
ML-KEM KeyGen (A6650)	PCT	PCT	User initiated module reinitialization	User initiated module reinitialization
RSA KeyGen (FIPS186-5) (A6650)	PCT	PCT	User initiated module reinitialization	User initiated module reinitialization
KAS-ECC-SSC Sp800-56Ar3 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
ECDSA SigGen (FIPS186-5) (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
Deterministic ECDSA SigGen (FIPS186-5) (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
EDDSA SigGen (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA SigGen (FIPS186-5) (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
AES-CBC (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
KDF SP800-108 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
Counter DRBG (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
cSHAKE-128 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
HMAC DRBG (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
TLS v1.3 KDF (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
TLS v1.2 KDF RFC7627 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
SHA2-512 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
SHA2-256 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
ML-KEM EncapDecap (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-256 (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
HKDF Extract and Expand	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization
PBKDF (A6650)	KAT	CAST	User initiated module reinitialization	User initiated module reinitialization

Table 22: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error	The module's error state.	POST, PCT or CAST failure	Reinitialize module	fatal error

Table 23: Error States

The module terminates after any self-test failure, therefore no further requests to the module are available. To recover the module from error states, the host platform must be power cycled, or the module be reinitialized. Power cycling or module reinitialization will cause the module to perform the pre-operational self-tests and transition to the approved mode of operation.

10.5 Operator Initiation of Self-Tests

The module operator can initiate the pre-operational and conditional self-tests by power cycling the host platform or reinitializing the module.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

The module `fips.o` can be obtained by building the source code in the `lib/fips140/v1.0.0-c2097c7c.zip` archive, included in the latest Go source distribution available at https://go.dev/dl.

The archive can be verified by issuing the command

```
sha256sum lib/fips140/v1.0.0-c2097c7c.zip
```

and ensuring the SHA2-256 digest matches

```
daf3614e0406f67ae6323c902db3f953a1effb199142362a039e7526dfb9368b
```

The set of files in the archive verified as specified above is the complete set of source files of the validated module, and it shall be used unmodified in the build process.

Once the zip file has been obtained and verified, and the latest version of Go for your distribution was obtained and installed (for bootstrap), the following steps (or equivalent) can be executed to compile it as part of an application (at `$SRCROOT`, with the Go source distribution at `$GOROOT`).

```
cd "$GOROOT/src"
./make.bash
cd "$SRCROOT"
GOFIPS140=v1.0.0 "$GOROOT/bin/go" build # or equivalent
```

Alternatively, the GOFIPS140 variable can be specified when building the toolchain:

```
cd "$GOROOT/src"
GOFIPS140=v1.0.0 ./make.bash
cd "$SRCROOT"
"$GOROOT/bin/go" build # or equivalent
```

(On Windows, `make.bat` is used to build the toolchain instead of `make.bash`.)

Once the application (at `$APPBIN`) is compiled, the status of the module can be verified with the following command (or equivalent).

```
"$GOROOT/bin/go" version -m "$APPBIN"
```

If the output includes a line matching the following, the module has been configured correctly and can be operated in approved mode.

```
build GOFIPS140=v1.0.0-c2097c7c
```

11.2 Administrator Guidance

Additional administrator guidance is provided separately in other operator documentation, including at https://go.dev/doc/.

11.3 Non-Administrator Guidance

Additional non-administrator guidance is provided separately in other operator documentation, including at https://go.dev/doc/.

11.4 Design and Rules

The module is designed to meet the applicable requirements of FIPS 140-3. The module initializes when powered on, and performs the pre-operational self-tests and CASTs as specified in Security Policy Section 10 - Self-Tests.

11.5 End of Life

The module can be removed by deleting the module binary from the host platform.

12 Mitigation of Other Attacks

The module implements the following mitigations:

- During RSA signature generation processes, countermeasures are implemented to protect against vulnerabilities related to the Chinese Remainder Theorem (CRT), such as those exploited by Lenstra's attack.
- The ECDSA scheme implements mitigation against both RNG failures (like Deterministic ECDSA) and fault attacks (like the RSA CRT attack mitigation).