



Hewlett Packard
Enterprise

HPE Aruba Networking VIA Cryptographic Module

Software Version 1.1

Non-Proprietary Security Policy

FIPS 140-3 Level 1

Document Version 1.0
June 2026

Copyright

© 2026 Hewlett Packard Enterprise Company. Hewlett Packard Enterprise Company trademarks include  , HPE Wireless Networks, HPE Networking, the registered HPE Networking the Mobile Edge Company logo, HPE Networking Mobility Management System®, Mobile Edge Architecture®, People Move. Networks Must Follow®, RFProtect®, Green Island®. All rights reserved. All other trademarks are the property of their respective owners. HPE Networking is a Hewlett Packard Enterprise company.

The resource assets in this firmware may include abbreviated and/or legacy terminology for HPE Networking products. See <https://www.hpe.com/us/en/networking/> for current and complete HPE Networking product lines and names.

Open Source Code

Certain Hewlett Packard Enterprise Company products include Open Source software code developed by third parties, including software code subject to the GNU General Public License (GPL), GNU Lesser General Public License (LGPL), or other Open Source Licenses. The Open Source code used can be found at this site:

<https://myenterpriselicense.hpe.com/cwp-ui/software>

Legal Notice

The use of Hewlett Packard Enterprise Company switching platforms and software or firmware, by all individuals or corporations, to terminate other vendors' VPN client devices constitutes complete acceptance of liability by that individual or corporation for this action and indemnifies, in full, Hewlett Packard Enterprise Company, from any and all legal actions that might be taken against it with respect to infringement of copyright on behalf of those vendors.



Hewlett Packard Enterprise

<https://www.hpe.com/us/en/networking/>

1701 E Mossy Oaks Rd,
Spring, TX, USA 77389

Phone: 1-888-342-2156

Table of Contents

1	General.....	10
1.1	Overview	10
1.2	Security Levels.....	10
1.3	Additional Information.....	11
1.4	Acronyms and Abbreviations	11
2	Cryptographic Module Specification.....	14
2.1	Description	14
2.1.1	Purpose and Use	14
2.1.2	Cryptographic Boundary	14
2.1.3	Tested Operational Environment's Physical Perimeter (TOEPP)	14
2.2	Tested and Vendor Affirmed Module Version and Identification	15
2.2.1	Executable Code Sets	15
2.2.2	Tested Operational Environments	15
2.2.3	Vendor-Affirmed Operational Environments	16
2.3	Excluded Components.....	16
2.4	Modes of Operation.....	16
2.5	Algorithms.....	17
2.5.1	Approved Algorithms	17
2.5.2	Vendor-Affirmed Algorithms.....	19
2.6	Security Function Implementations	19
2.7	Algorithm Specific Information	22
2.7.1	AES-GCM	22
2.7.2	SHA-1	22
2.7.3	RSA	22
2.7.4	ECDSA.....	22
2.8	RBG and Entropy	22
2.9	Key Generation.....	22
2.10	Key Establishment.....	23
2.11	Industry Protocols	23
3	Cryptographic Module Interfaces	24
3.1	Ports and Interfaces	24
4	Roles, Services, and Authentication.....	25

4.1	Authentication Methods	25
4.2	Roles	25
4.3	Approved Services	26
4.3.1	Show version	29
4.3.2	Show status	29
4.4	Non-Approved Services	29
4.5	External Software Loaded	29
5	Software Security	30
5.1	Integrity Techniques	30
5.2	Initiate on Demand	30
6	Operational Environment	31
6.1	Operational Environment Type and Requirements	31
6.2	Configuration settings and restrictions	31
7	Physical Security	32
8	Non-Invasive Security	33
9	Sensitive Security Parameters Management	34
9.1	Storage Areas	34
9.2	SSP Input-Output Methods	34
9.3	SSP Zeroization Methods	34
9.4	SSPs	35
9.5	Transitions	38
10	Self-Tests	39
10.1	Pre-Operational Self-Tests	39
10.2	Conditional Self-Tests	40
10.3	Periodic Self-Test Information	41
10.4	Error States	41
10.5	Operator Initiation of Self-Tests	42
10.6	Additional Information	42
11	Life-Cycle Assurance	43
11.1	Installation, Initialization, and Startup Procedures	43
11.2	Administrator Guidance	43

11.2.1	Zeroization	43
11.2.2	AES-GCM	43
11.3	Non-Administrator Guidance	43
11.6	End of Life.....	45
12	Mitigation of Other Attacks	46

List of Tables

Table 1: Security Levels.....	11
Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets).....	15
Table 3: Tested Operational Environments - Software, Firmware, Hybrid.....	16
Table 4: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid	16
Table 5: Modes List and Description	16
Table 6: Approved Algorithms	19
Table 7: Vendor-Affirmed Algorithms	19
Table 8: Security Function Implementations.....	21
Table 9: Ports and Interfaces	24
Table 10: Roles.....	25
Table 11: Approved Services	28
Table 12: Storage Areas	34
Table 13: SSP Input-Output Methods	34
Table 14: SSP Zeroization Methods	34
Table 15: SSP Table 1	35
Table 16: SSP Table 2	37
Table 17: Pre-Operational Self-Tests	39
Table 18: Conditional Self-Tests	40
Table 19: Pre-Operational Periodic Information	41
Table 20: Conditional Periodic Information.....	41
Table 21: Error States	41

List of Figures

Figure 1: Block Diagram	15
-------------------------------	----

Preface

This document may be freely reproduced and distributed whole and intact including the copyright notice. Products identified herein contain confidential commercial firmware. Valid license required.

Document Revision History

Version	Date	Description
1.0	June 2026	Initial FIPS 140-3 release for Hewlett Packard Enterprise HPE Aruba Networking VIA Crypto Module software version 1.1, a software library that provides cryptographic services required by the calling HPE Aruba Networking application (e.g., the HPE Aruba Networking Virtual Intranet Access (VIA) product).

1 General

This section describes:

- The purpose of this document.
- Hewlett Packard Enterprise documents related to this document contents.
- Where to go for additional Hewlett Packard Enterprise product information.
- Acronyms and abbreviations.
- The assurance security levels for each of the areas described in the FIPS 140-3 Standard.

1.1 Overview

This release supplement provides information regarding the Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* software version 1.1 FIPS 140-3 Level 1 validation from Hewlett Packard Enterprise (HPE). Throughout this document, references to HPE Aruba Networking are to the Hewlett Packard Enterprise division. The material in this supplement modifies the general Hewlett Packard Enterprise software documentation included with this product and should be kept with your Hewlett Packard Enterprise product documentation.

This supplement primarily covers the non-proprietary Cryptographic Module Security Policy for the Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* software version 1.1. This security policy describes how the module meets the security requirements of FIPS 140-3 Level 1 and how to place and maintain the module in the secure Approved mode. This policy was prepared as part of the FIPS 140-3 Level 1 validation of the product.

FIPS 140-3 (Federal Information Processing Standards Publication 140-3, *Security Requirements for Cryptographic Modules*) details the U.S. Government requirements for cryptographic modules. FIPS 140-3 aligns with ISO/IEC 19790:2012(E) and includes modifications of the Annexes that are allowed to the Cryptographic Module Validation Program (CMVP), as a validation authority. The testing for these requirements will be in accordance with ISO/IEC 24759:2017(E), with the modifications, additions or deletions of vendor evidence and testing allowed as a validation authority under paragraph 5.2. More information about the FIPS 140-3 standard and validation program is available on the National Institute of Standards and Technology (NIST) website at:

<https://csrc.nist.gov/projects/cryptographic-module-validation-program>

In addition, in this document, the Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* is referred to as the module, the cryptographic module, and VIA crypto module.

1.2 Security Levels

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1

Section	Title	Security Level
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	N/A
	Overall Level	1

Table 1: Security Levels

1.3 Additional Information

More information is available from the following sources:

- See the Hewlett Packard Enterprise web site for the full line of products from HPE Networking:
<https://www.hpe.com/us/en/networking/>
- The NIST Validated Modules web site contains contact information for answers to technical or sales-related questions for the product:
<https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search>

Enter Hewlett Packard Enterprise in the Vendor field then select Search to see a list of FIPS validated Hewlett Packard Enterprise cryptographic modules.

Select the Certificate Number for the Module Name 'HPE Aruba Networking VIA Crypto Module'.

1.4 Acronyms and Abbreviations

AES	Advanced Encryption Standard
AP	Access Point
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCCS	Canadian Centre for Cyber Security, a branch of CSE
CLI	Command Line Interface

CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CPSec	Control Plane Security protected
CSE	Communications Security Establishment
CSP	Critical Security Parameter
DF	Derivation Function
EAP	Extensible Authentication Protocol
ECO	External Crypto Officer
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
ESV	Entropy Source Validation
FE	Fast Ethernet
GE	Gigabit Ethernet
GHz	Gigahertz
GPC	General Purpose Computer
GPOS	General Purpose Operating System
HMAC	Hashed Message Authentication Code
Hz	Hertz
IKE	Internet Key Exchange
IPsec	Internet Protocol security
KAT	Known Answer Test
KEK	Key Encryption Key
L2TP	Layer-2 Tunnelling Protocol
LAN	Local Area Network
LED	Light Emitting Diode
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
PCT	Pairwise Consistency Test
PSP	Public Security Parameter
SFTP	Secure File Transfer Protocol
SHA	Secure Hash Algorithm
SNMP	Simple Network Management Protocol
SSP	Sensitive Security Parameter
SPOE	Serial & Power Over Ethernet
TEL	Tamper-Evident Label

TFTP	Trivial File Transfer Protocol
TOEPP	Tested Operational Environment Physical Perimeter
TPM	Trusted Platform Module
VIA	Virtual Intranet Access
WLAN	Wireless Local Area Network

2 Cryptographic Module Specification

2.1 Description

2.1.1 Purpose and Use

The Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* version 1.1 (also referred to as ‘the module’) is a software type cryptographic module that was validated under FIPS 140-3 Level 1 requirements and meets the claims made in this document. The module provides cryptographic services to HPE Aruba Networking applications (e.g., the HPE Aruba Networking Virtual Intranet Access (VIA) product - out of scope of this validation) through an Application Program Interface (API).

Module Type: Software

Module Embodiment: Multi-Chip Standalone

2.1.2 Cryptographic Boundary

The module’s cryptographic boundary is defined as the shared dynamic library file. All operations of the module occur via calls from HPE Aruba Networking applications and their respective internal daemons/processes. The module is linked to the calling HPE Aruba Networking application and no interface is provided to the end user, so only functions within the module can call the cryptographic functions. The HPE Aruba Networking applications that use the API calls with the module are out of scope for this validation.

2.1.3 Tested Operational Environment’s Physical Perimeter (TOEPP)

The TOEPP is the production-grade enclosure of the hardware chassis of the General-Purpose Computer (GPC) where the module is installed (see Figure 1 block diagram below).

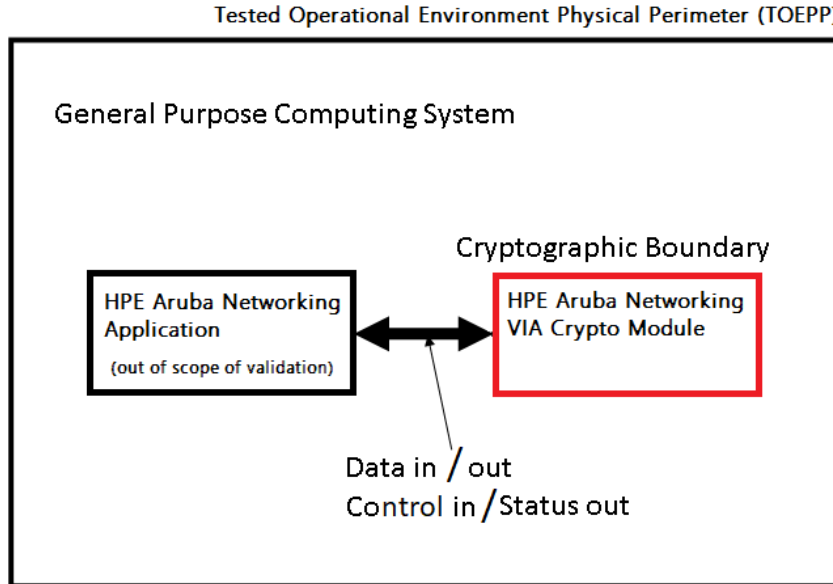


Figure 1: Block Diagram

2.2 Tested and Vendor Affirmed Module Version and Identification

2.2.1 Executable Code Sets

Package or File Name	Software/ Firmware Version	Features	Integrity Test
libancrypto.so (android)	1.1	Android	MAC verification
libancrypto.so (linux)	1.1	Linux	MAC verification
ancrypto.dll	1.1	Windows	MAC verification

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

2.2.2 Tested Operational Environments

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Android 16	Samsung Galaxy S24	Samsung Exynos 2400 (ARMv9.2-A)	No	-	1.1
Ubuntu 22.04 LTS	Dell Vostro 3500	Intel Core i3-1115G4 (Tiger Lake)	No	-	1.1

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Microsoft Windows 11 24 H2	HP EliteBook 840 G11	Intel Core Ultra 5 125U (Meteor Lake)	No	-	1.1

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

2.2.3 Vendor-Affirmed Operational Environments

Operating System	Hardware Platform
Android versions 9, 10, 11, 12, 13, 14, 15, 16	64-bit ARM
iOS versions 13 and above	64-bit ARM
macOS versions 11 and above	64-bit ARM and x86_64
RHEL/CentOS versions 8 and above	64-bit x86_64
Ubuntu 20.04 LTS and later	64-bit x86_64
Windows 10, 11	64-bit x86_64

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 components within the cryptographic boundary that are excluded from the module.

2.4 Modes of Operation

Modes List and Description

Mode Name	Description	Types of Services	Status Indicator
Approved Mode	Non-Approved logging messages are enabled when non-Approved services are invoked.	Approved	Global mode indicator: FIPS_isNonApprovedLoggingEnabled() returns true.

Table 5: Modes List and Description

2.5 Algorithms

2.5.1 Approved Algorithms

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A7716	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A7716	Direction - Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 128 Supports Counter larger than maximum value - No Incremental Counter - Yes Counter Tests Performed - No	SP 800-38A
AES-GCM	A7716	Direction - Decrypt, Encrypt IV Generation - External IV Generation Mode - 8.2.2 Key Length - 128, 192, 256 Tag Length - 128 IV Length - IV Length: 96 Payload Length - Payload Length: 128-264 Increment 8 AAD Length - AAD Length: 128-264 Increment 8	SP 800-38D
Counter DRBG	A7716	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - No Additional Input - Additional Input: 128-256 Increment 8, Additional Input: 192-256 Increment 8, Additional Input: 256-384 Increment 8 Entropy Input - Entropy Input: 256, Entropy Input: 320, Entropy Input: 384 Nonce - Nonce: 0 Personalization String Length - Personalization String Length: 128-256 Increment 8, Personalization String Length: 192-320 Increment 8, Personalization String Length: 256-384 Increment 8 Returned Bits - 128, 192, 256	SP 800-90A Rev. 1
DSA KeyGen (FIPS186-4)	A7716	L - 2048 N - 256	FIPS 186-4
DSA PQGGen (FIPS186-4)	A7716	P/Q Generation Methods - Probable G Generation Methods - Unverifiable L - 2048 N - 256 Hash Algorithm - SHA2-256	FIPS 186-4
ECDSA KeyGen (FIPS186-5)	A7716	Curve - P-256, P-384 Secret Generation Mode - testing candidates	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
ECDSA SigGen (FIPS186-5)	A7716	Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A7716	Component - No Curve - P-256, P-384 Hash Algorithm - SHA-1	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A7716	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A7716	MAC - MAC: 160 Key Length - Key Length: 192	FIPS 198-1
HMAC-SHA2-256	A7716	MAC - MAC: 256 Key Length - Key Length: 192, 256	FIPS 198-1
HMAC-SHA2-384	A7716	MAC - MAC: 384 Key Length - Key Length: 256, 384	FIPS 198-1
HMAC-SHA2-512	A7716	MAC - MAC: 512 Key Length - Key Length: 256, 384	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A7716	Domain Parameter Generation Methods - P-256, P-384 Hash Function Z - SHA2-384 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A7716	Domain Parameter Generation Methods - FC, MODP-2048 Hash Function Z - SHA2-384 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
RSA KeyGen (FIPS186-5)	A7716	Key Generation Mode - probable, probableWithProbableAux Modulo - 2048 Primality Tests - 2pow100, 2powSecStr Fixed Public Exponent - 010001 Info Generated By Server - Yes Private Key Format - standard Public Exponent Mode - fixed	FIPS 186-5
RSA SigGen (FIPS186-5)	A7716	Hash Pair - Hash Algorithm - SHA2-256	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
		Modulo - 2048 Signature Type - pkcs1v1.5	
RSA SigVer (FIPS186-4)	A7716	Signature Type - PKCS 1.5 Modulo - 1024, 2048 Hash Pair - Hash Algorithm - SHA-1 Public Exponent Mode - Fixed Fixed Public Exponent - 010001	FIPS 186-4
RSA SigVer (FIPS186-5)	A7716	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048 Signature Type - pkcs1v1.5 Fixed Public Exponent - 010001 Public Exponent Mode - fixed	FIPS 186-5
Safe Primes Key Generation	A7716	Safe Prime Groups - MODP-2048	SP 800-56A Rev. 3
SHA-1	A7716	Message Length - Message Length: 0-65536 Increment 8 Function - SHA1	FIPS 180-4
SHA2-256, SHA2-384, SHA2-512	A7716	Message Length - Message Length: 0-65536 Increment 8 Function - SHA2	FIPS 180-4

Table 6: Approved Algorithms

2.5.2 Vendor-Affirmed Algorithms

Name	Properties	Implementation	Reference
CKG	Key type:Symmetric and asymmetric	N/A	SP 800-133r2 section 4

Table 7: Vendor-Affirmed Algorithms

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Encrypt	BC-UnAuthEncrypt	Unauthenticated encryption		AES-CBC: (A7716) Direction: Encrypt AES-CTR: (A7716)

Name	Type	Description	Properties	Algorithms
				Direction: Encrypt
Decrypt	BC-UnAuthDecrypt	Unauthenticated decryption		AES-CBC: (A7716) Direction: Decrypt AES-CTR: (A7716) Direction: Decrypt
AuthEncrypt	BC-AuthEncrypt	Authenticated encryption		AES-GCM: (A7716) Direction: Encrypt
AuthDecrypt	BC-AuthDecrypt	Authenticated decryption		AES-GCM: (A7716) Direction: Decrypt
DRBG	DRBG	Random bit generation		Counter DRBG: (A7716) AES-CBC: (A7716)
SigGen	DigSig-SigGen	Signature generation		ECDSA SigGen (FIPS186-5): (A7716) RSA SigGen (FIPS186-5): (A7716) SHA2-256: (A7716) SHA2-384: (A7716) SHA2-512: (A7716)
SigVer	DigSig-SigVer	Signature verification		ECDSA SigVer (FIPS186-5): (A7716) RSA SigVer (FIPS186-5): (A7716) SHA2-256: (A7716) SHA2-384: (A7716) SHA2-512: (A7716)
Legacy SigVer	DigSig-SigVer	Signature verification	Modulo:1024 Hash:SHA-1	ECDSA SigVer (FIPS186-4): (A7716) RSA SigVer (FIPS186-4): (A7716) SHA-1: (A7716) SHA2-256: (A7716) SHA2-384: (A7716) SHA2-512: (A7716)
KeyGen	AsymKeyPair-KeyGen	Key pair generation		DSA KeyGen (FIPS186-4): (A7716) DSA PQGGen (FIPS186-4): (A7716) ECDSA KeyGen (FIPS186-5): (A7716) RSA KeyGen (FIPS186-5): (A7716) Safe Primes Key Generation: (A7716)
MAC	MAC	Generating or verifying a message authentication code		HMAC-SHA-1: (A7716) HMAC-SHA2-256: (A7716) HMAC-SHA2-384: (A7716) HMAC-SHA2-512: (A7716) SHA-1: (A7716) SHA2-256: (A7716)

Name	Type	Description	Properties	Algorithms
				SHA2-384: (A7716) SHA2-512: (A7716)
KAS-ECC-SSC	KAS-SSC	Shared secret computation using elliptic curves	IG D.F:Scenario 2, path (2)	KAS-ECC-SSC Sp800-56Ar3: (A7716)
KAS-FFC-SSC	KAS-SSC	Shared secret computation using finite fields	IG D.F:Scenario 2, path (2)	KAS-FFC-SSC Sp800-56Ar3: (A7716)
Hash	SHA	Generate a message digest		SHA2-256: (A7716) SHA2-384: (A7716) SHA2-512: (A7716)
CKG	CKG	Symmetric key generation and seeds for asymmetric keys		CKG: ()

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

The following algorithm specific information applies to algorithms used by the module in the Approved mode of operation.

2.7.1 AES-GCM

The module supports random IV generation for AES-GCM compliant with IG C.H scenario 2. IVs are generated from the output of the module's approved DRBG. At least 96 bits are requested by the cipher when initialized.

The Module does not persistently store AES-GCM keys or IVs. In the event that the module's power is lost, a new handshake is required to establish new keys and IVs. This condition is not enforced by the module but is met implicitly.

2.7.2 SHA-1

The module uses SHA-1 in an approved manner for digital signature verification and to effect the integrity test. The module also uses SHA-1 (within HMAC-SHA-1) for the generation of message authentication codes (MACs). Refer to section 9.5 for the transition timeframes related to SHA-1.

2.7.3 RSA

Verifying an RSA signature using a 1024-bit key, or computed over a SHA-1 hash, may only be performed on a signature that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M

2.7.4 ECDSA

Verifying an ECDSA signature computed over a SHA-1 hash may only be performed on a signature that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M

2.8 RBG and Entropy

The module generates random bits from its internal implementation of CTR_DRBG. This DRBG has been CAVP-tested and is compliant to SP 800-90Ar1. The DRBG is seeded and reseeded from an entropy source outside the module, but within the tested operational environment (TOEPP) of the host platform.

As described in section 2.1.1, the module is only available as a component of the HPE Aruba Networking Virtual Intranet Access (VIA) product. This product is responsible for obtaining entropy from the source within the host hardware platform, generating the seed, and loading that seed into the module. The mechanics of this have not been verified by the testing lab. However, it is believed that 384-bits of full entropy are loaded into the module.

This method of seeding is compliant to the March 26, 2024 version of IG 9.3.A Resolution 2(b). The following entropy caveat is applicable:

No assurance of the minimum strength of generated SSPs (e.g., keys).

2.9 Key Generation

The cryptographic module performs Cryptographic Key Generation (CKG) in compliance with section 4 of SP 800-133r2. Symmetric cryptographic keys and seeds for asymmetric keys are generated by the direct unmodified output of the module's approved DRBG.

2.10 Key Establishment

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

2.11 Industry Protocols

The module does not support any industry protocols.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

As a software module, the module interfaces are defined as Software or Firmware Module Interfaces (SFMI), and there are no physical ports. The logical interfaces are defined as the API of the cryptographic module. The interfaces are listed in the table below.

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
N/A	Status Output	API return codes, status information, error codes, non-approved service logs

Table 9: Ports and Interfaces

The module does not implement a control output interface.

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

4 Roles, Services, and Authentication

4.1 Authentication Methods

The Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* does not provide any identification or authentication methods of its own.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
CO	Role	Crypto Officer	None

Table 10: Roles

The module only supports a Crypto Officer role, which is assumed implicitly. The module does not support multiple concurrent operators, a user role, a maintenance role, nor bypass capability.

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
DRBG	Random bit generation	API return code	Command	Random bits	DRBG	CO - Entropy Input: W,E - DRBG Seed: G,E - DRBG Key: G,E - DRBG V: G,E - DRBG Output: G,R
Decrypt	Data decryption	API return code	API return code	Plaintext	Decrypt AuthDecrypt	CO - AES Key: W,E
Encrypt	Data encryption	API return code	Key, plaintext	Ciphertext	Encrypt AuthEncrypt	CO - AES Key: W,E
Hash	Generate message digest	API return code	Message	Hash	Hash	CO
Key generation	Symmetric key generation	API return code	Command	Key	DRBG CKG	CO - DRBG Output: E - Symmetric Key: G,R
Key pair generation	Generation of key pairs	API return code	Command, domain parameters	Key pairs	DRBG KeyGen CKG	CO - DRBG Output: E - EC DH Private Key: G,R - EC DH Public Key: G,R - ECDSA Signature Private Key: G,R - ECDSA Signature Public Key: G,R - RSA Signature Private Key: G,R - RSA Signature Public Key: G,R - DH Private Key: G,R - DH Public Key: G,R
Key wrapping	Key wrapping	API return code	Key, key to be wrapped	Wrapped key	AuthEncrypt	CO - Key Wrapping Key: W,E
Key unwrapping	Key unwrapping	API return code	Key, wrapped key	Unwrapped key	AuthDecrypt	CO - Key Wrapping Key: W,E
MAC generation	Message authentication code generation	API return code	Key, message	MAC	MAC	CO - HMAC Key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
MAC verification	Message authentication code verification	API return code	Key, message, MAC	Ruling	MAC	CO - HMAC Key: W,E
Self-test	Perform integrity and algorithm self-tests	API return code	Command	Status	None	CO
Show status	Display module status information	-	-	-	None	CO
Show version	Display module version information	-	-	-	None	CO
Signature generation	Digital signature generation	API return code	Key, message	Key, message	DRBG SigGen	CO - DRBG Output: E - ECDSA Signature Private Key: W,E - RSA Signature Private Key: W,E
Signature verification	Digital signature verification	API return code	Key, message, signature	Ruling	SigVer Legacy SigVer	CO - ECDSA Signature Public Key: W,E - RSA Signature Public Key: W,E
SSC ECC	Computation of shared secret based on elliptic curves	API return code	Keys	Shared secret	KAS-ECC-SSC	CO - EC DH Private Key: W,E - EC DH Public Key: W,E - EC DH Shared Secret: G,R
SSC FFC	Shared secret computation based on finite fields	API return code	Keys	Shared secret	KAS-FFC-SSC	CO - DH Private Key: W,E - DH Public Key: W,E - DH Shared Secret: G,R
Key destruction	Zeroise context	API return code	Command, Key handle	Message	None	CO - AES Key: Z - DH Private Key: Z - DH Public Key: Z - EC DH Private Key: Z - EC DH Public Key: Z - DH Shared Secret: Z - ECDSA Signature Private Key: Z - ECDSA Signature Public Key: Z - RSA Signature Private Key: Z - RSA Signature Public Key: Z - HMAC Key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Key Wrapping Key: Z - Symmetric Key: Z - DRBG Key: Z - DRBG V: Z

Table 11: Approved Services

The meaning of the letters used to describe the 'SSP Access' are:

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

The Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* is interfaced by the HPE Aruba Networking applications using API calls (the calling applications that run on the host GPC are out of scope for this validation). Successful completion of a security service (via API return code for success) when the module is in Approved mode denotes use of approved security service.

4.3.1 Show version

The module does not support an invocable show version service. The module name and version are output each time the HPE Aruba Networking application is instantiated. The system log will include the following entries:

```
Product Name: HPE Aruba Networking VIA Crypto Module  
Product Version: 1.1
```

4.3.2 Show status

The module does not support an invocable show status service. Current module status is output as part of each cryptographic service invocation. The completion of a service indicates that the module is operational.

4.4 Non-Approved Services

The module does not support a non-approved mode and does not offer any non-approved services.
N/A for this module.

4.5 External Software Loaded

The module does not support software loading.

5 Software Security

5.1 Integrity Techniques

The module performs a software integrity test when powered on (refer to Self-Tests for details). All cryptographic algorithm self-tests are run at power-up, prior to the first operational use of the cryptographic algorithm. The software integrity test verifies the integrity of the module by comparing a calculated HMAC-SHA-1 value against the stored HMAC value.

5.2 Initiate on Demand

The operator can initiate the software integrity test on demand by reloading the module into memory on the host or by calling `FIPS_powerupSelfTest()`. All data output via the data output interface is inhibited until the software integrity test has completed successfully. If the software integrity test fails, the module enters the error state (while in this state, the module provides no functionality). The temporary values generated during the software integrity test are zeroized upon completion of the integrity test. After the module is (re)loaded into memory, the operator can determine the version of the loaded module through reviewing the host log.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

How Requirements are Satisfied

The module operates on a general-purpose computer (GPC) running Microsoft Windows, Android, Linux, MacOS, or iOS as a general-purpose operating system (GPOS). The module's interfaces are logical and are provided through the Application Programming Interface (API) that a calling program can access. The logical interfaces expose services that applications directly call, and the API provides functions that may be called by a referencing application. The module distinguishes between the logical interfaces by logically separating the information according to the defined API. The module APIs are restricted command sets. These operating control mechanisms protect against unauthorized execution, unauthorized modification, and unauthorized reading of SSPs, control and status data.

6.2 Configuration settings and restrictions

The module is only used in conjunction with the calling HPE Aruba Networking application on a host general-purpose computer (GPC) running a single-user mode general purpose operating system (GPOS).

See section 11.1 for guidance on configuring the host device and installing the module.

7 Physical Security

The module is a multiple-chip standalone software module and does not include any physical components. No physical security mechanisms or protections are implemented, and the FIPS 140-3 section 7 requirements do not apply to the module.

8 Non-Invasive Security

Since the module has not been purposely designed, built and publicly documented to include non-invasive mitigation techniques, the Non-Invasive Security requirements are not applicable.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	All SSPs are stored in volatile RAM memory	Dynamic

Table 12: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API Import	Calling application	RAM	Plaintext	Manual	Electronic	
API Export	RAM	Calling application	Plaintext	Manual	Electronic	

Table 13: SSP Input-Output Methods

The SSPs are provided to the module in plaintext form via input API parameters, to and from the calling application running in the same operational environment.

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Service	The calling application calls the Key Destruction service APIs to zeroize the SSP.	The application program generates the SSP external to the module and is responsible for zeroizing the SSP by calling the Key Destruction service API.	Call to destruction API
Auto	The service using the SSP automatically zeroizes the SSP after use.	The SSP is single use and so is zeroized after use.	N/A
Reboot	Reboot the host device.	All RAM is zeroized by a power-cycle.	Power-cycling the host device

Table 14: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES Key	Key used for AES operations	128, 192, 256 - 128, 192, 256	Symmetric - CSP			Encrypt Decrypt AuthEncrypt AuthDecrypt
Entropy Input	Externally generated entropy	384 - 384	Entropy - CSP			DRBG
DRBG Seed	Internal state for DRBG	384 - 384	DRBG state - CSP	DRBG		DRBG
DRBG Key	Internal state for DRBG	256 - -	DRBG state - CSP	DRBG		DRBG
DRBG V	DRBG V	128 - -	DRBG state - CSP	DRBG		DRBG
DRBG Output	Random numbers used in cryptographic algorithms	128, 192, 256 - 128, 192, 256	DRBG state - CSP	DRBG		SigGen KeyGen CKG
DH Private Key	DH Private Key	256, 2048 - 112	Asymmetric - CSP	KeyGen		KAS-FFC-SSC
DH Public Key	DH Public Key	2048 - 112	Asymmetric - PSP	KeyGen		KAS-FFC-SSC
DH Shared Secret	DH Shared Secret	2048 - 112	Shared secret - CSP		KAS-FFC-SSC	
EC DH Private Key	EC DH Private Key	P-256, P-384 - 128, 192	Asymmetric - CSP	KeyGen		KAS-ECC-SSC
EC DH Public Key	EC DH Public Key	P-256, P-384 - 128, 192	Asymmetric - PSP	KeyGen		KAS-ECC-SSC
EC DH Shared Secret	EC DH Shared Secret	128, 192 - 128, 192	Shared secret - CSP		KAS-ECC-SSC	
ECDSA Signature Private Key	Key used for ECDSA Signature Generation	P-256, P-384 - 128, 192	Asymmetric - CSP	KeyGen		SigGen
ECDSA Signature Public Key	Key used for ECDSA Signature Verification	P-256, P-384 - 128, 192	Asymmetric - PSP	KeyGen		SigVer
RSA Signature Private Key	Key used for RSA Signature Generation	2048 - 112	Asymmetric - CSP	KeyGen		SigGen
RSA Signature Public Key	Key used for RSA Signature Verification	1024, 2048 - 80, 112	Asymmetric - PSP	KeyGen		SigVer Legacy SigVer
HMAC Key	Key used for HMAC Operations	192, 256, 384 - 192, 256, 384	Symmetric - CSP			MAC
Key Wrapping Key	Key used for key wrapping and unwrapping operations	128, 192, 256 - 128, 192, 256	Symmetric - CSP			AuthEncrypt AuthDecrypt
Symmetric Key	Symmetric key generated by the module	128, 192, 256 - 128, 192, 256	Symmetric - CSP	CKG		

Table 15: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES Key	API Import	RAM:Plaintext		Service Reboot	
Entropy Input	API Import	RAM:Plaintext		Auto	
DRBG Seed		RAM:Plaintext		Auto	Entropy Input:Derived From
DRBG Key		RAM:Plaintext		Service Reboot	DRBG Seed:Derived From
DRBG V		RAM:Plaintext		Service Reboot	DRBG Seed:Derived From
DRBG Output	API Export	RAM:Plaintext		Service Reboot	DRBG Key:Derived From DRBG V:Derived From
DH Private Key	API Import API Export	RAM:Plaintext		Service Reboot	DH Public Key:Paired With
DH Public Key	API Import API Export	RAM:Plaintext		Service Reboot	DH Private Key:Paired With
DH Shared Secret	API Export	RAM:Plaintext		Service Reboot	DH Private Key:Derived From DH Public Key:Derived From
EC DH Private Key	API Import API Export	RAM:Plaintext		Service Reboot	EC DH Public Key:Paired With
EC DH Public Key	API Import API Export	RAM:Plaintext		Service Reboot	EC DH Private Key:Paired With
EC DH Shared Secret	API Export	RAM:Plaintext		Service Reboot	EC DH Private Key:Derived From EC DH Public Key:Derived From
ECDSA Signature Private Key	API Import API Export	RAM:Plaintext		Service Reboot	ECDSA Signature Public Key:Paired With
ECDSA Signature Public Key	API Import API Export	RAM:Plaintext		Service Reboot	ECDSA Signature Public Key:Paired With
RSA Signature Private Key	API Import API Export	RAM:Plaintext		Service Reboot	RSA Signature Public Key:Paired With
RSA Signature Public Key	API Import API Export	RAM:Plaintext		Service Reboot	RSA Signature Public Key:Paired With
HMAC Key	API Import	RAM:Plaintext		Service Reboot	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
Key Wrapping Key	API Import	RAM:Plaintext		Service Reboot	
Symmetric Key	API Export	RAM:Plaintext		Service Reboot	

Table 16: SSP Table 2

9.5 Transitions

The module implements the following security methods that are forecasted to transition to non-approved over the lifetime of the validation:

Algorithm	Use
SHA-1	In HMAC-SHA-1
HMAC-SHA-1	For MAC Generation

SHA-1 becomes disallowed for applying cryptographic protection in January 2031.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
Integrity Test	HMAC-SHA-1 with 117-byte key	KAT	SW/FW Integrity	Successful module initialization	MAC verification

Table 17: Pre-Operational Self-Tests

Pre-operational self-tests are performed automatically when the module is loaded into memory. While the module is executing the pre-operational self-tests, services are not available, and input and output are inhibited because the module will not return to the calling application until the self-tests are complete.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES encrypt	256 bit keys, 128 bit message	KAT	CAST	Successful module initialization	Encrypt	On load
AES decrypt	256 bit keys, 128 bit message	KAT	CAST	Successful module initialization	Decrypt	On load
AES-GCM encrypt	256 bits	KAT	CAST	Successful module initialization	Encrypt	On load
AES-GCM decrypt	256 bits	KAT	CAST	Successful module initialization	Decrypt	On load
DRBG	256-bit, with DF	KAT	CAST	Successful module initialization	Instantiate, Generate and Reseed	On load
ECDSA	P-256 with SHA2-256, P-384 with SHA2-384, fixed k	KAT	CAST	Successful module initialization	Sign & Verify	On load
HMAC	SHA-1, SHA2-256, SHA2-384, SHA2-512	KAT	CAST	Successful module initialization	Generate	On load
ECC-SSC	P-384	KAT	CAST	Successful module initialization	Compute	On load
FFC-SSC	L=2048, N=256	KAT	CAST	Successful module initialization	Compute	On load
RSA	2048-bit, PKCS#1-v1.5, SHA2-256	KAT	CAST	Successful module initialization	Sign & Verify	On load
SHA	SHA-1, SHA2-256, SHA2-384, SHA2-512	KAT	CAST	Successful module initialization	Generate	On load
ECC key pairs	-	PCT	PCT	No error	Sign & Verify	On key pair generation
FFC key pairs	-	PCT	PCT	No error	Sign & Verify	Shared secret computation
RSA key pairs	SHA2-256	PCT	PCT	No error	Sign & Verify	On key pair generation

Table 18: Conditional Self-Tests

All CASTs are performed automatically when the module is loaded into memory, prior to executing the pre-operational self-tests.

10.3 Periodic Self-Test Information

The following self-tests can be invoked by the Crypto Officer as described in section 10.5 below. The module cannot invoke these tests automatically.

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Integrity Test	KAT	SW/FW Integrity	On demand	Manually

Table 19: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES encrypt	KAT	CAST	On demand	Manually
AES decrypt	KAT	CAST	On demand	Manually
AES-GCM encrypt	KAT	CAST	On demand	Manually
AES-GCM decrypt	KAT	CAST	On demand	Manually
DRBG	KAT	CAST	On demand	Manually
ECDSA	KAT	CAST	On demand	Manually
HMAC	KAT	CAST	On demand	Manually
ECC-SSC	KAT	CAST	On demand	Manually
FFC-SSC	KAT	CAST	On demand	Manually
RSA	KAT	CAST	On demand	Manually
SHA	KAT	CAST	On demand	Manually
ECC key pairs	PCT	PCT	-	-
FFC key pairs	PCT	PCT	-	-
RSA key pairs	PCT	PCT	-	-

Table 20: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Critical Error	Critical Error state, the module provides no functionality	Failed integrity test Failed CAST	Reload module	Nonzero value for gFIPS_powerupStatus
PCT Error	An error state that is entered when a pairwise consistency test fails on asymmetric keypair generation.	Failed PCT	Automatic	Keypair generation function returns error

Table 21: Error States

When a cryptographic algorithm self-test or pre-operational self-test fails, the module enters the Critical Error state. While in this state, the module provides no functionality and inhibits data output, logs the error, and halts - subsequent calls to the module will also fail, thus no further cryptographic operations are possible. To recover from a self-test failure, the module should be reloaded into memory.

When a conditional pairwise consistency test fails on asymmetric keypair generation, the module returns an error.

10.5 Operator Initiation of Self-Tests

The operator may initiate self-tests on-demand by reloading the module into memory on the host or by calling `FIPS_powerupSelfTest()`.

10.6 Additional Information

Upon successful completion of the self-tests when the module is loaded into memory, the module displays results on the host GPC console:

Windows:	FIPS_powerupSelfTest Success
iOS:	FIPS powerup self test success
Android:	Finished Mocana initialization...
Linux:	Power up self test passed...

In the event any self-test fails, the module will enter a Critical Error state (while in this state, the module provides no functionality and inhibits data output), logs the error, and halts - subsequent calls to the module will also fail, thus no further cryptographic operations are possible. To recover from a self-test failure, the module should be reloaded into memory. After a self-test failure, the module displays results on the host GPC console:

Windows:	FIPS_powerupSelfTest() failed error <code>
iOS:	FIPS powerup self test failed with error <code>
Android:	Mocana initialization error <code>
Linux:	!!ERROR!! :Power up self test failed:<code>

When the software integrity test fails when the module is loaded into memory, the module enters the Critical Error state and sets `gFIPS_powerupStatus` to `ERR_FIPS_INTEGRITY_FAILED`.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Software Installation

The module is distributed as part of an HPE Aruba Networking application.

Start-up Procedures

Once the module starts up successfully, after passing all Cryptographic Algorithm Self-Tests (CASTs) and Pre-Operational Self-Tests (POSTs), the calling application must call

```
FIPS_enableNonApprovedLogging()
```

To verify that the module is operating in a compliant manner, the calling application can call:

```
FIPS_isNonApprovedLoggingEnabled()
```

A response of True indicates that the module has been configured to operate in a compliant manner. Any other response indicates that the module is operating in a non-compliant state.

11.2 Administrator Guidance

11.2.1 Zeroization

In the module, there is a context structure associated with every available cryptographic algorithm. Context structures hold sensitive information such as cryptographic keys. These context structures must be destroyed via respective key destruction service API calls when the calling HPE Aruba Networking application software no longer needs to use a specific algorithm. The key destruction service API call will zeroize all sensitive information, including cryptographic keys, before freeing the dynamically allocated memory.

11.2.2 AES-GCM

The Module's implementation of AES-GCM shall be used together with one or more applications outside the module's cryptographic boundary that implement the IPSec, TLS or SSH protocol. The keys and IVs for AES-GCM shall be obtained from the protocol's corresponding KDF.

For general-purpose encryption, to ensure compliance with IG C.H, either the key or the IV for AES-GCM must be randomly generated from the module's approved DRBG. Obtaining both from outside the module is non-compliant.

11.3 Non-Administrator Guidance

The module is not distributed as a standalone library for direct download and is only used in conjunction with the calling HPE Aruba Networking application when a trusted and verified HPE Aruba Networking application product is downloaded in electronic form from the HPE Networking Support Portal (NSP) by authenticated licensed customer personnel and installed on a host general-purpose computer (GPC) using a general purpose operating system (GPOS) platform specified in section 2.2, Tested and Vendor Affirmed Module Version and Identification, or one where portability is maintained.

Documentation for any Hewlett Packard Enterprise product can be found on the HPE Networking Support Portal (NSP). Filters can be used to limit the displayed results by Product(s), Product Series, Version(s), and File Category. For example,

- The HPE Aruba Networking Virtual Intranet Access (VIA) version 4.x product uses the *HPE Aruba Networking VIA Crypto Module*. Full HPE Aruba Networking Virtual Intranet Access (VIA) documentation for VIA 4.x can be found at the link provided below after authentication.

<https://networkingsupport.hpe.com/downloads;pageSize=100;fileTypes=DOCUMENT;softwareGroups=VIA;softwareMajorVersions=4>

- The HPE Aruba Networking Virtual Intranet Access (VIA) version 4.x product uses HPE Aruba Networking Gateways, including Mobility Conductors and Mobility Controllers, for setting FIPS mode within the VIA Connection profiles. Full Hewlett Packard Enterprise AOS-8 Gateway documentation for Mobility Conductors and Mobility Controllers can be found at the link provided below after authentication.
<https://networkingsupport.hpe.com/downloads;pageSize=100;fileTypes=DOCUMENT;softwareGroups=AOS-8%20GW>

The following Hewlett Packard Enterprise documents can be referenced to ensure that the *HPE Aruba Networking VIA Crypto Module* and the calling HPE Aruba Networking application product software are installed and operated correctly in the Approved mode:

- HPE Aruba Networking VIA 4.x User Guide
- ArubaOS 8.12.0.0 User Guide
- ArubaOS 8.x CLI Reference Guide

The Crypto Officer shall perform the following steps to set-up the HPE Aruba Networking calling application product software to support the *HPE Aruba Networking VIA Crypto Module* in Approved mode:

1. The Crypto Officer shall work with their Hewlett Packard Enterprise representative to obtain appropriate software licenses for the HPE Aruba Networking calling application product software.
2. Since the HPE Aruba Networking VIA Crypto Module software is used in conjunction with the calling HPE Aruba Networking application product software, the Crypto Officer shall download the calling HPE Aruba Networking application product software in electronic form from the HPE Networking Support Portal (NSP) after authenticating as a licensed HPE customer. The module was tested on the platforms listed above in section 2.2, Tested and Vendor Affirmed Module Version and Identification.
3. Since the calling HPE Aruba Networking application product software can use HPE Aruba Networking Gateways, including Mobility Conductors and Mobility Controllers, running a trusted and verified Hewlett Packard Enterprise AOS software image, for setting the module's FIPS mode within the managed Connection profiles, the Crypto Officer (CO) shall review the reference documents listed below in section 11.2, Administrator Guidance. Select the host device deployment scenario that best fits your installation and follow the scenario's deployment procedures.
4. Install the calling HPE Aruba Networking application on a host general-purpose computer (GPC) running a single-user mode general purpose operating system (GPOS) platform specified in section 2.3, Operating Environments, or one where portability is maintained.
5. Configure the HPE Aruba Networking Gateway, including a Mobility Conductor or Mobility Controller, running a trusted and verified Hewlett Packard Enterprise AOS software image, for the calling HPE Aruba Networking application software deployment.
 - a. Add and enable the appropriate software licenses for the HPE Aruba Networking calling application product software.
 - b. Follow the configuration guidance in the reference documents listed below in section 11.2, Administrator Guidance to configure the HPE Aruba Networking Gateway running Hewlett Packard Enterprise AOS software for the calling HPE Aruba Networking application software deployment.
 - c. Within the configured Connection Profile created, the Crypto Officer (CO) shall enable the FIPS module such that the calling HPE Aruba Networking application software checks for FIPS compliance during startup.
 - i. If using the AOS WebUI for configuration:
 1. Navigate to Configuration → Authentication → L3 Authentication → Connection
 2. Under Connection Profile: New Profile, click + to add a new profile
 3. Enter a new profile name

4. Select 'Enable FIPS Module'
 5. Click Submit
 6. Select Deploy Changes in the Pending Changes window.
 - ii. If using the AOS CLI for configuration, enter:


```
#aaa authentication via connection-profile <profile name>
#enable-fips
#exit
#write memory
Saving Configuration...
Configuration Saved.
```
 - iii. To show that FIPS has been enabled in the connection profile, enter:


```
#show aaa authentication via connection-profile <profile name>
...
Enable FIPS Module                Enabled
...
```
 - iv. To refresh the calling HPE Aruba Networking application's Connection Profile on the host general-purpose computer (GPC):
 1. Disconnect the device.
 2. Exit the calling software.
 3. Immediately relaunch the calling software.
 4. Repeat steps 1-3 a second time to refresh the profile.
 5. Invoke the show version service to verify the name and version of the Hewlett Packard Enterprise *HPE Aruba Networking VIA Crypto Module* matches that on the CMVP certificate.
6. Monitor the calling HPE Aruba Networking application software messages displayed on the host GPC console during startup.

If any of the checks fail, error messages will be displayed on the host GPC console. If the errors persist after the calling HPE Aruba Networking application software has been reloaded into memory, contact Hewlett Packard Enterprise.

11.6 End of Life

As a software library, the module itself does not have any special end of life procedures. All module SSPs can be zeroized by unloading the module from memory.

To determine if an Hewlett Packard Enterprise product is considered end of life, refer to the Hewlett Packard Enterprise end-of life information at <https://networkingsupport.hpe.com/end-of-life>. If a Hewlett Packard Enterprise product is deemed end-of-life, the CO should work with their Hewlett Packard Enterprise representative to determine the appropriate Hewlett Packard Enterprise product upgrade path to use a newer Approved version.

For secure sanitization and zeroization of SSP values, follow the guidance in the Key Destruction service entry above in section 9.3, SSP Zeroization Methods. Since the module is not distributed as a standalone library for direct download and is only used in conjunction with the calling HPE Aruba Networking application, if the module is deprecated, the module will be upgraded to a newer Approved validated version by loading the newer module version into memory on the GPC host. Loading the newer module into memory on the GPC host zeroizes all SSPs stored in volatile RAM memory and then all self-tests including the software integrity test will be run on the GPC host. Installation of the calling HPE Aruba Networking application product software on the GPC host can be supported with the help of a Hewlett Packard Enterprise-Certified Mobility Professional (ACMP).

12 Mitigation of Other Attacks

The module has not been purposely designed, built and publicly documented to mitigate one or more specific attacks. The Mitigation of Other Attacks requirements are not applicable, per FIPS 140-3 IG 12.A.