Aruba VIA Cryptographic Module

Non-Proprietary Security Policy

FIPS 140-2 Level 1

a Hewlett Packard
Enterprise company

Version 1.2

October 2022

Copyright

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:

http://www.arubanetworks.com/open_source

Legal Notice

The use of Aruba switching platforms and software, 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, Aruba. from any and all legal actions that might be taken against it with respect to infringement of copyright on behalf of those vendors.

aruba
a Hewlett Packard
Enterprise company

www.arubanetworks.com

6280 America Center Dr
San Jose, CA, USA 95002
Phone: 408.941.4300
Fax 408.752.0626

Contents

Figures

Tables

Document Revision History

Version	Date	Description
1.0	December 2021	Initial FIPS 140-2 Level 1 release for Aruba VIA Cryptographic Module version 1.0
1.1	August 2022	Updates to address CMVP Reviewer comments

Preface

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

1. Purpose of this Document

This release supplement provides information regarding the Aruba VIA Cryptographic Module FIPS 140-2 Level 1 validation from Aruba Networks. The material in this supplement modifies the general Aruba software documentation included with this product and should be kept with your Aruba product documentation. Aruba Networks is a Hewlett Packard Enterprise company.

This supplement primarily covers the non-proprietary Cryptographic Module Security Policy for the Aruba VIA Cryptographic Module. This security policy describes how the module meets the security requirements of FIPS 140-2 Level 1 and how to place and maintain the module in a secure FIPS 140-2 mode. This policy was prepared as part of the FIPS 140-2 Level 1 validation of the product.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2, *Security Requirements for Cryptographic Modules*) details the U.S. Government requirements for cryptographic modules. More information about the FIPS 140-2 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 Aruba VIA Cryptographic Module is referred to as the VIA Cryptographic Module, AVCM, the cryptographic module, or the module.

1.1. Related Documents

The module is not distributed as a standalone library and is only used in conjunction with Aruba Networks solutions. As such, there is no direct User Guidance.

An example of an Aruba product that uses the FIPS module is Aruba Virtual Intranet Access (VIA) 4.x – see https://www.arubanetworks.com/techdocs/VIA/4x/Content/Overview/Preface.htm and https://www.arubanetworks.com/techdocs/VIA/4x/Content/Overview/Feature%20Parity41.htm?Highlight=fips

1.1.1. Additional Product Information

More information is available from the following sources:

- The Aruba Networks Web-site contains information on the full line of products from Aruba Networks:

  https://www.arubanetworks.com

- 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 **Aruba** in the Vendor field then select Search to see a list of FIPS certified Aruba products.

  Select the Certificate Number for the Module Name ‘Aruba VIA Cryptographic Module’.

2. Overview

The Aruba VIA Cryptographic Module Version 1.0, is a software shared library that provides cryptographic services required by Aruba Networks software applications. The Module's logical cryptographic boundary is the shared dynamic library files (ancrypto.dll, libancrypto.so, and libancrypto.a) and their integrity check HMAC files.

The module is a multi-chip standalone embodiment installed on a General Purpose Computer (GPC).

The software version validated is **AVCM Version 1.0**.

Aruba's development processes are such that future releases under AVCM 1.0 should be FIPS validate-able and meet the claims made in this document. Only the versions that explicitly appear on the certificate, however, are formally validated. The CMVP makes no claim as to the correct operation of the module or the security strengths of the generated keys when operating under a version that is not listed on the validation certificate.

The tested platforms are:

- Android 11 (64-bit ARM) on Samsung Galaxy S20
- Windows 10 (64-bit x86_64) on HP ZBook G7
- iOS 14 (64-bit ARM) on iPhone 11

The list of vendor affirmed devices for the virtual appliances are listed below. Aruba believes all functionality claimed within this Security Policy can be successfully met with these devices.

- Android 8.x, 9.x, 10.x
- Windows 8.1
- Windows 10 Home, Professional
- Linux:
    - Ubuntu 14.04, 16.04, 18.04, 20.04
    - CentOS 7+
    - RHEL 7.3+, 8+
    - Debian 9+
- iOS 7.x, 9.x, 10.x, 12.x, 13.x, 14.x
- MacOS 10.11, 10.12, 10.13, 10.14, 10.15, 11

The CMVP makes no claim as to the correct operation of the module or the security strengths of the generated keys when ported to an operational environment which is not listed on the validation certificate.

2.1. Cryptographic Module Boundaries

For FIPS 140-2 Security Level 1 validation, the module has been tested as a multi-chip standalone software module. The logical boundary of the module consists of the extents of the library files which make up the module. All operations of the module occur via calls from Aruba applications and their respective internal daemons/processes. As such there are no untrusted services calling the services of the module, as APIs are not exposed to any processes except the Aruba components. This module is linked to the calling Aruba application and no interface is provided to the user, so only functions within the module can call the cryptographic functions. The Aruba applications that use the API calls with the module are out of scope for this validation.

The physical boundary is the surface of the GPC computer chassis and the modifiable operational environment runs completely within the physical boundary.

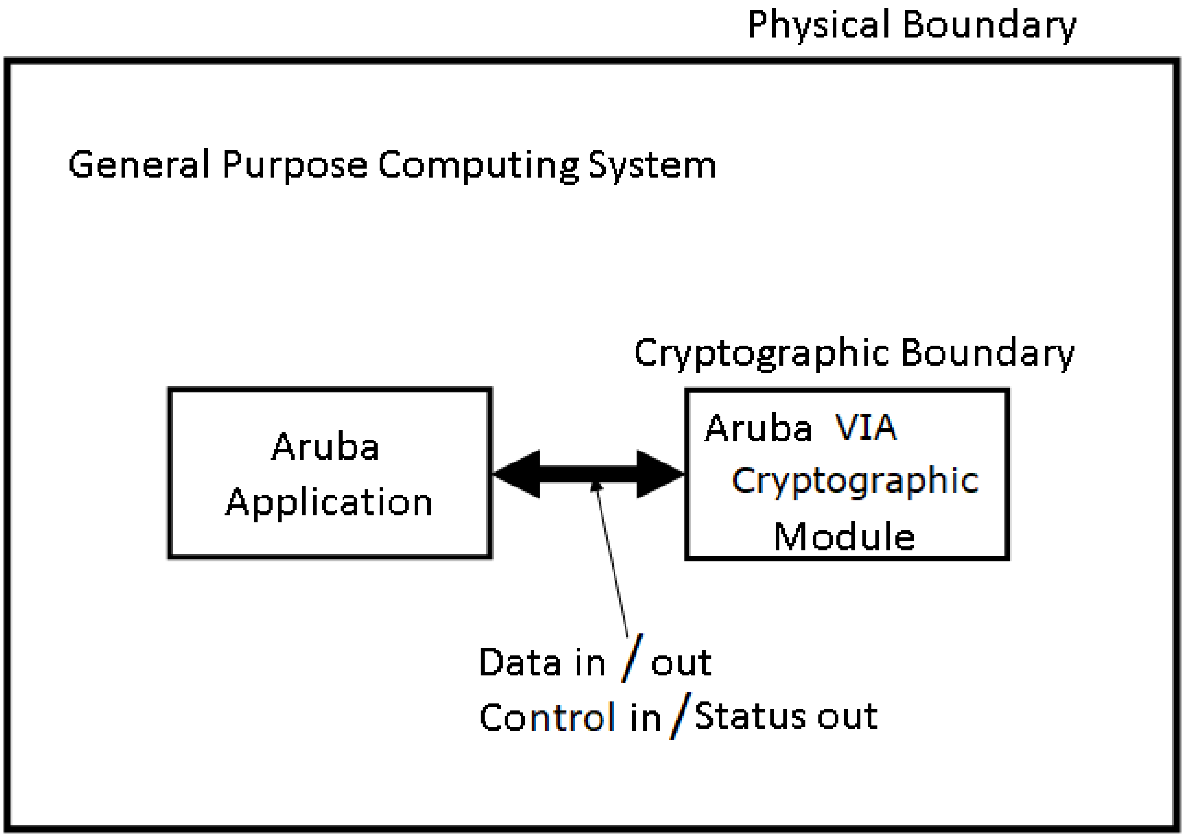

Figure 1 – Functional Block Diagram of Module Cryptographic Boundary and Interfaces

2.2. Intended Level of Security

The module is intended to meet overall FIPS 140-2 Security Level 1 requirements.

Table 1 – Intended Level of Security

Section	Section Title	Security Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC	1
9	Self-tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A
Overall	**Overall module validation level**	**1**

3. Physical Security

The module is a software module and does not implement any physical security mechanisms. It must be run on a production grade platform (such as a standard commercially made PC, laptop, server, smartphone, etc.) to meet requirements from FIPS 140-2 Security Level 1.

4. Operational Environment

The module operates on a general-purpose computer (GPC) running Microsoft Windows, Android, Linux, MacOS, or iOS as a general purpose operating system (GPOS). For FIPS purposes, the module is running on this operating system in single user mode and does not require any additional configuration to meet the FIPS requirements.

The module was tested on the platforms listed in section 2 above. The platforms used during testing met Federal Communications Commission (FCC) Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) requirements for business use as defined by 47 Code of Federal Regulations, Part 15, Subpart B, Class A.

FIPS 140-2 validation compliance is maintained when the module is operated on other versions of the GPOS running in single user mode, assuming that the requirements outlined in NIST IG G.5 are met.

5. Logical Interfaces

The API provided by the module is mapped onto the FIPS 140-2 logical interfaces: data input, data output, control input, and status output. Each of the FIPS 140-2 logical interfaces relates to the module's callable interface, as described in the following table.

The interface ports shown below include the keyboard, CDROM drive, floppy disk, mouse, network port, parallel port, USB ports, monitor port and power connection of the GPC. 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 (see the following section for the list of available functions).

Table 2 – FIPS 140-2 Logical Interfaces

FIPS 140-2 Logical Interface	Module Virtual Interface	Module Physical Interface
Data Input Interface	Input parameters of API function calls	• Network Interface
Data Output Interface	Output parameters of API function calls	• Network Interface
Control Input Interface	API function calls	• Command input interfaces
Status Output Interface	For FIPS mode, function calls returning status information and return codes provided by API function calls	• Display Controller
Power Interface	N/A	• Host GPC Power Interface

Data input and output, control input, status output, and power interfaces are defined as follows:

- Data input and output are the packets that use the firewall, VPN, and routing functionality of the modules.
- Control input consists of virtual control inputs for power and reset through the power and reset interface. It also consists of all of the data that is entered into the controller while using the Host interfaces.
- Status output consists of the status indicators displayed through the status data that is output from the module while using the Host management interfaces, and the log file.
    - The hosts console indicates the virtual state such as power-up (or rebooting), utilization level, and activation state. The log file records the results of self-tests, configuration errors, and monitoring data.
- A power supply is used by the virtualization host.

The module distinguishes between the logical interfaces by logically separating the information according to the defined API.

The output data path is provided by the data interfaces and is logically disconnected from processes performing key generation or zeroization. No key information will be output through the data output interface when the module zeroizes keys. No key information is output during key generation.

6. Roles, Services, and Authentication

There are two roles in the module that operators may assume: a Crypto Officer role and a User role. The Administrator maps to the Crypto-Officer role and the client Users map to the User role. The module does not support a Maintenance role and/or bypass capability.

The module does not support authentication to access services, as allowed by FIPS 140-2 Security Level 1 requirements. The User and Crypto-Officer roles are implicitly assumed by the entity accessing services implemented by the Module.

6.1. Crypto Officer Role

The Crypto Officer role has the ability to configure, manage, and monitor the module.

See the table below for descriptions of the Approved/allowed services available to the Crypto Officer role.

Table 3 – Crypto-Officer Services

Service	Description	Input	Output	CSP/Algorithm Access (please see Table 6 below for details)
DH Key Exchange	Use DH Private Component to generate the DH Shared Secret	DH Private Component	DH Shared Secret	5, 7 (read/write/delete)
ECDH Key Generation	Use ECDH Public and Private Components to generate the ECDH key pair	ECDH Public and Private Components	ECDH key pair	8, 9 (read/write/delete)
ECDH Key Exchange	Use ECDH Private Component to generate the ECDH Shared Secret	ECDH Private Component	ECDH Shared Secret	8, 10 (read/write/delete)
RSA Key Generation	Generate RSA Public/Private key pair for RSA Signature Generation / Verification	Commands and configuration data	RSA Public/Private key pair	11, 12 (read/write/delete)
RSA Signature Generation	Use RSA Private Key to generate RSA Signature	RSA Private Key	RSA Signature	11 (read/write/delete)
RSA Signature Verification	Use RSA Public Key to verify RSA Signature	RSA Public Key	Status of commands	12 (read)
RSA Key Wrapping Encryption	Use RSA Public Key to perform RSA Key Wrapping encryption	RSA Public Key	RSA Key Wrapping Public Key	12, 14 (read/write)
RSA Key Wrapping Decryption	Use RSA Private Key to perform RSA Key Wrapping decryption	RSA Private Key	RSA Key Wrapping Private Key	11, 13 (read/write/delete)

ECDSA Key Generation	Generate ECDSA Public/Private key pair for ECDSA Signature Generation / Verification	Commands and configuration data	ECDSA Public/Private key pair	15, 16 (read/write/delete)
ECDSA Signature Generation	Use ECDSA Private Key to generate ECDSA Signature	ECDSA Private Key	ECDSA Signature	15 (read/write/delete)
ECDSA Signature Verification	Use ECDSA Public Key to verify ECDSA Signature	ECDSA Public Key	Status of commands	16 (read)
AES Encryption / Decryption	Use AES key to encrypt / decrypt	AES key	outputs, status, and data	18 (read/write/delete)
Triple-DES[1] Encryption / Decryption	Use Triple-DES key to encrypt / decrypt	Triple-DES key	outputs, status, and data	17 (read/write/delete)
SHA-1/256/384/512 Message Digest Key Generation	Used to generate SHA-1/256/384/512 message digest output	Commands and configuration data	outputs, status, and data	None (no CSP access)
HMAC-SHA-1/256/384/512 Message Authentication Code Generation	Use HMAC-SHA-1/256/384/512 Key to generate HMAC-SHA-1/256/384/512 Message Authentication Code output	HMAC-SHA-1/256/384/512 key	outputs, status, and data	19 (read/write/delete)
AES-CTR DRBG Random Bit Generation	Use SP-800-90A AES-CTR DRBG with Entropy Input, V and "Key" values to generate random bits for key and IV generation	DRBG entropy input, DRBG V and 'Key' values	outputs, status, and data	1, 2, 3, 4 (read/write/delete)
Key Destruction	Destroy all CSPs. There is a context structure associated with every cryptographic algorithm available in this module. Context structures hold sensitive information such as cryptographic keys. These context structures must be destroyed via respective API calls when the application software no longer needs to use a specific algorithm any more. This API call will zeroize all sensitive information including cryptographic keys before freeing the dynamically allocated memory.	Commands and configuration data	Status of commands and configuration data	All CSPs will be destroyed.

[1] In FIPS Mode, Triple-DES is only used in the Self-Tests

6.2. User Role

The table below lists the Approved/allowed services available to the User role:

Table 4 – User Services

Service	Description	Input	Output	CSP/Algorithm Access (please see Table 6 below for details)
Show Status	Show the FIPS status of the module	command	status	N/A
Self-Tests	Run the FIPS Self-Tests	command	self-tests results	N/A

Note: No role is required to invoke the Self-Tests service by reloading the library into executable memory.

6.3. Authentication Mechanisms

As allowed by FIPS 140-2 Security Level 1, the module does not support authentication to access services. The User and Crypto-Officer roles are implicitly assumed by the entity accessing services implemented by the Module.

6.4. Services Available in Non-FIPS Mode

- All of the services that are available in FIPS mode are also available in non-FIPS mode.
- If not operating in the Approved mode as per the procedures in sections 10.1, 10.2 and 10.3, then non-Approved algorithms and/or sizes are available.

6.5. Non-Approved Services Non-Approved in FIPS Mode

- AES EAX Encryption / Decryption
- AES Message Authentication Code (MAC)
- AES XCBC Encryption / Decryption
- ARC2, ARC4 Encryption / Decryption
- Blowfish Encryption / Decryption
- DES Encryption / Decryption
- HMAC-MD5 Message Authentication Code (MAC)
- MD2, MD4, MD5 Hashes
- RSA PKCS #1 v2.1 RSAES-OAEP Encryption / Decryption
- User role Read Version service

7. Cryptographic Key Management

7.1. FIPS Approved Algorithms

The software in each module contains the following cryptographic algorithm implementations/crypto libraries to implement the different FIPS approved cryptographic algorithms that will be used for the corresponding security services supported by the module in FIPS mode:

- ancrypto.dll algorithm implementation (for Windows 64-bit)
- libancrypto.so algorithm implementation (for Android 64-bit and Linux 64-bit)
- libancrypto.a algorithm implementation (for iOS 64-bit)

Below are the detailed lists for the FIPS approved algorithms and the associated certificate implemented by each algorithm implementation.

Note that not all algorithm modes that appear on the module's CAVP certificates are utilized by the module, and the tables below list only the algorithm modes that are utilized by the module.

Table 5 – FIPS Approved Cryptographic Functions with CAVP Certificates

ancrypto.dll					
CAVP Certificate #	Algorithm	Standard	Mode/Method	Key Lengths, Curves, Moduli	Use
A2147	AES	FIPS 197, SP 800-38A, SP 800-38D	CBC, CTR (ext only, encryption only), GCM	128, 192, 256	Data Encryption/Decryption
Vendor Affirmed	CKG	SP 800-133 Rev2	N/A	N/A	Cryptographic Key Generation
A2147	CVL RSASP1 PKCS 1.5	FIPS 186-4	PKCS1 v1.5	MOD 2048	RSA Signature Primitive
A2147	DRBG	SP 800-90A	AES CTR	256	Deterministic Random Bit Generation
A2147	ECDSA	FIPS 186-4	PKG, PKV, SigGen, SigVer	P256, P384	Digital Key Generation and Verification, Digital Signature Generation and Verification
A2147	HMAC	FIPS 198-1	HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512	Key Size < Block Size	Message Authentication
A2147	KAS-SSC ECC / FFC	SP 800-56A Rev3	FFC: dhEphem, ECC: Ephemeral Unified	FFC: MODP-2048 and FC w/SHA2-384 Two-Step KDF ECC: P-256, P-384 w/SHA2-384 KAS Roles - initiator, responder	Key Agreement Scheme – Shared Secret Computation
A2147	RSA	FIPS 186-2	SHA-1, SHA2-256, SHA2-384, SHA2-512 PKCS1 v1.5	1024 (legacy SigVer only), 2048	Digital Signature Verification
A2147	RSA	FIPS 186-4	SHA-1[2], SHA2-256, SHA2-384, SHA2-512 PKCS1 v1.5	2048, 1024 (for legacy SigVer only)	Digital Key Generation, Signature Generation and Verification

[2] SHA-1 is only Approved for use with Signature Verification.

A2147	SHS	FIPS 180-4	SHA-1, SHA2-256, SHA2-384, SHA2-512 Byte Only	160, 256, 384, 512	Message Digest
A2147	Triple-DES[3]	SP 800-67	TCBC	192	Data Encryption/Decryption

7.2. Non-FIPS Approved but Allowed Cryptographic Algorithms

- RSA (key wrapping; key establishment methodology provides 112 bits of encryption strength)

7.3. Non-FIPS Approved Cryptographic Algorithms

The cryptographic module implements the following non-approved algorithms that are not permitted for use in the FIPS 140-2 mode of operations:

- AES EAX
- AES MAC (non-compliant)
- AES XCBC
- ARC2, ARC4
- Blowfish
- DES
- HMAC-MD5
- MD2, MD4, MD5
- RSA PKCS #1 v2.1 RSAES-OAEP (non-compliant)
- RSA (non-compliant less than 112 bits of encryption strength)
- Null Encryption
- Diffie-Hellman (key agreement; non-compliant less than 112 bits of encryption strength)
- EC Diffie-Hellman (key agreement; non-compliant less than 112 bits of encryption strength)
- Diffie-Hellman Group14 with SHA-256 (non-compliant)
- Diffie-Hellman Key Generation (non-compliant)

During operation, the module can switch service by service between an Approved mode of operation and a non-Approved mode of operation. The module will transition to the non-Approved mode of operation when one of the above non-Approved security functions is utilized in lieu of an Approved one. The module can transition back to the Approved mode of operation by utilizing an Approved security function. It is the responsibility of the consuming application to zeroize all keys and CSPs prior to and after utilizing these non-Approved algorithms and services.

[3] In FIPS Mode, Triple-DES is only used in the Self-Tests

8. Critical Security Parameters

The following are the Critical Security Parameters (CSPs) used in the module. The application that uses the module is responsible for appropriate destruction and zeroization of the key material. The library provides functions for key allocation and destruction which overwrites the memory that is occupied by the key information with zeros before it is deallocated. The keys used by the module for cryptographic purposes are determined by the calling application. The calling application is required to provide keys in accordance with FIPS 140-2 requirements.

Table 6 – CSPs/Keys Used in the Module

#	CSP Name	Algorithm / Key Size	Generation/Use	Entry/Output Storage	Zeroization
1	DRBG Entropy Input	SP800-90A CTR_DRBG (512 bits)	Entropy inputs to the DRBG function used to construct the DRBG seed. 64 bytes are retrieved from the entropy source (external to the module boundary) on each call by any service that requires a random number.	Entry: Plaintext, Electronic, from host OS. Output: N/A Stored in volatile RAM (plaintext).	Zeroized automatically after use
2	DRBG Seed	SP800-90A CTR_DRBG (512-bits)	Input to the DRBG that determines the internal state of the DRBG. Generated using DRBG without derivation function that includes the entropy input from the entropy source, by any service that requires a random number.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	Zeroized automatically after use
3	DRBG Key	SP800-90A CTR_DRBG (256 bits)	This key is generated internally by the host OS, retrieved by any service that requires a random number. This is the DRBG key used for SP800-90A CTR_DRBG.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	Zeroized automatically after use
4	DRBG V	SP800-90A CTR_DRBG (128 bits)	This key is generated internally by the host OS, retrieved by any service that requires a random number. Internal V value used as part of SP800-90A CTR_DRBG.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	Zeroized automatically after use

5	Diffie-Hellman Private Component	Diffie-Hellman Group 14 (384 bits)	This key is generated externally and entered via DH API. Used for establishing DH shared secret.	Entry: Plaintext, Electronic, from host OS. Output: N/A Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
6	Diffie-Hellman Public Component	Diffie-Hellman Group 14 (2048 bits)	This key is generated externally and entered via DH API. Used for establishing DH shared secret.	Entry: Plaintext, Electronic, from host OS. Output: Transmit Host Public Component during DH Exchange Stored in volatile RAM (plaintext).	N/A
7	Diffie-Hellman Shared Secret	Diffie-Hellman Group 14 (2048 bits)	Established during Diffie-Hellman Exchange.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	Zeroized after the session is closed
8	EC Diffie-Hellman Private Component	EC Diffie-Hellman Groups 19 and 20 (Curves: P-256 and P-384)	Generated internally by calling FIPS Approved DRBG during EC Diffie-Hellman Exchange. Used for establishing ECDH shared secret.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
9	EC Diffie-Hellman Public Component	EC Diffie-Hellman Groups 19 and 20 (Curves: P-256 and P-384)	Generated internally by calling FIPS Approved DRBG during EC Diffie-Hellman Exchange. Used for establishing ECDH shared secret.	Entry: Receive Client Public Component during DH Exchange. Output: Transmit Host Public Component during DH Exchange Stored in volatile RAM (plaintext).	N/A
10	EC Diffie-Hellman Shared Secret	EC Diffie-Hellman (Curves: P-256 and P-384)	Established during EC Diffie-Hellman Exchange.	Entry: N/A Output: N/A Stored in volatile RAM (plaintext).	Zeroized after the session is closed

11	RSA Private Key	RSA Private Key (2048 bits)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used to create RSA digital signatures.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
12	RSA Public Key	RSA Public Key (2048 bits)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used to verify RSA digital signatures.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	N/A
13	RSA Key Wrapping Private Key	RSA Key Wrapping Private Key (2048 bits)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used for RSA Key Wrapping decryption operation.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
14	RSA Key Wrapping Public Key	RSA Key Wrapping Public Key (2048 bits)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used for RSA Key Wrapping encryption operation.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	N/A
15	ECDSA Private Key	ECDSA suite B (P-256 and P-384 curves)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used to create ECDSA digital signatures.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
16	ECDSA Public Key	ECDSA suite B (P-256 and P-384 curves)	This key is generated internally by calling FIPS Approved DRBG in the module. This key can also be generated externally and entered by the CO. Used to verify ECDSA digital signatures.	Entry: Plaintext if generated externally Output: Plaintext Stored in volatile RAM (plaintext).	N/A

17	Triple-DES Key	Triple-DES (192 bits)	This key is generated externally and entered by the CO. This key is used during Triple-DES encryption or decryption operations.	Entry: Plaintext Output: N/A Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
18	AES Key	AES (128, 192, 256 bits)	This key is generated externally and entered by the CO. This key is used during AES encryption or decryption operations.	Entry: Plaintext Output: N/A Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.
19	HMAC Key	HMAC (192, 256, 384, 512 bits)	This key is generated externally and entered by the CO. This key is used during HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 or HMAC-SHA-512 operations.	Entry: Plaintext Output: N/A Stored in volatile RAM (plaintext).	An application program which uses the API may destroy the key. The Key Destruction service zeroizes this CSP.

Notes:

- AES GCM IV generation is performed in compliance with the Implementation Guidance A.5 scenario 2. The IV is generated internally and randomly using the Approved DRBG that is internal to the module’s boundary and has a length of 96 bits.
- The module falls under scenario 2(b) of IG 7.14 and receives at minimum 256 bits of entropy from the external entropy source. Aruba has included the following entropy caveat.

  No assurance of the minimum strength of generated keys.
- CSPs generated in FIPS mode cannot be used in non-FIPS mode, and vice versa.
- All symmetric keys and random seeds for asymmetric keys are the direct unmodified output of the DRBG.
- In FIPS Mode, Triple-DES is only used in the Self-Tests.

9. Self-Tests

The module performs Power-On Self-Tests automatically when the module is loaded into memory. In addition, the module also performs Conditional tests upon key generation or random number generation.

Input, output, and cryptographic functions cannot be performed while the Module is in a self-test or error state because the module is single-threaded and will not return to the calling application until the power-up self-tests are complete. The cryptographic module is available to perform services only after successfully completing the power-up self-tests. In the event of a self-test error, the module will log the error and will halt, subsequent calls to the module will also fail - thus no further cryptographic operations are possible. To recover from a self-test failure and/or to perform self-tests on demand, the module should be reloaded into memory.

The module performs the following **Power-On Self-Tests (POSTs)**:

- AES-CBC Encrypt and Decrypt KATs
- AES-CTR Encrypt and Decrypt KATs
- AES-GCM Encrypt and Decrypt KATs
- DH (2048) Pairwise Consistency Test
- DRBG KAT (AES-CTR DRBG Generate function tested in every instance the module is loaded in memory)
- ECDH Pairwise Consistency Test
- ECDSA Pairwise Consistency Test
- HMAC (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512) KATs
- HMAC-SHA-1 module integrity check using HMAC files
- KAS-SSC (SP 800-56A Rev3) KATs (FFC and ECC)
- RSA Encrypt and Decrypt KATs
- RSA Sign and Verify KATs
- SHS (SHA-1, SHA-256, SHA-384 and SHA-512) KATs
- Triple-DES Encrypt and Decrypt KATs.

The module performs the following **Conditional Tests**:

- CRNG Test on Approved DRBG
- ECDSA Pairwise Consistency Test
- RSA Pairwise Consistency Test
- SP800-90A Section 11.3 Health Tests for DRBG (Instantiate, Generate and Reseed)
- SP800-56A Rev3 assurances as per SP 800-56A Rev3 Sections 5.5.2, 5.6.2 and 5.6.3.

The module will log self-test results as a ‘FIPS_powerupSelfTest’ (Windows) or ‘PowerUpSelfTest’ (Linux, Android or iOS) success or fail message into a log file:

Windows Success: “`FIPS Powerup Self Finished Successfully`”
Windows Failure: “`FIPS_powerupSelfTest() failed error <code>`”

Android Success: “`Finished Mocana initialization...`”
Android failure: “`Mocana initialization error <code>`”

Linux Success: “`Power up self test passed...`”
Linux Failure: “`!!ERROR!! :Power up self test failed: <error code>`”

10. Crypto Officer Guidance and Secure Operation

This section describes how to configure, initialize and operate the cryptographic module for FIPS-Approved mode of operation. When configured, initialized, and operated per this Security Policy, the module will only operate in the FIPS-Approved mode of operation.

10.1. Software Installation

The module is not available for direct download. The module is to be installed on a single-user mode operating system platform specified in Section 2, Overview, or one where portability is maintained.

10.2. Key Destruction Service

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 API calls when the application software no longer needs to use a specific algorithm. This specific API call will zeroize all sensitive information, including cryptographic keys, before freeing the dynamically allocated memory.

10.3. Secure Operation

The Crypto Officer must ensure that the cryptographic module is always operating in a FIPS-Approved mode of operation. To assist in this requirement, the module complies with the following security rules:

- The cryptographic module shall provide two distinct roles - the Cryptographic Officer role and the User role.
- The cryptographic module does not provide any operator authentication.
- The cryptographic module shall encrypt/decrypt message traffic using AES algorithms.
- The cryptographic module shall perform all self-tests described in Section 9, Self-Tests.
- The cryptographic module is available to perform services only after successfully completing the power-up self-tests.
- At any time, the operator shall be capable of commanding the module to perform the power-up self-tests by reloading the cryptographic module into memory.
- Data output shall be inhibited during key generation, self-tests, zeroization, and error states.
- Status information shall not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
- The module shall not support concurrent operators.
- The module is a cryptographic library and it is intended to be used with a calling application. The calling application is responsible for the usage of the primitives in the correct sequence. The application that uses the module is responsible for appropriate destruction and zeroization of the key material.
- The module relies on an entropy source external to the module boundary.
- The keys used by the module for cryptographic purposes are determined by the calling application. The calling application is required to provide keys in accordance with FIPS 140-2 requirements.
- The module logs must be monitored. If a strange activity is found, the Crypto Officer should take the module off line and investigate.
- The guidelines in this Security Policy's sections 6.5, 7.3, and 10 must be adhered to.

11. User Guidance

The module is not distributed as a standalone library and is only used in conjunction with Aruba Networks solutions. As such, there is no direct User Guidance.

11.1. Aruba Documentation from the Aruba Support Portal

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

For example, the Aruba Virtual Intranet Access (VIA) version 4.x product uses the Aruba VIA Cryptographic Module. Full Aruba Virtual Intranet Access (VIA) documentation for VIA 4.x can be found at the link provided below.

https://asp.arubanetworks.com/downloads;fileTypes=DOCUMENT;products=Aruba%20Virtual%20Intranet%20Access%20%28VIA%29;softwareMajorVersions=4

12. Mitigation of Other Attacks

The module does not contain additional security mechanisms beyond the requirements for FIPS 140-2 Security Level 1 cryptographic modules.