Aruba Mobility Controller Virtual Appliances

with ArubaOS FIPS Firmware Non-Proprietary Security Policy FIPS 140-2 Level 1

Version 2.5

September 2023

Open Source Code

Certain Aruba 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, a Hewlett Packard Enterprise company 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 Networks, Inc. from any and all legal actions that might be taken against it with respect to infringement of copyright on behalf of those vendors.

www.arubanetworks.com

6280 America Center Dr
San Jose, CA, USA 95002

Phone: 408.227.4500

Fax 408.227.4550

Contents

Preface

This security policy document can be copied and distributed freely.

1 Purpose of this Document

This release supplement provides information regarding the Aruba Mobility Controllers Virtual Appliances with ArubaOS FIPS Firmware with FIPS 140-2 Level 1 validation from Aruba Networks. The material in this supplement modifies the general Aruba firmware documentation included with this product and should be kept with your Aruba product documentation.

This supplement primarily covers the non-proprietary Cryptographic Module Security Policy for the Aruba Mobility Controllers Virtual Appliances with ArubaOS FIPS Firmware. 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:

http://csrc.nist.gov/groups/STM/cmvp/index.html

In addition, in this document, the Aruba Mobility Controllers Virtual Appliances with ArubaOS FIPS Firmware are referred to as the controller, VMC, or the module.

1.1 Related Documents

The following items are part of the complete installation and operations documentation included with this product:

- ArubaOS 8.X.0.0 Virtual Appliance Installation Guide
- ArubaOS 8.X.0.0 User Guide
- ArubaOS 8.X.0.0 CLI Reference Guide
- ArubaOS 8.X.0.0 Getting Started Guide
- ArubaOS 8.X.0.0 Migration Guide

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

2 Overview

Work environments are transforming to digital workplaces, with billions of mobile workers using their mobile devices to access applications designed to make them more productive. Mobility and IoT – along with these business critical applications – drive increasing demand on the network. At the same time, these mobile workers expect a consistent user experience from their mobile device and applications where ever they are. To enable an always-on network with the desired performance for high density

environments, enterprises must deliver a wireless network that accommodates these requirements and yet provides them with the efficiency and flexibility of a Virtual Machine (VM) deployment to move at the speed of their business. ArubaOS 8 provides new features and capabilities that include the introduction of the Mobility Controller Virtual Appliance (MC-VA). The MC-VA functions in a similar way to the 72xx and 7xxx Mobility Controllers by centralizing wireless network visibility and control. Deployed as a Virtual Appliance (VA), this controller provides plenty of capacity and speed for BYOD and 802.11 ac devices for both campus or branch deployment. The other innovation in ArubaOS 8 is the Aruba Mobility Master – the next generation of master controller that is needed to configure and manage your cluster of mobility controllers, whether virtualized or appliance-based. The Mobility Controller Virtual Appliance can be deployed as standalone or managed by Mobility Master where it can support up to 100K users. Organizations with virtualized initiatives can take advantage of the following benefits of the Mobility Controller Virtual Appliance: Flexible deployment and ease of operation Customers have the flexibility of deploying MC-VA as a VA benefiting from ease of operation and deployment to remote. The VA form factor makes it easy to dynamically scale to support the needs of a rapidly growing enterprise, enabling a much more efficient use of resources by adding more CPU and storage resources. By moving to a VA-based deployment that has more memory and compute, more services can be managed on the network. MC-VA can be deployed on VMware ESXi or open source KVM hypervisor

The Aruba Mobility Master is the next generation of master controller that can be either deployed as a virtual machine (VM) or installed on an x86-based hardware appliance. The Mobility Master provides better user experience, flexible deployment, simplified operations and enhanced performance. Existing Aruba customers can migrate their master controller configuration and licenses over to the Mobility Master and start taking advantage of these unique capabilities. Aruba Mobility Controller Virtual Appliance can support over 32,000 wireless devices and performs stateful firewall policy enforcement at speeds up to 40 Gbps – plenty of capacity for BYOD (Bring Your Own Device) and 802.11ac devices. Fully application-aware, the module prioritizes mobile apps based on user identity and offers exceptional scale for BYOD transactions and device densities.

Customers have the flexibility of deploying a VM or an x86-based hardware appliance depending on their environment and needs. Customers who already have a VM environment can benefit from ease of operation and right-size their VM by adjusting their CPU or memory. Moving to a VM-based deployment that has more memory and compute allows you to manage more services on the network. The Virtual Mobility Controller can run on open source KVM or VMware ESXi hypervisor.The module configurations validated during the cryptographic module testing included:

The module configurations validated during the cryptographic module testing included:

- The firmware version is **ArubaOS 8.10.0.2-FIPS**

Aruba's development processes are such that future releases under AOS 8.10 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:

- ESXi 6.5 running on HPE ProLiant ML110 Gen10 with an Intel Xeon Silver with AES-NI
- ESXi 6.5 running on HPE ProLiant ML110 Gen10 with an Intel Xeon Silver without AES-NI

The virtual appliances included in this validation are:

- JY902AAE, ARUBA MC-VA-50 (US) CNTRLR LIC 50 AP Aruba MC-VA-50 Mobility Controller Virtual Appliance License (US) with Support for up to 50 AP E-LTU
- JY903AAE, ARUBA MC-VA-250 (US) CNTRLR LIC 250 AP Aruba MC-VA-250 Mobility Controller Virtual Appliance License (US) with Support for up to 250 AP E-LTU

- JY904AAE, ARUBA MC-VA-1K (US) CNTRLR LIC 1000 AP Aruba MC-VA-1K Mobility Controller Virtual Appliance License (US) with Support for up to 1000 AP E-LTU

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.

- HPE EdgeLine 20, Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz
- DTECH M3-SE-SVR4, Intel(R) Xeon(R) CPU E3-1505M v6 @ 3.00GHz
- DTECH M3x, Intel(R) Core(TM) i5-7300U CPU @ 2.60GHz
- Klas Telecom TDC Blade, Intel® Xeon(R) CPU D-1541 @ 2.10GHz
- Klas Telecom VoyagerVMm, Intel(R) Core(TM) i5-5350U CPU @ 1.80GHz
- PacStar PS451-4330 Series, Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz
- PacStar PS451-1258 Series, Intel(R) Xeon(R) CPU E3-1258L v4 @ 1.80GHz
- IAS VPN Gateway Module NANO-VM, Intel(R) Atom(TM) E3900 CPU @ 1.60GHz
- IAS VPN Gateway Module Classic Plus, Intel(R) Core(TM) i7-6xxx CPU @ 3.40GHz
- Device running an equivalent Intel Atom, i5, i7, or Xeon processor on ESXi 6.5

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 Level 1 validation, the module has been tested as a multi-chip standalone firmware module. The logical cryptographic boundary is defined as the entirety of the OVA file installed on the hypervisor which contains the firmware image. The physical boundary is the surface of the computer chassis.

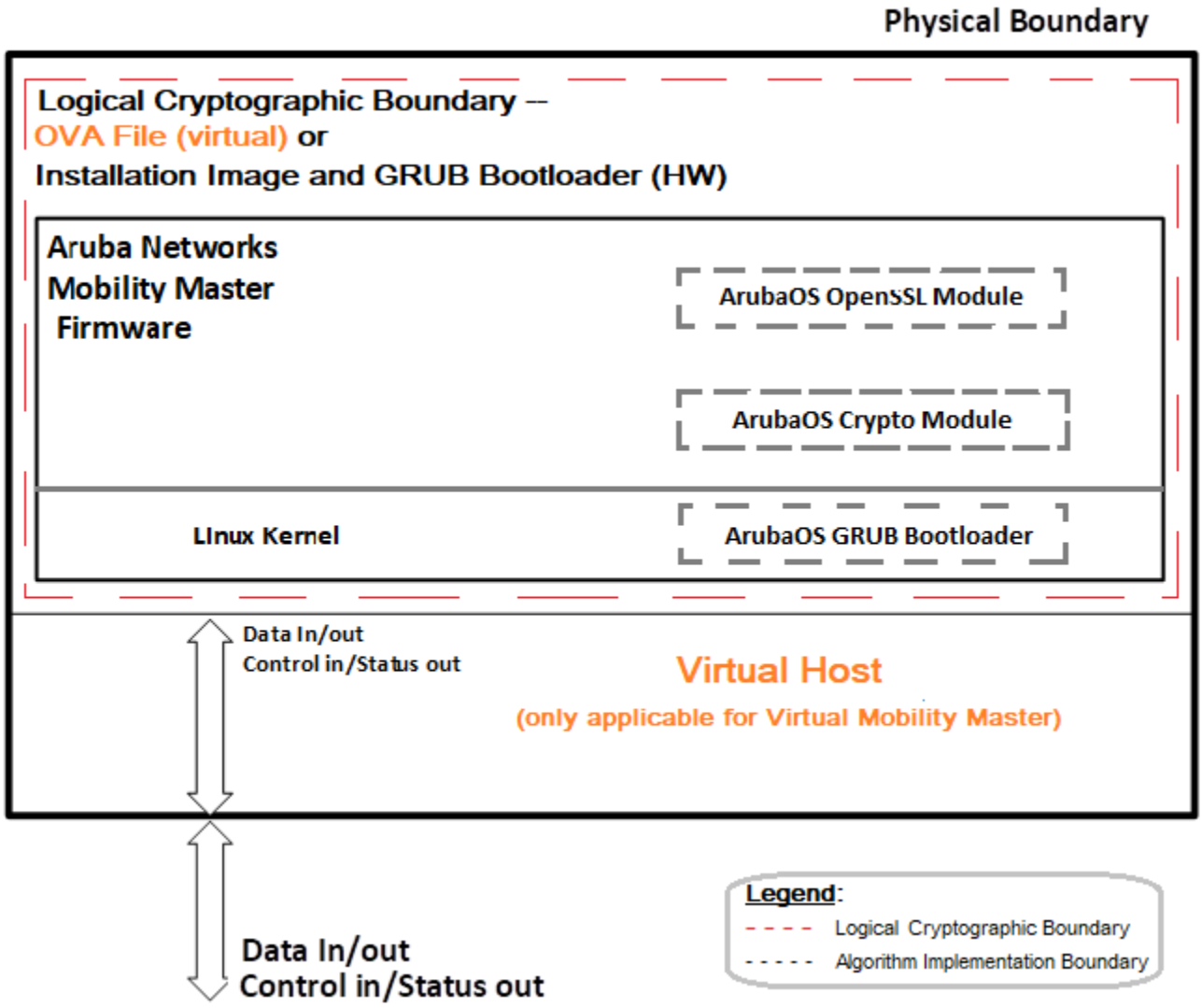

Figure 1: *Functional Block Diagram of the System Component Stack*

2.2 Intended Level of Security

The module is intended to meet overall FIPS 140-2 Level 1 requirements as shown in Table 1.

Table 1 Intended Level of Security

Section	Section Title	Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services, and Authentication	2
4	Finite State Model	1
5	Physical Security	1
6	Operational Environment	N/A
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 firmware module. It must be run on a production grade platform (such as a standard commercially made PC, laptop, server, etc) to meet requirements from FIPS 140-2 level 1.

4 Operational Environment

The operational environment of the Virtual appliance is limited and non-modifiable. The module was tested on Intel Xeon Silver running on ESXi 6.5. The platform used during testing met Federal Communications Commission (FCC) FCC Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) requirements for business use as defined by 47 Code of Federal Regulations, Part15, Subpart B, Class A.

5 Logical Interfaces

Interfaces on the module can be categorized as the following FIPS 140-2 logical interfaces:

- Data Input Interface
- Data Output Interface
- Control Input interface
- Status Output Interface
- Power Interface

All of these physical interfaces are separated into logical interfaces defined by FIPS 140-2, as described in the following table.

Table 2 FIPS 140-2 Logical Interfaces

FIPS 140-2 Logical Interface	Module Virtual Interface	Physical Interface
Data Input Interface	Virtual Ethernet Ports	• Host Platform Ethernet ports
Data Output Interface	Virtual Ethernet Ports	• Host Platform Ethernet Ports
Control Input Interface	Virtual Ethernet Ports	• Host Platform Ethernet Ports
Status Output Interface	Virtual Ethernet Ports	• Host Platform Ethernet Ports
Power Interface	N/A	• Host PC Power Interface

Data input and output, control input, status output, and power interface 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 different forms of data, control, and status traffic over the network ports by analyzing the packets header information and contents.

6 Roles and Services

The module supports role-based authentication, meeting level 2 requirements. 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. For additional non-security-relevant services offered by the module, please refer to the ArubaOS User Guide listed in section 8.7.

6.1 Crypto Officer Role

The Crypto Officer role has the ability to configure, manage, and monitor the module. This role can be present on the controller in a standalone configuration or provided through the Aruba Mobility Master when the controller is operating as a managed device. Four management interfaces can be used for this purpose:

- SSHv2 CLI

The Crypto Officer can use the CLI to perform non-security-sensitive and security-sensitive monitoring and configuration. The CLI can be accessed remotely by using the SSHv2 secured management session over the Virtual Ethernet ports.

- Web Interface

The Crypto Officer can use the Web Interface as an alternative to the CLI. The Web Interface provides a highly intuitive, graphical interface for a comprehensive set of controller management tools. The Web Interface can be accessed from a TLS-enabled Web browser using HTTPS (HTTP over TLS) on logical port 4343.

- SNMP v3

The Crypto Officer can also use SNMPv3 to remotely perform non-security-sensitive monitoring and use ‘get’ and ‘getnext’ commands.

- Mobility Master

The Crypto Officer can use the Mobility Master interface to configure the controller when operating as a managed device.

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

Table 3 Crypto-Officer Services

Service	Description	Input	Output	CSP/Algorithm Access (see Table 9 below for details)
SSHv2	Provide authenticated and encrypted remote management sessions while using the CLI	SSHv2 key agreement parameters, SSH inputs, and data	SSHv2 outputs and data	25, 26 (read/write/delete)
SNMPv3	Provides ability to query management information	SNMPv3 requests	SNMPv3 responses	31, 32, 33, 34, 35 (read/write/delete)
IKEv1/IKEv2-IPSec	Access the module's IPSec services in order to secure network traffic	IKEv1/IKEv2 inputs and data; IPSec inputs, commands, and data	IKEv1/IKEv2 outputs, status, and data; IPSec outputs, status, and data	17 (read) 5, 6, 7, 8, 9, 10 (read/write/delete) 18, 19, 20, 21, 22, 23, 24 (read/delete)
Configuring Network Management	Create management Users and set their password and privilege level; configure the SNMP agent	Commands and configuration data	Status of commands and configuration data	31, 32, 33, 35 (read) 34 (delete)
Configuring the module	Define synchronization features for module	Commands and configuration data	Status of commands and configuration data	None
Configuring Internet Protocol	Set IP functionality	Commands and configuration data	Status of commands and configuration data	None
Configuring Quality of Service (QoS)	Configure QOS values for module	Commands and configuration data	Status of commands and configuration data	None
Configuring VPN	Configure Public Key Infrastructure (PKI); configure the Internet Key Exchange (IKEv1/IKEv2) Security Protocol; configure the IPSec protocol	Commands and configuration data	Status of commands and configuration data	17 (read) 13, 14, 15, 16 (read) 18, 19, 20, 21, 22, 23, 24 and 25 (delete)
Configuring DHCP	Configure DHCP on module	Commands and configuration data	Status of commands and configuration data	None

Table 3 Crypto-Officer Services

Service	Description	Input	Output	CSP/Algorithm Access (see Table 9 below for details)
Configuring Security	Define security features for module, including Access List, Authentication, Authorization and Accounting (AAA), and firewall functionality	Commands and configuration data	Status of commands and configuration data	11, 12 (read/write/delete)
Manage Certificates	Install and delete X.509 certificates	Commands and configuration data; Certificates and keys	Status of certificates, commands, and configuration	13, 14, 15, 16 (write/delete)
NTP Authentication Service	Configure and connect to NTP server using authentication key	Commands and data	NTP output, status, and data	41 (write/delete)
HTTP over TLS	Secure browser connection over Transport Layer Security acting as a Crypto Officer service (web management interface)	TLS inputs, commands, and data	TLS outputs, status, and data	5, 6, 7, 27, 28, 29 and 30 (read/write/delete) 3, 4 (read/write) 1, 2 (read)
Openflow Agent	Agent run on device for use with Mobility Master SDN. Leveraged by the SDN for discovering of hosts and networks, configuration of networks, and collection of statistics.	Configuration Data and statistic collection	Status of commands and configuration data	None
Status Function	Cryptographic officer may use CLI "show" commands or view WebUI via TLS to view the controller configuration, routing tables, and active sessions; view health, temperature, memory status, voltage, and packet statistics; review accounting logs, and view physical interface status	Commands and configuration data	Status of commands and configurations	None

Table 3 Crypto-Officer Services

Service	Description	Input	Output	CSP/Algorithm Access (see Table 9 below for details)
IPSec tunnel establishment for RADIUS protection	Provided authenticated/encrypted channel to RADIUS server	IKEv1/IKEv2 inputs and data; IPSec inputs, commands, and data	IKEv1/IKEv2 outputs, status, and data; IPSec outputs, status, and data	11 and 17 (read/write/delete) 18, 19, 20, 21, 22, 23, 24 and 25 (write/delete) 3, 4 (read/write) 1, 2 (read)
Self-Test	Perform FIPS start-up tests on demand	None	Error messages logged if a failure occurs	None
Configuring Bypass Operation	Configure bypass operation on the module	Commands and configuration data	Status of commands and configuration data	None
Updating Firmware	Updating firmware on the module. Any firmware loaded into this module that is not shown on the module certificate is out of the scope of this validation and requires a separate FIPS 140-2 validation.	Commands and configuration data	Status of commands and configuration data	1, 40 (read)
Configuring Online Certificate Status Protocol (OCSP) Responder	Configuring OCSP responder functionality	OCSP inputs, commands, and data	OCSP outputs, status, and data	25, 26, 27, 28 and 29 (read)
Configuring Control Plane Security (CPSec)	Configuring Control Plane Security mode to protect communication with APs using IPSec and issue self signed certificates to APs. Hybrid CPsec allows for the ability to enable or disable independently for each zone and allow zones to contain different configurations. Can interact with hardware and virtual appliances through multizone/mesh when CPsec is enabled.	Commands and configuration data, IKEv1/IKEv2 inputs and data; IPSec inputs, commands, and data	Status of commands, IKEv1/ IKEv2 outputs, status, and data; IPSec outputs, status, and data and configuration data, self signed certificates	12 and 27 (read/write/delete) 18, 19, 20, 21, 22, 23, 24 (write/delete) 3, 4 (read/write) 1, 2 (read)

Table 3 Crypto-Officer Services

Service	Description	Input	Output	CSP/Algorithm Access (see Table 9 below for details)
Zeroization	The cryptographic keys stored in SDRAM memory can be zeroized by rebooting the module. The cryptographic keys (IKEv1 Pre-shared key and WPA2 Pre-Shared Key) stored in the flash can be zeroized by using the command 'wipe out flash' or overwriting with a new secret. The 'no' command in the CLI can be used to zeroize IKE, IPSec and CA CSPs. Please See CLI guide for details. The other keys/CSPs (RSA/ECDSA public key/private key and certificate) stored in Flash memory can be zeroized by using the command "wipe out flash".	Command	Progress information	All CSPs will be destroyed.

6.2 User Role

Table 4 below lists the services available to User role:

Table 4 User Service

Service	Description	Input	Output	CSP Access (see Table 9 below for CSP details)
IKEv1/IKEv2-IPSec	Access the module's IPSec services in order to secure network traffic	IPSec inputs, commands, and data	IPSec outputs, status, and data	6, 7, 8, 9, 10, 11 (read, write, delete) 13, 14, 15, 16 (read) 18, 19, 20, 21, 22, 23, 24 (read/delete) 4, 5 (read/write) 2, 3 (read)
HTTP over TLS	Access the module's TLS services in order to secure network traffic	TLS inputs, commands, and data	TLS outputs, status, and data	6, 7, 8, 9, 10, 11, 27, 28, 29 and 30 (read/write/delete) 4, 5 (read/write)

Service	Description	Input	Output	CSP Access (see Table 9 below for CSP details)
				2, 3 (read)
802.11i Shared Key Mode	Access the module's 802.11i services in order to secure network traffic	802.11i inputs, commands and data	802.11i outputs, status and data	36, 37, 38 and 39 (create/read/delete) 4, 5 (read/write)
802.11i with EAP-TLS	Access the module's 802.11i services in order to secure network traffic	802.11i inputs, commands and data	802.11i outputs, status, and data	14, 15, 16, 17 (read) 36, 37, 38 and 39 (read/delete) 4, 5 (read/write)

6.3 Unauthenticated Services

The Aruba VMC can perform VLAN, bridging, firewall, routing, and forwarding functionality without authentication. These services do not involve any cryptographic processing.

- Internet Control Message Protocol (ICMP) service
- Network Time Protocol (NTP) service
- Network Address Resolution Protocol (ARP) service

Additional unauthenticated services include performance of the power-on self-tests.

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 8, then non-Approved algorithms and/or sizes are available.
- Debugging via the virtual console port (non-approved).
- For additional non-security-relevant services offered by the module, please refer to the ArubaOS User Guide listed in section 13.5.

6.5 Non-Approved Services Non-Approved in FIPS Mode

- WPA3
- WPA-2 Multiple Pre-Shared Key (MPSK), where every client connected to the WLAN SSID may have its own unique PSK.
- IPSec/IKE using Triple-DES
- SSH using HMAC-SHA-256
- SSH using Diffie-Hellman Group 14 with SHA-256

6.6 Authentication Mechanisms

The module supports role-based authentication. Role-based authentication is performed before the Crypto Officer enters privileged mode using admin password via Web Interface or SSHv2. Role-based authentication is also performed for User authentication.

This includes password and RSA/ECDSA-based authentication mechanisms. The strength of each authentication mechanism is described below.

Table 5 Estimated Strength of Authentication Mechanisms

Authentication Type	Role	Strength
Password-based authentication (SSH and Web Interface)	Crypto Officer	Passwords are required to be a minimum of eight ASCII characters and a maximum of 32 with a minimum of one letter and one number. Given these restrictions, the probability of randomly guessing the correct sequence is one (1) in 3,608,347,333,959,680 (this calculation is based on the assumption that the typical standard American QWERTY computer keyboard has 10 Integer digits, 52 alphabetic characters, and 32 special characters providing 94 characters to choose from in total. The calculation should be 94^8 (Total number of 8-digit passwords) – 84^8 (Total number of 8-digit passwords without numbers) – 42^8 (Total number of 8-digit passwords without letters) + 32^8 (Total number of 8-digit passwords without letters or numbers, added since it is double-counted in the previous two subtractions) = 3,608,347,333,959,680). At optimal network conditions (assuming 1ms round-trip latency), an attacker would only get 60,000 guesses per minute. Therefore the associated probability of a successful random attempt during a one-minute period is 60,000/3,608,347,333,959,680, which is less than 1 in 100,000 required by FIPS 140-2.
RSA-based authentication (IKEv1/IKEv2/TLS/EAP-TLS/RADIUS)	User	The module supports 2048-bit RSA key authentication during IKEv1, IKEv2, TLS, and EAP-TLS. RSA 2048 bit keys correspond to 112 bits of security. Assuming the low end of that range, the associated probability of a successful random attempt is 1 in 2^112, which is less than 1 in 1,000,000 required by FIPS 140-2. At optimal network conditions (assuming 1ms round-trip latency), an attacker would only get 60,000 guesses per minute. Therefore the associated probability of a successful random attempt during a one-minute period is 60,000/2^112, which is less than 1 in 100,000 required by FIPS 140-2.
RSA-based authentication (SSH/HTTP over TLS)	Crypto Officer	The module supports 2048-bit RSA key authentication during IKEv1, IKEv2, TLS, and EAP-TLS. RSA 2048 bit keys correspond to 112 bits of security. Assuming the low end of that range, the associated probability of a successful random attempt is 1 in 2^112, which is less than 1 in 1,000,000 required by FIPS 140-2. At optimal network conditions (assuming 1ms round-trip latency), an attacker would only get 60,000 guesses per minute. Therefore the associated probability of a successful random attempt during a one-minute period is 60,000/2^112, which is less than 1 in 100,000 required by FIPS 140-2. These keys can be used for admin authentication.
ECDSA-based authentication	User	ECDSA signing and verification is used to authenticate to the module during IKEv1/IKEv2, TLS, and EAP-TLS. Both P-256 and P-384 curves are supported. ECDSA P-256 provides 128 bits of equivalent security, and P-384 provides 192

(IKEv1/IKEv2/TLS/EAP-TLS/RADIUS)		bits of equivalent security. Assuming the low end of that range, the associated probability of a successful random attempt is 1 in 2^128, which is less than 1 in 1,000,000 required by FIPS 140-2. At optimal network conditions (assuming 1ms round-trip latency), an attacker would only get 60,000 guesses per minute. Therefore the associated probability of a successful random attempt during a one-minute period is 60,000/2^128, which is less than 1 in 100,000 required by FIPS 140-2.
ECDSA-based authentication (SSH/HTTP over TLS)	Crypto Officer	ECDSA signing and verification is used to authenticate to the module during HTTP over TLS. Both P-256 and P-384 curves are supported. ECDSA P-256 provides 128 bits of equivalent security, and P-384 provides 192 bits of equivalent security. Assuming the low end of that range, the associated probability of a successful random attempt is 1 in 2^128, which is less than 1 in 1,000,000 required by FIPS 140-2. At optimal network conditions (assuming 1ms round-trip latency), an attacker would only get 60,000 guesses per minute. Therefore the associated probability of a successful random attempt during a one-minute period is 60,000/2^128, which is less than 1 in 100,000 required by FIPS 140-2. These keys can be used for admin authentication.
Pre-shared key-based authentication (RADIUS)	User	The password requirements are the same as the CO role above, except that the maximum ASCII characters can be 128. Assuming the weakest option of 8 ASCII characters, the authentication mechanism strength is the same as the Password-based authentication above.
Pre-shared key-based authentication (IKEv1/IKEv2)	User	The password requirements are the same as the CO role above, except that the maximum ASCII characters can be 64. Additionally, exactly 64 HEX characters can be entered. Assuming the weakest option of 8 ASCII characters, the authentication mechanism strength is the same as the Password-based authentication above.
Pre-shared key based authentication (802.11i)	User	The password requirements are the same as the IKEv1/IKEv2 shared secret above, except that the maximum ASCII characters can be 63. Assuming the weakest option of 8 ASCII characters, the authentication mechanism strength is the same as the IKEv1/IKEv2 shared secret above.
SSH Master Public Certificate (SSH)	Crypto Officer	RSA-based certificate used for authentication by the CO to connect to the Mobility Master which provides interface to the controller if running as a managed device. Same authentication mechanism strength as RSA-based authentication above.

6.7 Cryptographic Algorithms and Key Management

6.7.1 Implemented Algorithms

The 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:

- ArubaOS OpenSSL Module algorithm implementation
- ArubaOS Crypto Module algorithm implementation
- ArubaOS Bootloader algorithm implementation.

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

The module supports the following cryptographic implementations.

Table 6 Cryptographic Algorithms implemented by ArubaOS OpenSSL Module

CAVP Certificate #	Algorithm	Standard	Mode/Method	Key Lengths, Curves, Moduli	Use
A2690	AES	FIPS 197, SP 800-38A	ECB, CBC, CTR (ext only, encryption only)	128, 192, 256	Data Encryption/Decryption
A2690	AES	FIPS 197, SP 800-38A, SP 800-38D	GCM, CCM	128, 256	Data Encryption/Decryption
Vendor Affirmed	CKG	SP 800-133	CTR_DRBG	N/A	Cryptographic Key Generation (using output from DRBG[1] as per IG D.12)
A2690	CVL IKEv1, TLS, SSH, SNMP	SP800-135	IKEv1: DSA, PSK TLS: v1.0/1.1, v1.2	IKEv1: DH 2048-bit; SHA-256, SHA-384 SSH: SHA-1 TLS: SHA-256, SHA-384, SHA-512	Key Derivation
A2690	CVL IKEv1	SP800-135	IKEv1	IKEv1: SHA-1	Key Derivation
A2690	DRBG	SP 800-90A	AES CTR	256	Deterministic Random Number Generation
A2690	DSA	FIPS 186-4	keyGen, pqgGen	L=2048, N=256, SHA2-256	Key Generation, Digital Key Generation

[1] Resulting symmetric keys and seeds used for asymmetric key generation are unmodified output from SP 800-90A DRBG.

A2690	ECDSA	186-4	KeyGen, KeyVer, SigGen, SigVer	KeyGen: P-256, P-384 KeyVer: P-256, P-384 SigGen: P-256, P-384 with SHA2-256, SHA2-384, SHA2-512 SigVer: P-256, P-384 with SHA-1, SHA2-256, SHA2-384, SHA2-512	Key Generation and Verification, Digital Signature Generation and Verification
A2690	HMAC	FIPS 198-1	HMAC-SHA1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512	Key Size < Block Size	Message Authentication
A2690	KAS-SSC	SP 800-56A Rev3	FFC: dhEphem, ECC: Ephemeral Unified	FFC: FC with SHA2-256 ECC: P-256 with SHA2-256 KAS Roles - initiator, responder	Key Agreement Scheme – Shared Secret Computation
N/A	KAS	SP 800-56A Rev3 SP 800-135	KAS-SSC Cert A2690 CVL Cert A2690	N/A	Key Agreement Scheme – IG D.8, scenario X1 (2)
N/A	KAS	SP 800-56A Rev3 SP 800-56C Rev1	KAS-SSC Cert A2690 KDA Cert A2690	N/A	Key Agreement Scheme – IG D.8, scenario X1 (2)
A2690	KDA	SP 800-56C Rev1	Two-step key derivation	HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384	Key Derivation Algorithm

A2690	KBKDF	SP 800-108	CTR	HMAC-SHA1, HMAC-SHA256, HMAC-SHA384	Deriving Keys
A2690	RSA	FIPS 186-2	SigVer: SHA-1[2], SHA2-256, SHA2-384, SHA2-512 PKCS1 v1.5	1024 (for legacy SigVer only), 2048	Digital Signature Verification
A2690	RSA	FIPS 186-4	KeyGen, SigGen: SHA2-256, SHA2-384, SHA2-512 PKCS1 v1.5 SigVer: SHA-1[3], SHA2-256, SHA2-384, SHA2-512 PKCS1 v1.5	KeyGen: 2048 SigGen: 2048 SigVer: 1024 (for legacy SigVer only), 2048	Key Generation, Digital Signature Generation and Verification
A2690	SHS	FIPS 180-4	SHA-1, SHA-256, SHA-384, SHA-512 Byte Only	160, 256, 384, 512	Message Digest
A2690	Triple-DES	SP 800-67	ECB, CBC	192	Data Encryption/Decryption
AES Cert A2690	KTS	SP 800-38F	AES-GCM[4]	128, 256	Key Wrapping/Key Transport via IKE/IPSec
AES Cert A2690and HMAC Cert A2690	KTS	SP 800-38F	AES-CBC[5] HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	128, 192, 256 Key Size < Block Size	Key Wrapping/Key Transport via IKE/IPSec

Note:

- IKEv1, TLS, SSH and SNMP protocols have not been reviewed (apart from the KDF) or tested by the CAVP and CMVP
- In FIPS Mode, Triple-DES is only used in the Self-Tests and with the KEK.

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

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

[4] key establishment methodology provides 128 or 256 bits of encryption strength

[5] key establishment methodology provides between 128 and 256 bits of encryption strength

Table 7 Cryptographic Algorithms implemented by ArubaOS Crypto Module

CAVP Certificate #	Algorithm	Standard	Mode/Method	Key Lengths, Curves, Moduli	Use
A2689	AES	FIPS 197, SP 800-38A SP 800-38D	CBC, GCM	128, 192, 256	Data Encryption/Decryption
A2689	CVL IKEv2 (KDF)	SP800-135	IKEv2	IKEv2: DH 2048-bit; SHA-256, SHA-384	Key Derivation
A2689	CVL IKEv2 (KDF)	SP800-135	IKEv2	IKEv2: SHA-1	Key Derivation
A2689	DSA	FIPS 186-4	keyGen, pqgGen	L=2048, N=256, SHA2-256	Key Generation, Digital Key Generation
A2689	ECDSA	186-4	PKG, PKV, SigGen, SigVer	P256, P384	Digital Key Generation, Digital Key Veritifcation, Signature Generation and Verification
A2689	HMAC	FIPS 198-1	HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 HMAC-SHA-1-96, HMAC-SHA-256-128, HMAC-SHA-384-192 (In FIPS Mode, HMAC-SHA-512 is only used in the Self-Tests)	Key Size < Block Size	Message Authentication
A2689	KAS-SSC	SP 800-56A Rev3	FFC: dhEphem, ECC: Ephemeral Unified	FFC: FC with SHA2-256 ECC: P-256 with SHA2-256 KAS Roles - initiator, responder	Key Agreement Scheme – Shared Secret Computation
N/A	KAS	SP 800-56A Rev3 SP 800-135	KAS-SSC Cert A2689 CVL Cert. A2689	N/A	Key Agreement Scheme – IG D.8, scenario X1 (2)
A2689	RSA	FIPS 186-4	SHA-1, SHA-256, SHA-384 PKCS1 v1.5	2048	Digital Key Generation, Signature Generation and Verification

A2689	SHS	FIPS 180-4	SHA-1, SHA-256, SHA-384, SHA-512 Byte Only (In FIPS Mode, SHA-512 is only used in the Self-Tests)	160, 256, 384, 512	Message Digest
A2689	Triple-DES	SP 800-67	TCBC	192	Data Encryption/Decryption
AES Cert A2689	KTS	SP 800-38F	AES-GCM[6]	128, 256	Key Wrapping/Key Transport via IKE/IPSec
AES Cert A2689 and HMAC Cert A2689	KTS	SP 800-38F	AES-CBC[7] HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-5128	128, 192, 256 Key Size < Block Size	Key Wrapping/Key Transport via IKE/IPSec

Notes:

- In FIPS Mode, Triple-DES is only used in the Self-Tests.
- IKEv2 protocols have not been reviewed (apart from the KDF) or tested by the CAVP and CMVP.

Table 8 – Cryptographic Algorithms implemented by ArubaOS Bootloader

ArubaOS Bootloader					
CAVP Certificate #	**Algorithm**	**Standard**	**Mode/Method**	**Key Lengths, Curves, Moduli**	**Use**
A2688	RSA	FIPS 186-4	SHA-1, SHA-256	2048	Digital Signature Verification
A2688	SHS	FIPS 180-4	SHA-1, SHA-256 Byte Only	160, 256	Message Digest

Note:

- Only Firmware signed with SHA-256 is permitted in the Approved mode. Digital signature verification with SHA-1, while available within the module, shall only be used while in the non-Approved mode.

[6] key establishment methodology provides 128 or 256 bits of encryption strength

[7] key establishment methodology provides between 128 and 256 bits of encryption strength

[8] In FIPS Mode, HMAC-SHA-512 is only used in the Self-Tests.

6.7.2 Non-FIPS Approved but Allowed Cryptographic Algorithms

- MD5 (used for older versions of TLS)
- NDRNG (used solely to seed the approved DRBG)
- RSA (key wrapping; key establishment methodology provides 112 bits of encryption strength)

6.7.3 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:

- DES
- HMAC-MD5
- MD5
- RC4
- RSA (non-compliant less than 112 bits of encryption strength)
- Null Encryption
- ECDSA (non-compliant when using 186-2 signature generation)
- Triple-DES as used in IKE/IPSec
- HMAC-SHA-256 as used in SSH
- Diffie-Hellman Group14 with SHA-256

Notes:

- DES, MD5, HMAC-MD5 and RC4 are used for older versions of WEP in non-FIPS mode

6.8 Critical Security Parameters

The following are the Critical Security Parameters (CSPs) used in the module.

Table 9 CSPs/Keys Used in the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
General Keys/CSPs					
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 gotten from the entropy source on each call by any service that requires a random number.	Stored in SDRAM memory (plaintext)	Zeroized by rebooting the module
2	DRBG seed	SP800-90a CTR_DRBG (384-bits)	Input to the DRBG that determines the internal state of the DRBG. Generated using DRBG derivation function that includes the entropy input from the entropy source by any service that requires a random number	Stored in SDRAM memory (plaintext)	Zeroized by rebooting the module
3	DRBG Key	SP800-90a CTR_DRBG (256 bits)	This is the DRBG key used for SP800-90a CTR_DRBG.	Stored in SDRAM memory (plaintext)	Zeroized by rebooting the module
4	DRBG V	SP800-90a CTR_DRBG V (128 bits)	Internal V value used as part of SP800-90a CTR_DRBG	Stored in SDRAM memory (plaintext)	Zeroized by rebooting the module
5	Diffie-Hellman private key	Diffie-Hellman Group 14 (224 bits)	Generated internally by calling FIPS approved DRBG (cert #C413) during Diffie-Hellman Exchange. Used for	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
			establishing DH shared secret.		
6	Diffie-Hellman public key	Diffie-Hellman Group 14 (2048 bits)	Generated internally by calling FIPS approved DRBG (cert #C413) during Diffie-Hellman Exchange. Used for establishing DH shared secret.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
7	Diffie-Hellman shared secret	Diffie-Hellman Group 14 (2048 bits)	Established during Diffie-Hellman Exchange. Used for deriving IPSec/IKE and SSH cryptographic keys.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
8	EC Diffie-Hellman private key	EC Diffie-Hellman (Curves: P-256 or P-384).	Generated internally by calling FIPS approved DRBG (cert #C413) during EC Diffie-Hellman Exchange. Used for establishing ECDH shared secret.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
9	EC Diffie-Hellman public key	EC Diffie-Hellman (Curves: P-256 or P-384).	Generated internally by calling FIPS approved DRBG (cert #C413) during EC Diffie-Hellman Exchange. Used for establishing ECDH shared secret.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
10	EC Diffie-Hellman shared secret	EC Diffie-Hellman (Curves: P-256 or P-384)	Established during EC Diffie-Hellman Exchange. Used for deriving IPSec/IKE and TLS cryptographic keys.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
11	RADIUS server shared secret	8-128 characters shared secret	Entered by CO role. Used for RADIUS server authentication.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
12	Crypto Officer Password	8-32 characters password	Entered by CO role. Used for CO role authentication.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
13	RSA Private Key	RSA 2048 bit private key	This key is generated by calling FIPS approved DRBG (cert #C413) in the module. Used for IKEv1, IKEv2, SSH, TLS, OCSP (signing OCSP messages) and EAP-TLS peers authentication. This key can also be entered by the CO.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash'
14	RSA public key	RSA 2048 bits public key	This key is generated by calling FIPS approved DRBG (cert #C413) in the module. This Key can also be entered by the CO. Used for IKEv1, IKEv2, SSH, TLS, OCSP (verifying OCSP messages) and EAP-TLS peers authentication.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash'
15	ECDSA Private Key	ECDSA suite B P-256 and P-384 curves	This key is generated by calling FIPS approved DRBG (cert #C413) in the module. Used for IKEv1, IKEv2, TLS and EAP-TLS peers authentication. This key can also be entered by the CO.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash'

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
16	ECDSA Public Key	ECDSA suite B P-256 and P-384 curves	This key is generated by calling FIPS approved DRBG (cert #C413) in the module. This Key can also be entered by the CO. Used for IKEv1, IKEv2, TLS and EAP-TLS peers authentication.	Stored in Flash memory (plaintext)	Zeroized by using command 'wipe out flash'.
IPSec/IKE					
17	IKE Pre-shared secret	Shared secret (8 - 64 ASCII or 64 HEX characters)	Entered by CO role. Used for IKEv1and IKEv2 peers authentication.	Stored in Flash memory (plaintext).	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
18	skeyid	Shared Secret (160/256/384 bits)	A shared secret known only to IKE peers. It was established via key derivation function defined in SP800-135 KDF (IKEv1). Used for deriving other keys in IKE protocol implementation.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module.
19	skeyid_d	Shared Secret (160/256/384 bits)	A shared secret known only to IKE peers. It was derived via key derivation function defined in SP800-135 KDF (IKEv1). Used for deriving IKE session authentication key.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
20	SKEYSEED	Shared Secret (160/256/384 bits)	A shared secret known only to IKE peers. It was derived via key derivation function defined in SP800-135 KDF (IKEv2) and it will be used for deriving IKE	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
			session authentication key.		
21	IKE session authentication key	HMAC-SHA-1/256/384 (160/256/384 bits)	The IKE session (IKE Phase I) authentication key. This key is derived via key derivation function defined in SP800-135 KDF (IKEv1/IKEv2). Used for IKEv1/IKEv2 payload integrity verification.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
22	IKE session encryption key	AES (128/192/256 bits, CBC)	The IKE session (IKE Phase I) encrypt key. This key is derived via key derivation function defined in SP800-135 KDF (IKEv1/IKEv2). Used for IKE payload protection.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
23	IPSec session encryption key	AES (128/192/256 bits, CBC) and AES-GCM (128/256 bits)	The IPsec (IKE phase II) encryption key. This key is derived via a key derivation function defined in SP800-135 KDF (IKEv1/IKEv2). Used for IPSec traffics protection.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
24	IPSec session authentication key	HMAC-SHA-1 (160 bits)	The IPsec (IKE Phase II) authentication key. This key is derived via using the KDF defined in SP800-135 KDF (IKEv1/IKEv2). Used for IPSec traffics integrity verification.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
SSHv2					
25	SSHv2 session key	AES (128/192/256 bits) CBC Mode, CTR Mode	This key is derived via a key derivation function defined in SP800-135 KDF (SSHv2). Used for SSHv2 traffics protection.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
26	SSHv2 session authentication key	HMAC-SHA-1, HMAC-SHA1-96 (160-bit)	This key is derived via a key derivation function defined in SP800-135 KDF (SSHv2). Used for SSHv2 traffics integrity verification.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
TLS					
27	TLS pre-master secret	48 bytes secret	This key is transferred into the module, protected by TLS RSA public key.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
28	TLS master secret	48 bytes secret	This key is derived via the key derivation function defined in SP800-135 KDF (TLS) using the TLS Pre-Master Secret.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
29	TLS session encryption key	AES-CBC Mode (128/256 bits), AES-GCM Mode (128/256 bits)	This key is derived via a key derivation function defined in SP800-135 KDF (TLS). Used for TLS traffics protection. Uses Triple-DES when using TLSv1.0	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
30	TLS session authentication key	HMAC-SHA-1/256/384 (160/256/384 bits)	This key is derived via a key derivation function defined in SP800-135 KDF (TLS). Used for	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
			TLS traffic integrity verification.		
SNMPv3					
31	SNMPv3 authentication password	8-31 characters password	Entered by CO role. User for SNMPv3 authentication.	Stored in Flash memory (plaintext).	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
32	SNMPv3 Authentication Key	AES-CFB key (128 bits)	This key is derived via a key derivation function defined in SP800-135 KDF (SNMPv3). Used for SNMPv3 authentication.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
33	SNMPv3 engine ID	10 - 24 hex character password	Entered by CO role. A unique string used to identify the SNMP engine.	Stored in Flash memory (plaintext).	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
34	SNMPv3 privacy key	AES-CFB key (128 bits)	This key is derived via a key derivation function defined in SP800-135 KDF (SNMPv3). Used for SNMPv3 traffics protection.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
35	SNMPv3 Privacy Protocol Password	8 - 31 characters password	Entered by CO role. A unique string used to protect SNMP privacy protocol.	Stored in Flash memory (plaintext).	Zeroized by using command 'wipe out flash' or by overwriting with a new secret
802.11i					
36	802.11i Pre-shared secret	Shared secret (8-63 ASCII or 64 HEX characters)	Entered by CO role. Used for 802.11i client/server authentication	Stored in Flash memory (plaintext).	Zeroized by using command 'wipe out flash' or by overwriting with a new secret

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
37	802.11i Pair-Wise Master key (PMK)	Shared secret (256 bits)	The PMK is transferred to the module, protected by IPSec secure tunnel. Used to derive the Pairwise Transient Key (PTK) for 802.11i communications.	Stored in SDRAM (plaintext).	Zeroized by rebooting the module
38	802.11i Pairwise Transient Key (PTK)	HMAC (384 bits)	This key is used to derive WPA2/WPA3 session key by using the KDF defined in SP800-108 and SP800-56C Rev1.	Stored in SDRAM memory (plaintext)	Zeroized by rebooting the module
39	802.11i session key	AES-CCM (128 bits)	Derived during WPA2/WPA3 4-way handshake by using the KDF defined in SP800-108 and SP800-56C Rev1 then used as the session key.	Stored in SDRAM memory (plaintext).	Zeroized by rebooting the module
Factory Key					
40	Firmware Integrity Public Key	RSA (2048 bits)	This is an RSA public key which is loaded into the module during manufacturing. Read in by bootloader during image verification.	Stored in firmware image (plaintext).	The zeroization requirements do not apply to this key as it is a public key.
NTP					
41	NTP Authentication Key	SHA-1 (160-bit)	Entered by CO role. A unique string used for authentication to the NTP server.	Stored in Flash memory (plaintext).	Zeroized by using command ‘wipe out flash’ or by deleting the NTP configuration.
Mobility Master					
42	Master Public Certificate	RSA (2048 bits)	This key is generated by calling FIPS approved DRBG (cert #C413) in the module.	Stored in Flash memory (plaintext).	Zeroized by using command ‘wipe out flash’

#	Name	Algorithm/Key Size	Generation/Use	Storage	Zeroization
			Used for SSH to the Mobility Master when connecting to the controllers for management.		

- AES GCM IV generation is performed in compliance with the Implementation Guidance A.5 scenario 1. The AES-GCM IV is used in the TLS and IPSec/IKEv2 protocols.
    - When used with TLS, it is internally generated deterministically in compliance with TLSv1.2 GCM cipher suites as described in SP 800-52 Rev 2, Section 3.3.1. Per RFC 5246, when the nonce_explicit part of the IV exhausts the maximum number of possible values for a given session key, the module will trigger a handshake to establish a new encryption key.
    - When used with IPSec/IKEv2, it is internally generated deterministically in compliance with RFCs 4106 and 5282. Additionally, the module uses RFC 7296 compliant IKEv2 to establish the shared secret SKEYSEED from which the AES GCM encryption keys are derived. Per RFC 7296, when the IV exhausts the maximum number of possible values for a given security association the module will trigger a rekeying with IKEv2 to establish a new encryption key for the security association.
- CKG (vendor affirmed to SP 800-133 Rev2): For keys identified as being “Generated internally by calling FIPS approved DRBG", the generated seed used in the asymmetric key generation is an unmodified output from the DRBG.
- The module generates a minimum of 256 bits of entropy for use in key generation.
- CSPs generated in FIPS mode cannot be used in non-FIPS mode, and vice versa.
- CSPs labeled as “Entered by CO” are entered into the module via SSH/TLS.

6.9 Self-Tests

The module performs Power On Self-Tests on power up. In addition, the module also performs Conditional tests after being configured into the FIPS mode. In the event any self-test fails, the module will enter an error state, log the error, and reboot automatically.

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

- ArubaOS OpenSSL Module (Firmware)
    - AES (Encrypt/Decrypt) KATs
    - DRBG KATs
    - ECDSA (P-256, P-384) (Sign/Verify) KATs
    - HMAC (HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384 and HMAC-SHA2-512) KATs
    - KAS-SSC (SP 800-56A Rev3) KATs (FFC and ECC)
    - KDA (SP 800-56C Rev1) KAT (two-step KDF with HMAC)
    - KBKDF KAT
    - KDF135 KATs (IKEv1 KDF, TLS KDF, SSH KDF, SNMP KDF)
    - RSA (2048) (Sign/Verify) KATs
    - SHS (SHA-1, SHA2-256, SHA2-384 and SHA2-512) KATs
    - Triple-DES (Encrypt/Decrypt) KATs
- ArubaOS Crypto Module (Firmware)
    - AES (Encrypt/Decrypt) KATs
    - AES-GCM (Encrypt/Decrypt) KATs
    - KAS-SSC (SP 800-56A Rev3) KATs (FFC and ECC)
    - ECDSA (Sign/Verify) KATs
    - HMAC (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512) KATs
    - RSA (Sign/Verify) KATs
    - SHS (SHA-1, SHA-256, SHA-384 and SHA-512) KATs
    - Triple-DES (Encrypt/Decrypt) KATs
- ArubaOS Bootloader (Firmware)
    - Firmware Integrity Test: RSA PKCS#1 v1.5 (2048 bits) signature verification with SHA-256 (the integrity test is the KAT)

The module performs the following Conditional Tests:

- ArubaOS OpenSSL Module (Firmware)
    - Bypass Tests (Wired Bypass Test and Wireless Bypass Test)
    - CRNG Test on Approved DRBG
    - CRNG Test for NDRNG

    - ECDSA Pairwise Consistency Test
    - Firmware Load Test - RSA PKCS#1 v1.5 (2048 bits) signature verification with SHA-256
    - RSA Pairwise Consistency Test
    - SP800-90A Section 11.3 Health Tests for CTR_DRBG (Instantiate, Generate and Reseed)
    - DSA Pairwise Consistency Test
    - SP800-56A Rev3 assurances as per SP 800-56A Rev3 Sections 5.5.2, 5.6.2 and 5.6.3.
- ArubaOS Crypto Module (Firmware)
    - ECDSA Pairwise Consistency Test
    - RSA Pairwise Consistency Test
    - Diffie-Hellman Pairwise Consistency Test
    - DSA Pairwise Consistency Test
    - SP800-56A Rev3 assurances as per SP 800-56A Rev3 Sections 5.5.2, 5.6.2 and 5.6.3.

Upon successful completion of the power-up self tests, the module logs a KATS: passed message to the console.

6.10 Alternating Bypass State

The module implements an alternating bypass state when:

- If the VLAN is one that is associated with an IPSec map, then traffic will be encrypted, otherwise it will not be
- a configuration provides wireless access without encryption

The alternating bypass status can be identified by retrieving whether or not the VLAN association is with an IPSec map, or the wireless network configuration.

7 Installing the Module

This chapter covers the installation of the Mobility Controller Virtual Appliances with FIPS 140-2 Level 1 validation. The Crypto Officer is responsible for ensuring that the following procedures are used to install the module properly.

This chapter covers the following installation topics:

- Requirements for the module components
- Selecting a proper environment for the module
- Install the module on the hypervisor server
- Power on the module using virtual machine management client

7.1 Pre-Installation Checklist

You will need the following during installation:

- Aruba Mobility Controller Virtual Appliances Controller components (host server, VM Host SW and Aruba Mobility Controller Virtual Appliances installation disk).
- Cool, non-condensing air 0 to 40 ºC (32 to 104 ºF). May require air conditioning.
- Management Station (PC) with 10/100 Mbps Ethernet port and virtual machine management client software.

7.1.1 Product Examination

The units are shipped to the Crypto Officer in factory-sealed boxes using trusted commercial carrier shipping companies. The Crypto Officer should examine the carton for evidence of tampering. Tamper-evidence includes tears, scratches, and other irregularities in the packaging.

7.1.2 Package Contents

The product carton should include the following:

- Aruba Mobility Controller Virtual Appliances Series Virtual Controller OVA file
- Aruba User Documentation

8 Ongoing Management

The Aruba Mobility Controller Virtual Appliances meet FIPS 140-2 Level 1 requirements. The information below describes how to keep the controller in FIPS-approved mode of operation. The Crypto Officer must ensure that the controller is kept in a FIPS-approved mode of operation.

8.1 Crypto Officer Management

The Crypto Officer must ensure that the controller is always operating in a FIPS-approved mode of operation. This can be achieved by ensuring the following:

- The admin role must be root.
- Passwords must be at least eight characters long.
- VPN services can only be provided by IPsec or L2TP over IPsec.
- Access to the controller Web Interface is permitted only using HTTP over a TLS tunnel. Basic HTTP and HTTP over SSL are not permitted.
- Only SNMP read-only may be enabled.
- Only FIPS-approved algorithms can be used for cryptographic services (such as HTTPS, L2, AES-CBC, SSH, and IKEv1/IKEv2-IPSec), which include AES, Triple-DES, SHA-1, HMAC SHA-1, and RSA signature and verification.
- TFTP can only be used to load backup and restore files. These files are: Configuration files (system setup configuration), the WMS database (radio network configuration), and log files. (FTP and TFTP over IPsec can be used to transfer configuration files.)
- The controller logs must be monitored. If a strange activity is found, the Crypto Officer should take the controller off line and investigate.
- The 'no' command in the CLI can be used to zeroize IKE, IPsec and CA CSPs. Please See CLI guide for details.
- All configuration performed through the Mobility Master when configured as a managed device must ensure that only the approved algorithms and services are enabled on the FIPS-enabled controller.
- The guidelines in Sections 6.6.3 and 8 of this SP must be adhered to.

8.2 User Guidance

The User accesses the VMC VPN functionality as an IPsec client. The user can also access the controller 802.11i functionality as an 802.11 client. Although outside the boundary of the VMC, the User should be directed to be careful not to provide authentication information and session keys to others parties.

8.3 Setup and Configuration

The Aruba Mobility Controller Virtual Appliances meet FIPS 140-2 Level 1 requirements. The sections below detail the FIPS-approved mode of operation.

8.4 Setting Up Your Virtual Controller

To set up your controller:

1. Make sure that the module is not connected to any device on your network.
2. Boot up the module.
3. Connect your PC or workstation to a physical port mapped to the module interface.

Follow the procedures as discussed in the *ArubaOS Virtual Appliance Installation Guide*. The link to this document can be found in Section 8.7 below.

When running as a managed device:

1. Make sure that the controller is connected only to the Mobility Master on your network.
2. Boot up the controller.
3. Connect to the Mobility Master.

Follow the procedures as discussed in the *ArubaOS Virtual Appliance Installation Guide*. The link to this document can be found in Section 8.7 below.

8.5 Enabling FIPS Mode

For FIPS compliance, users cannot be allowed to access the VMC until the CO changes the mode of operation to FIPS mode.

Login to the VMC using an SSHv2 client. Enable FIPS mode using the following commands:

```
#configure terminal
```

Enter Configuration commands, one per line. End with CNTL/Z

```
(config) #fips enable
(config) #exit
#write memory
Saving Configuration...
Configuration Saved.
```

To verify that FIPS mode has been enabled, issue the command "`show fips`".

Once FIPS mode is enabled, the CO should also ensure that the serial port has been disabled through the following command (the serial port must be disabled while operating under the FIPS approved mode of operation):

```
(config) #mgmt-user console-block
```

8.6 Disallowed FIPS Mode Configurations

When you enable FIPS mode, the following configuration options are disallowed:

- All WEP features
- WPA
- TKIP mixed mode
- Any combination of DES, MD5, and PPTP
- Firmware images signed with SHA- 1
- Enhanced PAPI Security
- Null Encryption
- TLS with Diffie-Hellman Group 2
- Certificates with less than 112 bits security strength as used with IKEv1, IKEv2, IPSec, TLS/EAP-TLS, SSH, and/or user authentication.
- Telnet
- EAP-TLS Termination

- Diffie-Hellman Group14 with SHA-256.
- IPSec/IKE using Triple-DES
- SSH using HMAC-SHA-256
- WPA3
- WPA-2 PSK

In addition to the above options, use of backups (via the backup command) are only permitted under FIPS mode if the backup is immediately transferred out of the module via SCP and then deleted from flash.

8.7 Full Documentation

Full documentation can be found at the link provided below.

https://asp.arubanetworks.com/downloads;fileTypes=DOCUMENT;products=Aruba%20Mobility%20Controllers%20%28AOS%29