F5, Inc.

BIG-IP Tenant Cryptographic Module

Module Version: 17.1.0.1

FIPS Security Level 2

FIPS 140-3 Non-Proprietary Security Policy

Last update: January 2025

Prepared by:
atsec information security corporation
4516 Seton Center Parkway, Suite 250
Austin, TX 78759
www.atsec.com

Table of Contents

Copyrights and Trademarks

F5®, BIG-IP®, and TMOS®, are registered trademarks of F5, Inc.

Intel®, Atom® and Xeon® are registered trademarks of Intel Corporation.

1 General

1.1 Description

This document is the non-proprietary FIPS 140-3 Security Policy for the BIG-IP Tenant Cryptographic Module with firmware version 17.1.0.1. The document contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for a Security Level 2 module.

This document provides all tables and diagrams (when applicable) required by NIST SP 800-140B.

1.2 Security Levels

ISO/IEC 24759 Section 6. [Number Below]	FIPS 140-3 Section Title	Security Level
1	General	2
2	Cryptographic Module Specification	2
3	Cryptographic Module Interfaces	2
4	Roles, Services, and Authentication	2
5	Software/Firmware Security	2
6	Operational Environment	N/A
7	Physical Security	2
8	Non-Invasive Security	N/A
9	Sensitive Security Parameter Management	2
10	Self-Tests	2
11	Life-Cycle Assurance	2
12	Mitigation of Other Attacks	N/A

Table 1 - Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use: The BIG-IP Tenant Cryptographic Module (hereafter referred to as “the module”) is a smart evolution of F5’s market leading Application Delivery Controller (ADC) technology, and specifically designed for F5 hardware and the underlying platform layer. Traffic Management Operating System (TMOS) is the foundation and architecture for F5’s ADCs running on the BIG-IP platform. Together, BIG-IP hardware and the firmware components TMOS is a highly optimized system providing control over the acceleration, security, and management through purpose-built hardware and software systems. F5OS platform layer is tightly integrated with F5’s TMOS firmware. In the following documentation TMOS and BIG-IP are interchangeably used where system and feature modules are concerned.

Module Type: Firmware

Module Embodiment: Multi Chip Standalone

2.2 Operating Environments

Operating system	Hardware Platform	Processors	PAA/ Acceleration
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.5.1	r4800	Intel® Atom® P5342 Snow Ridge	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.5.1	r5900	Intel® Xeon® Silver 4314 Ice Lake	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.5.1	r5920-DF	Intel® Xeon® Silver 4314 Ice Lake	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.5.1	r10900	Intel® Xeon® Gold 6312U Ice Lake	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.5.1	r10920-DF	Intel® Xeon® Gold 6312U Ice Lake	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-A 1.7.0	r12900-DS	Intel® Xeon® Platinum 8351N Ice Lake	N/A
BIG-IP 17.1.0.1 Tenant on F5OS-C 1.6.0	VELOS CX410 BX110	Intel® Xeon® D-2177NT Skylake	N/A

Table 2 - Tested Operating Environments

2.3 Modes of Operation

The module supports two modes of operation:

- in Approved mode of operation only approved or vendor affirmed security functions can be used.
- in non-Approved mode of operation only non-approved security functions can be used.

The module enters operational mode after pre-operational and conditional algorithms self-tests succeed. The module automatically switches between the approved and non-approved modes depending on the services requested by the operator. The status indicator of the mode of operation is equivalent to the indicator of the service that was requested. SSPs used or stored in the Approved mode are not used in the non-Approved mode, and vice versa.

In the Approved Mode, the cryptographic module provides the cryptographic algorithms whose CAVP certificates are in Table 3 below. The Control (or Management) Plane refers to the connection from an administrator to the BIG-IP for system management. The Data Plane refers to the traffic passed between external entities and internal servers.

Not all the ACVP tested capabilities are used by the module in approved mode of operation.

2.4 Algorithms

2.4.1 Approved Algorithms and Vendor Affirmed Algorithms

CAVP Cert		Algorithm and Standard	Mode / Method	Description / Key Size(s)/ Key Strength(s)	Use / Function
Control Plane	Data Plane				
A3729	N/A	AES [FIPS 197, SP800-38A, SP800-38C, SP800 38D]	ECB, CBC, GCM, CCM, CTR	128 / 192 / 256-bit keys with key strengths from 128 to 256 bits	Encryption and decryption
A3729	A3730	KTS (AES) [FIPS 197, SP800-38D, SP800- 38F]	GCM, CCM	128 / 256-bit AES keys with key strengths 128 or 256 bits	Key wrapping / unwrapping
A3729	A3730		AES-CBC key and HMAC-SHA2-256, or HMAC-SHA2-384	128 / 256-bit AES and HMAC keys with key strengths 128 or 256 bits	
A3729	N/A		AES-CBC/ AES-CTR keys and HMAC-SHA-1, HMAC-SHA2-256	128 / 256-bit AES and HMAC keys with key strengths from 128 or 256 bits	
A3729	N/A	AES [FIPS 197, SP800-38B, SP800 38D]	GMAC	128 / 192 / 256-bit AES keys with key strengths from 128 and 256 bits	MAC generation and verification
N/A	A3730	AES [FIPS 197, SP800-38A, SP800-38C, SP800 38D]	CBC, GCM, CCM	128 / 256-bit keys with key strengths 128 and 256 bits	Encryption and decryption
N/A	A3730	AES [FIPS 197, SP800-38B, SP800 38D]	GMAC	128 / 256-bit keys with key strengths 128 and 256 bits	MAC generation and verification
A3729	N/A	CTR_DRBG [SP800-90Ar1]	AES 256 in CTR mode, with / without derivation function, prediction resistance disabled / enabled	Entropy input (256-bits), V (128-bits) and key (256-bits) values	Random number generation

CAVP Cert		Algorithm and Standard	Mode / Method	Description / Key Size(s)/ Key Strength(s)	Use / Function
Control Plane	Data Plane				
N/A	A3730	CTR_DRBG [SP800-90A r1]	AES 256 in CTR mode, with derivation function, prediction resistance disabled	Entropy input (256-bits), V (128-bits) and key (256-bits) values	Random Number Generation
A3729	N/A	RSA [FIPS 186-4]	B.3.3 Random Probable Primes	2048 and 4096-bit keys with key strengths 112 and 150-bits	Key pair generation
A3729	A3730	RSA [FIPS 186-4]	PKCS#1v1.5: SHA2-256, SHA2-384	2048, 3072 and 4096-bits keys with key strengths 112 to 150-bits	Signature generation and verification
N/A	A3730	RSA [FIPS 186-4]	PKCSPSS: SHA2-256, SHA2-384	2048, 3072 and 4096-bits keys with key strengths 112 to 150-bits	Signature generation and verification
A3729	A3730	Safe Primes key generation/ verification [SP800-56Ar3]	Safe Primes groups	ffdhe2048, ffdhe3072, and ffdhe4096 with key strengths 112 to 150-bits	Key pair generation and verification using Safe Primes
A3729	A3730	ECDSA [FIPS 186-4]	B.4.2 Testing Candidates	P-256 and P-384 with key strengths 128 and 192-bits	Key pair generation / verification
A3729	A3730	ECDSA [FIPS 186-4]	SHA2-256, SHA2-384, SHA2-512	P-256 and P-384 with key strengths 128 and 192-bits	Signature generation and verification
A3729	A3730	SHS [FIPS180-4]	SHA-1 SHA2-256 SHA2-384 SHA2-512	N/A	Message digest
A3729	A3730	HMAC [FIPS 198-1]	HMAC-SHA-1 HMAC-SHA2-256 HMAC-SHA2-384 HMAC-SHA2-512	112 bits to 1024-bits with key strengths 112 to 256-bits	Message authentication

CAVP Cert		Algorithm and Standard	Mode / Method	Description / Key Size(s)/ Key Strength(s)	Use / Function
Control Plane	Data Plane				
A3729	A3730	KAS-ECC-SSC [SP800-56Ar3]	Ephemeral Unified: KAS Role: initiator, responder	P-256, P-384 with key strengths 128 and 192-bits	Shared Secret Computation used in Key Agreement Scheme (KAS) IG D.F scenario 2 (path 2)
A3729	A3730	KAS-FFC-SSC [SP800-56Ar3]	dhEphem KAS Role: initiator, responder	ffdhe2048, ffdhe3072, ffdhe4096 with key strengths 112 to 150-bits	Shared Secret Computation used in Key Agreement Scheme KAS) IG D.F scenario 2 (path 2)
A3729 (CVL)	N/A	SSH KDF[1] [SP800-135r1]	AES-128, AES-256 with SHA2-256, SHA2-384	256-bit keys with 256-bits key strength	Key derivation (CVL)
A3729 (CVL)	A3730	TLS KDF[1] [SP800-135r1] RFC7627	TLS v1.2	256-bits	Key derivation (CVL)
(vendor affirmed)	(vendor affirmed)	CKG Section 4 example 1 [SP800-133r2] CTR_DRBG [SP800-90Ar1] Diffie-Hellman and EC Diffie-Hellman [SP800-56Ar3] RSA, ECDSA [FIPS 186-4]	DRBG produces random numbers use for key generation of asymmetric algorithms	RSA Sizes: 2048 and 4096-bits key with 112 and 150-bits key strength	Key generation
				ECDSA, EC Diffie-Hellman: P-256 and P-384 with 128 and 192-bits key strength	
				Safe Primes: ffdhe2048, ffdhe3072, ffdhe4096 with 112, 128, 150-bits key strength	

Table 3 - Approved Algorithms

[1] No parts of the TLS / SSH protocols except the KDF has been reviewed or tested by the CAVP and CMVP

2.4.2 Non-Approved, Allowed Algorithms and Non-Approved, Allowed Algorithms with No Security Claimed

There are no non-Approved algorithms allowed in the approved mode along with their usage with or without security claimed.

2.4.3 Non-Approved, Not Allowed Algorithms

The following table lists the non-Approved algorithms along with their usage.

Algorithm/ Functions	Use/ Function
AES modes: OFB, CFB, XTS and KW; AES-GCM in IPsec protocol; DES, RC4, Triple-DES, SM2, SM4	Symmetric encryption and decryption
RSA	Asymmetric encryption and decryption
RSA key generation	with modulus size other than 2048, and 4096-bit with ANSI X9.31 standard for all key sizes
DSA	domain parameter generation, domain parameter verification, key pair generation
DSA digital signature	signature generation and verification using any key size
EdDSA digital signature	signature generation and verification using Ed25519
ECDSA key generation/ verification	with curves other than P-256 and P-384
RSA digital signature	- Signature generation and verification: PKCS#1 v1.5 using 2048, 3072 or 4096-bits modulus with SHA-1, SHA2-224, SHA2-512 - Signature generation and verification using PKCS #1 v1.5 scheme with modulus other than 2048, 3072 or 4096 bits, for all SHA sizes - Signature generation and verification PSS using 2048, 3072 or 4096-bits modulus with SHA-1, SHA2-224, SHA2-512 - Signature generation and verification using Probabilistic Signature Scheme (PSS) specified in ANSI X9.31 standard
ECDSA digital signature	- Signature generation and verification using curves other than P-256 and P-384, all SHA sizes - Signature generation and Signature verification using curves P-256 and P-384 with SHA-1, SHA2-224
SHA2-224 SM3 MD5	Message digest
HMAC-SHA2-224 AES-CMAC Triple-DES	Message authentication

Algorithm/ Functions	Use/ Function
AES-GCM in IPsec protocol	
Diffie-Hellman EC Diffie-Hellman	Key Agreement Scheme: - Diffie-Hellman using groups other than ffdhe2048, ffdhe3072, ffdhe4096 - Diffie-Hellman using MODP groups in IPsec/IKE protocol - EC Diffie-Hellman ephemeral Unified using curves other than P-256 and P-384 - EC Diffie-Hellman static Unified and OnePassDh using P-256, P-384 - EC Diffie-Hellman in IPsec/IKE protocol using P-384
TLS KDF SSH KDF SNMP KDF IKEv1, IKEv2 KDF	Key derivation function in the context of: - TLS using MD5/ SHA-1/ SHA2-224 / SHA2-512 - SSH using SHA-1/ SHA2-224/ SHA2-512 - SNMP using any SHA variant - IKE using any SHA variant
TLS used in SSL Orchestrator (SSLO)	ciphersuites algorithms implemented by f5-rest-node

Table 4 - Non-Approved Not Allowed Algorithms

2.5 Module Photographs

Figures below show the platforms on which the module was tested.

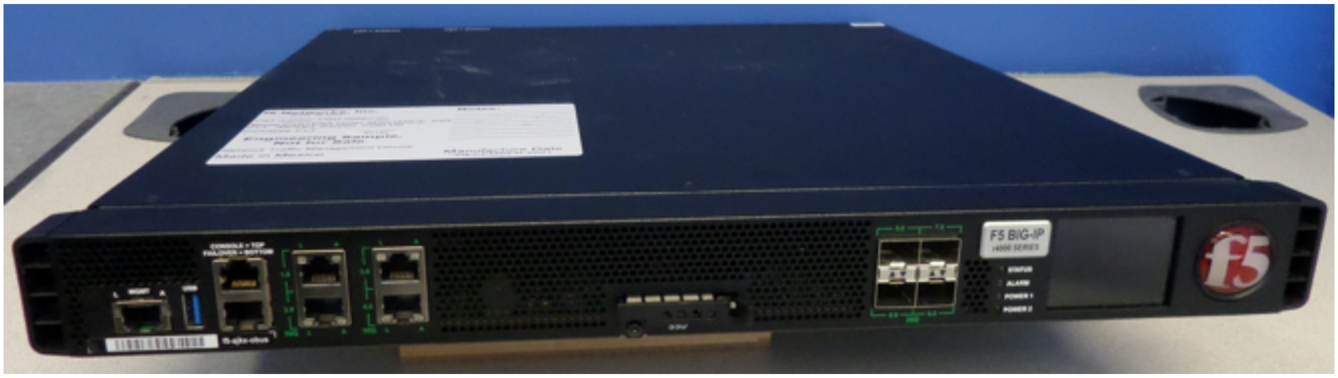

Figure 1 - r4800

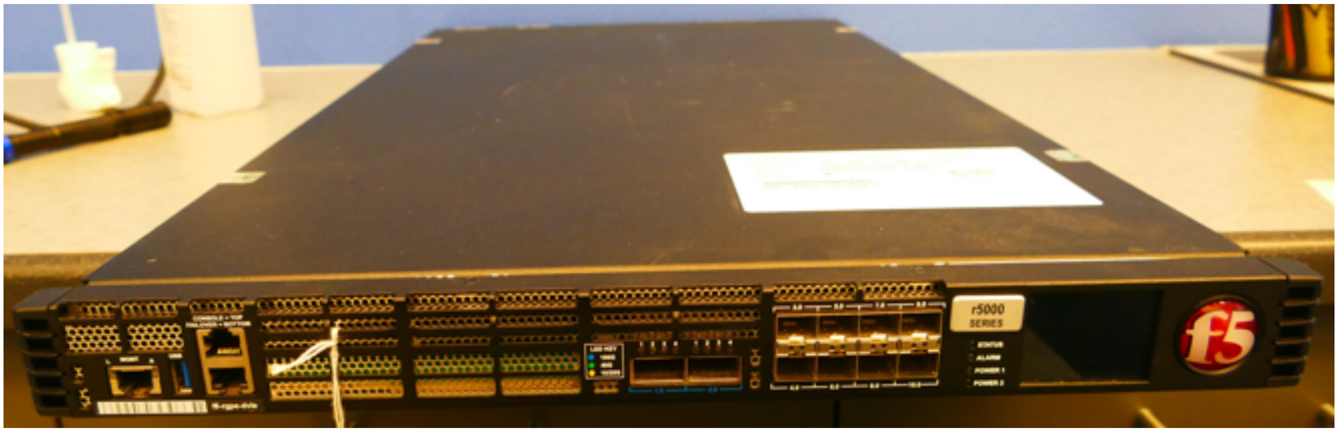

Figure 2 - r5900

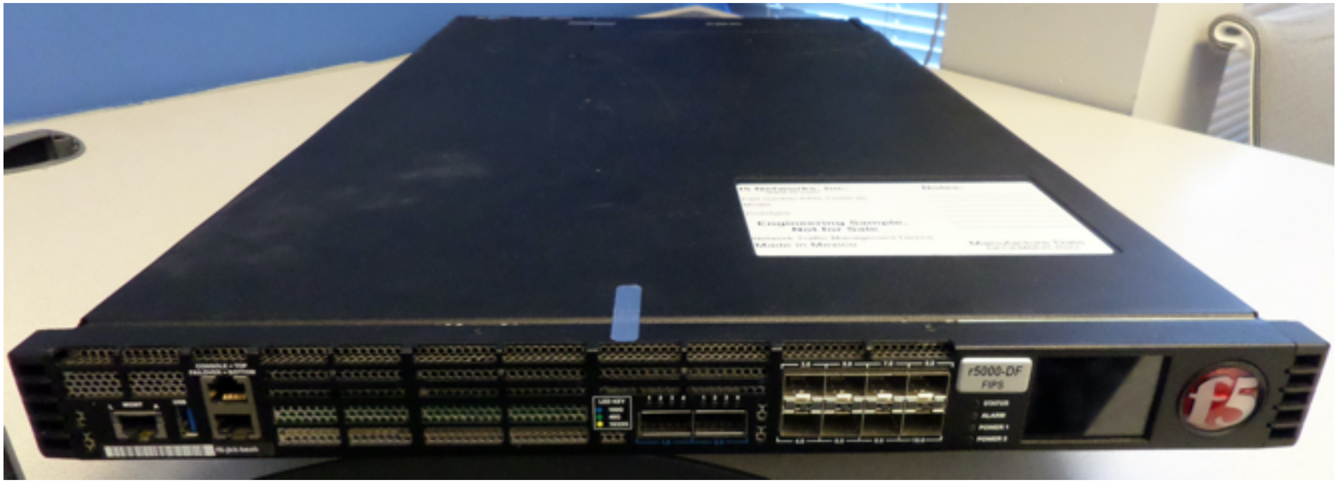

Figure 3 - r5920-DF

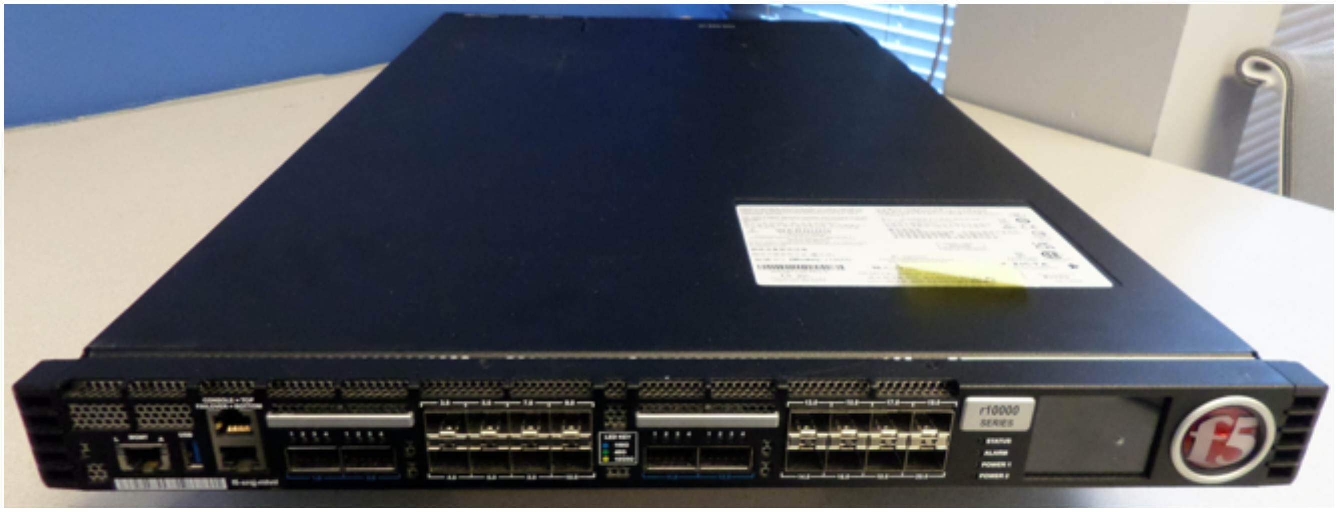

Figure 4 –r10900, r10920-DF and r12900-DS
(same chassis for the test platforms)

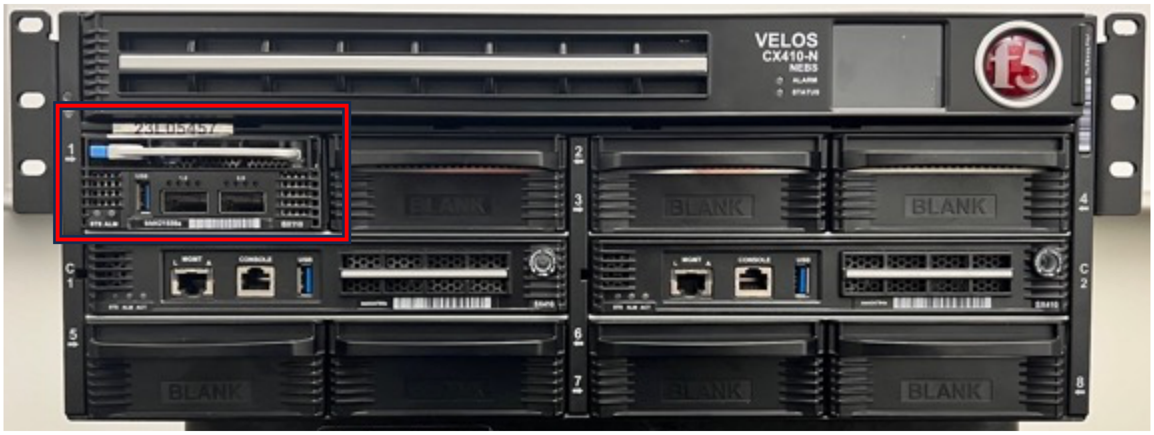

Figure 5 – VELOS CX410 BX110
with 7 filler panels and one blade. The BX110 tested blade in slot #1 delineated with red rectangle.

2.6 Block Diagram and Cryptographic Boundary Descriptions

The block diagram below shows the module cryptographic boundary, its interfaces with the host operational environment, the host platform, the flow of status output (SO), control input (CI), data input (DI) and data output (DO). The module cryptographic boundary is defined by the red dotted line in Figure 6. The TOEPP is defined by the tested platforms listed in Table 2 and delineated by the black rectangle in Figure 6. The description of the ports and interfaces can be found in Table 5.

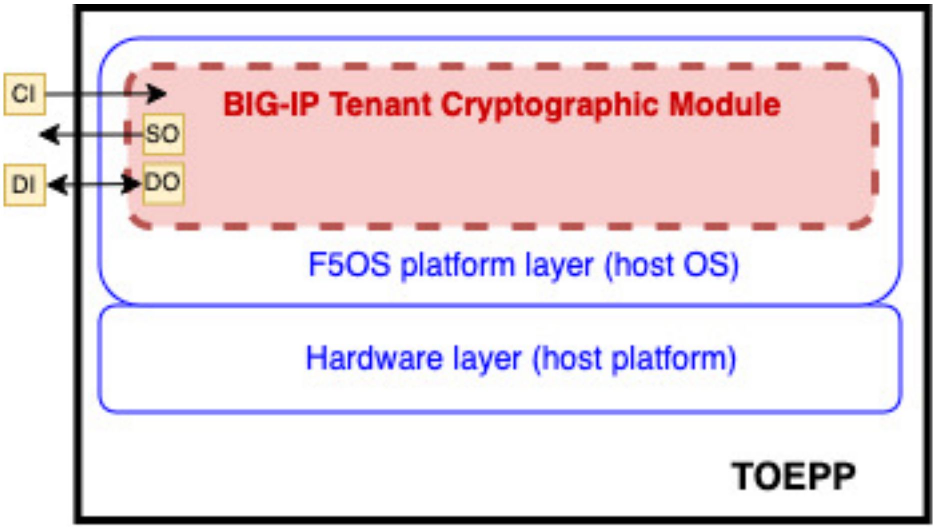
CI
SO
DI
DO
BIG-IP Tenant Cryptographic Module
F5OS platform layer (host OS)
Hardware layer (host platform)
TOEPP

Figure 6 - Block Diagram

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The logical interfaces are the commands through which users of the module request services. There are no external input or output devices to the module can be used for data input, data output, status output or control input.

For the purpose of the FIPS 140-3 validation, the physical ports are interpreted to be the physical ports of the hardware platform on which it runs.

Physical port	Logical Interface[2]	Data That Passes
N/A	Data Input	TLS/SSH protocol input messages; Configuration commands for interface management
N/A	Data Output	TLS/SSH protocol output messages; Status log
N/A	Control Input	API which control system state (e.g. reset system, power-off system)
N/A	Status Output	API which provides system status information
Power Interface	Power Input	PSU

Table 5 - Ports and Interfaces

[2] The module does not implement Control Output interface.

4 Roles, Services, and Authentication

4.1 Roles

The module supports one CO role and one User role. Maintenance role is not supported. The FIPS 140-3 roles are defined below and corresponding service with input and output are described in Table 6.

- Crypto Officer (CO) role: The Crypto Officer is represented by the administrator of the module ("administrator" is the CO). This entity performs module installation and initialization. This role has full access to the system and has the ability to create, delete, and manage other User roles on the system. At initialisation of the module, the CO is the only available role and only the CO can create the user roles.
- The FIPS140-3 User role is mapped to multiple module roles: Auditor, Certificate Manager, Firewall Manager, iRule Manager, Operator, Resource Manager and User Manager. Each of the module roles are responsible for different components of the system (e.g. auditing, certificate and key management, user management, etc.).

The list of services available to the CO and user roles are defined in Table 8 and Table 9.

FIPS 140-3 Role	Module Role	Service	Input	Output
CO User	administrator User Manager Resource Manager Auditor	List users	None	List of user accounts
CO User	administrator User Manager	Create additional User	Username, password	Confirmation of account creation
CO User	administrator User Manager	Modify existing Users	Username	Confirmation of account modification
CO User	administrator User Manager	Delete user	Username	Confirmation of deletion
CO User	administrator User Manager	Unlock user	Username	Confirmation of unlock
CO User	administrator User	Update own password	Own password	Confirmation of update of password
CO User	administrator User Manager	Update others password	Username, password	Confirmation of update
CO	administrator	Configure password policy	New password policy	Confirmation of configuration change
CO User	administrator Certificate Manager Resource Manager	Create / delete TLS key / certificate	Key/ certificate identification information	Confirmation of key/ certificate creation or deletion
CO User	administrator Auditor Certificate Manager Resource Manager	Display / log expiration data of installed certificates	List of certificates to display	Certificate expiration information
CO User	administrator Auditor Certificate Manager	List private keys	List of private keys to display	List of key metadata i.e. creation time, key size and checksum

FIPS 140-3 Role	Module Role	Service	Input	Output
	Resource Manager			
CO User	administrator Certificate Manager	Import TLS certificate	Certificate to import	Confirmation of import of certificate
CO User	administrator Certificate Manager	Export certificate file	Certificate to export	Exported Certificate file
CO User	administrator Resource Manager	SSH-keyswap	SSH key to create / delete	Confirmation of SSH key creation / deletion
CO User	administrator Firewall Manager	Configure firewall	Policy rules, address lists	Confirmation of policy configuration
CO User	administrator Firewall Manager	Show firewall state	N/A	Display the current system wide state of the firewall rules.
CO User	administrator Firewall Manager	Show statistics of firewall rules on the BIG-IP system	N/A	List of statistics of firewall rules
CO User	administrator Firewall Manager	Configure firewall users	Firewall user and configuration information	Confirmation of configuration
CO User	administrator Auditor Resource Manager	View system audit log	N/A	Display of system audit logs
CO User	administrator Auditor	Export analytics logs system	N/A	Display System Analytics Logs
CO User	administrator Resource Manager	Enable / disable audit	N/A	Confirmation of enabling or disabling of audit
CO User	administrator Resource Manager	Configure boot options	Boot options	Confirmation of configuration of boot options
CO User	administrator Resource Manager	Configure SSH access options	SSH access, IP address list	Confirmation of configuration of SSH access options
CO User	administrator Resource Manager User Manager	Configure SSH user configuration	ssh/authorized_keys file	Confirmation of configuration of SSH user configuration
CO User	administrator Operator	Modify nodes and pool members	Which nodes and pool members to modify	Confirmation of modification of nodes and pool members
CO User	administrator Firewall Manager Resource Manager	Configure nodes	List of nodes to create / modify / view / delete	Confirmation of creation / modification / display / deletion of nodes

FIPS 140-3 Role	Module Role	Service	Input	Output
CO User	administrator iRule Manager Firewall Manager Resource Manager	Configure iRules	List of iRules to create / modify/ view/ delete	Confirmation of creation / modification / display / deletion of iRules
CO	administrator	Reboot System	N/A	Confirmation of system reboot
CO	administrator	Secure Erase	Selected file	Confirmation of full system zeroization
CO User	administrator User	SSH session service	User, address, password, algorithms, key sizes	Confirmation of SSH session establishment
CO User	administrator User	Closing SSH session	N/A	Confirmation of SSH session closure
CO User	administrator User	TLS session service	Address, algorithms, keys, primary secret	Confirmation of establishment of TLS session
CO User	administrator User	Closing TLS session	N/A	Confirmation of TLS session closure
CO User	administrator User	Show version	None	Version information, and module name
CO User	administrator User	Show license	None	FIPS license information
CO User	administrator User	Show status	None	Status of the specific service passed in the show status command
CO User	administrator User	Self- test	Power	Pass/ fail results of self-tests

Table 6 - Roles, Service Commands, Input and Output

4.2 Authentication

The module supports role-based authentication. The module supports concurrent operators belonging to different roles (one CO role and one User role) which create different authenticated sessions, while achieving the separation between the concurrent operators.

Two interfaces can be used to access the module:

- CLI: The module offers a CLI called traffic management shell (tmsh) which is accessed remotely using the SSHv2 secured session over the Ethernet connection.
- Web Interface (WebUI): The Web interface consists of HTTPS over TLS-enabled web browser which provides a graphical interface for system management tools.

The User role can access the module through CLI or WebUI. However, the CO can restrict User role access to have the User accessing through WebUI only.

The module does not maintain authenticated sessions upon power cycling. Power-cycling the system requires the authentication credentials to be re-entered. When entering password authentication data through the Web interface, any character entered will be obfuscated (i.e. replace the character entered with a dot on the entry box). When entering password authentication data through the CLI, the module does not display any character entered by the operator in stdin (e.g. keyboard).

Table 7 lists the required role-based authentication method for the Crypto Office role and the User role depending upon which interface is being used.

Role	Authentication Method	Authentication Strength
Crypto Officer User	role-based authentication with Password (CLI or WebUI)	The password must consist of a minimum of 8 characters with at least one from each of the three-character classes. Character classes are defined as: digits (0-9), ASCII lowercase letters (a-z), ASCII uppercase letters (A-Z) Assuming a worst-case scenario where the password contains six numerical digits, one ASCII lowercase letter and one ASCII uppercase letter. The probability of guessing every character successfully is (1/10)^6 * (1/26)^1 * (1/26)^1 = 1/676,000,000. Note: this is less than 1/1,000,000. The maximum number of login attempts is limited to 3 after which the account is locked. This means that, in the worst case, an attacker has the probability of guessing the password in one minute as 3/676,000,000. Note: This is less than 1/100,000.
Crypto Officer User	role-based authentication with SSH ECDSA key pair (CLI only)	The ECDSA using P-256 or P-384 curves for key based authentication yields a minimum security-strength of 128 bits. The chance of a random authentication attempt falsely succeeding is at most 1/(2^{128}) that is less than 1/1,000,000. The maximum number of login attempts is limited to 1 after which the account switch to password authentication. Then the attacker probability of succeeding to establish the connection depends on the probability of guessing the password and it is, as above, 3/676,000,000 less than 1/100,000.

Table 7 - Authentication Methods

4.3 Approved Services

Table 8 lists the Approved services, the service name, description, the Approved security function being used by the service, the keys and SSPs accessed by the service, the roles used by the service, access rights to keys and SSPs and the FIPS 140-3 service indicator returned by the service.

The environment variable SECURITY_FIPS140_CIPHER_STRICT is exported with the cipher restriction status. If the cipher_restricted status is enabled, the status output from the service indicator is returned in the high speed login /var/log remote.log file as “'Service Indicator: Approved”. If the cipher_restricted status is disabled, there is no service indicator output.

For SSH service the service indicator is implicit: when the SSH connection is established the service with the cipher selected is approved.

The following variables are used in the Access rights to keys or SSPs column:

- **G = Generate**: The module generates or derives the SSP.

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

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
List users	Display list of all User accounts	N/A	N/A	CO, User Manager, Resource Manager, Auditor	N/A	None
Create additional User	Create additional User	N/A	password	CO, User Manager	W	None
Modify existing Users	Modify existing Users	N/A	N/A	CO, User Manager	N/A	None
Delete User	Delete User	N/A	N/A	CO, User Manager	N/A	None
Unlock User	Remove lock from user who has exceeded login attempts	N/A	N/A	CO, User Manager	N/A	None
Update own password	Update own password	N/A	password	CO, User	W	None
Update others password	Update others password	N/A	password	CO, User Manager	W	None
Configure Password Policy	Set password policy features	N/A	N/A	CO	N/A	None
Create TLS certificate	Self-signed certificate creation	RSA / ECDSA SigGen	RSA public and private keys 2048/ 4096 bit ECDSA public and private keys with P-256 and P-384	CO, Certificate Manager, Resource Manager	E	Service Indicator: Approved
Create TLS key	Used for the SSL Certificate key file	RSA / ECDSA KeyGen CTR_DRBG	RSA public and private keys 2048/ 4096 bit ECDSA public and private keys with P-256 and P-384	CO, Certificate Manager, Resource Manager	G	Service Indicator: Approved
			DRBG seed		E	
			DRBG internal state (V and key values)		E,W	
			Entropy input		E	

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Delete TLS certificate / key	Self-signed certificate / key deletion	N/A	RSA public and private keys 2048/ 4096 bit ECDSA public and private keys with P-256 and P-384	CO, Certificate Manager, Resource Manager	Z	None
List certificate	Display / log expiration data of installed certificates	N/A	N/A	CO, Auditor, Certificate Manager, Resource Manager	N/A	None
List private keys	List private keys	N/A	N/A	CO, Auditor, Certificate Manager, Resource Manager	N/A	None
Import TLS Certificate	Import TLS Certificate	N/A	TLS ECDSA public key with P-256 and P-384; TLS RSA public key with 2048, 3072 and 4096	CO, Certificate Manager	W	None
Export Certificate File	Export Certificate File	N/A	TLS ECDSA public key with P-256 and P-384; TLS RSA public key with 2048, 3072 and 4096	CO, Certificate Manager	R	None
Create ssh-keyswap	Utility service create ssh keys	ECDSA KeyGen CTR_DRBG	ECDSA public and private keys with P-256 and P-384 curves	CO, Resource Manager	G	Service Indicator: Approved
Delete ssh-keyswap	Utility service delete ssh keys	N/A	ECDSA public and private keys	CO, Resource Manager	Z	None
Configure Firewall	Set policy rules, and address lists for use by firewall rules.	N/A	N/A	CO, Firewall Manager	N/A	None
Show firewall state	Display the current system-wide state of firewall rules	N/A	N/A	CO, Firewall Manager	N/A	None

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Shows statistics	Shows statistics of firewall rules on the BIG-IP system	N/A	N/A	CO, Firewall Manager	N/A	None
View System Audit Log	Display logs/files of configuration changes	N/A	N/A	CO, Auditor, Resource Manager	N/A	None
Export Analytics Logs System	Export Analytics Logs System	N/A	N/A	CO, Auditor	N/A	None
Enable/ Disable Audit	Enable/ Disable Audit	N/A	N/A	CO, Resource Manager	N/A	None
Configure Boot Options	Enable Quiet boot, Manage boot locations	N/A	N/A	CO, Resource Manager	N/A	None
Configure SSH access options	Enable / Disable SSH access, Configure IP address allow list	N/A	N/A	CO, Resource Manager	N/A	None
Configure SSH user configuration	Update ssh/ authorized_keys file for user authentication	N/A	SSH ECDSA public key	CO, Resource Manager User Manager	W	None
Configure Firewall Users	Configure Firewall Users	N/A	N/A	CO, Firewall Manager	N/A	None
Modify nodes and pool members	Enable / Disable nodes and pool members	N/A	N/A	CO Operator	N/A	None
Configure nodes	Create, modify, view, delete nodes	N/A	N/A	CO Firewall Manager, Resource Manager	N/A	None
Configure iRules	Create, modify, view, delete, iRules	N/A	N/A	CO iRule Manager, Firewall Manager, Resource Manager	N/A	None

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Reboot System	Restart cryptographic module	N/A	SSPs listed in Table 12	CO	Z	None
Secure Erase	Full system zeroization	N/A	SSPs listed in Table 12	CO	Z	None
Establish SSH session	Key authentication	ECDSA	SSH ECDSA public key with P-256 and P-384 curves	CO User	W	SSH connection successful
	Password authentication	N/A	Password	CO User	W	SSH connection successful
	Key exchange	KAS-ECC-SSC P-256 / P-384	SSH EC Diffie-Hellman public key with P-256 and P-384	CO User	W	SSH connection successful
			SSH EC Diffie-Hellman private key with P-256 and P-384		E	
			SSH shared secret		G	
	Key derivation	[SP 800-135r1] SSH KDF	SSH shared secret	CO User	E	SSH connection successful
			SSH derived session key (AES, HMAC)		G	
Maintain SSH Session	Data Encryption and Decryption	AES-CBC AES-CTR	SSH derived session key (AES)	CO User	E	SSH connection successful
	Data Integrity (MAC): HMAC-with SHA-1/ SHA2-256	HMAC	SSH derived session key (HMAC)	CO User	E	SSH connection successful
Close SSH Session	Close SSH Session	N/A	SSH EC Diffie-Hellman public and private keys; SSH shared secret; SSH derived session key	CO User	Z	None
Establish TLS Session	SigGen / SigVer	ECDSA / RSA	TLS ECDSA public key with P-256 and P-384; TLS RSA public key with 2048, 3072 and 4096	CO User	R	Service Indicator: Approved

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
	Key Exchange	EC Diffie-Hellman with SP 800-135r1 TLS KDF Diffie-Hellman with SP 800-135r1 TLS KDF	TLS Diffie-Hellman private key with ffdhe2048, ffdhe3072, ffdhe4096 or TLS EC Diffie-Hellman private key with P-256 and P-384	CO User	E	Service Indicator: Approved
			TLS Diffie-Hellman public key with ffdhe2048, ffdhe3072, ffdhe4096 or TLS EC Diffie-Hellman public key with P-256 and P-384		W	
			TLS pre-primary secret		E, G	
			TLS primary secret		G	
Maintain TLS Session	Data Encryption, Data Authentication	AES-CBC with HMAC-SHA2-256 / SHA2-384 or AES-GCM, AES-CCM	TLS derived session key (AES and HMAC or authentication cipher)	CO User	E	Service Indicator: Approved
Close TLS session	Close TLS session	N/A	TLS Diffie-Hellman public and private keys; TLS EC Diffie-Hellman public and private keys; TLS pre-primary secret; TLS primary secret; TSL derived session key	CO User	Z	None
Show version	Return the module name and version	N/A	N/A	CO User	N/A	None
Show license	Return license indication	N/A	N/A	CO User	N/A	None

Service	Description	Approved Security Functions	Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Show status	Return the module status	N/A	N/A	CO User	N/A	None
Self- test	Execute integrity test, Execute the CASTs	Algorithms listed in table section 10	N/A (key for self-tests are not SSPs)	CO User	N/A	None

Table 8 - Approved Services

4.4 Non-Approved Services

Service	Description	Algorithms Accessed	Role	Indicator
Establish TLS session	Signature generation and verification	algorithms listed in Table 4 rows DSA, RSA, ECDSA, EdDSA digital signature	User / CO	No indicator
	Key exchange	- TLS KDF using MD5, SHA-1, SHA2-224, SHA2-512 - Diffie-Hellman with groups other than ffdhe2048, ffdhe3072, ffdhe4096 - EC Diffie-Hellman ephemeral Unified using curves other than P-256 and P-384 - EC Diffie-Hellman Static Unified and OnePassDh using P-256 and P-384	User / CO	No indicator
Maintain TLS session	Data encryption Data authentication	HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2-512 Triple-DES, Camellia, SEED DSA with all key and SHA sizes	User / CO	No indicator
IPsec /IKEv2	Protocol configuration	- Authentication: HMAC-SHA2-224, AES-CMAC, AES-GCM - Encryption: AES-GCM, Triple-DES - Key exchange: EC Diffie-Hellman with P-384, Diffie-Hellman using MODP groups	User / CO	FIPS-1403 Approved: No
iControl REST access	Access to the system through REST	RSA keypair with 2048, 3072 and 4096 (REST API)	User / CO	No indicator
SSLO configuration and usage	Management of the module protected by iApplx authentication	TLS used in SSLO ciphersuites implemented by f5-rest-node. java, icrd_child	User / CO	No indicator
Configuration using SNMP	Protocol configuration	SNMP KDF using any SHA variant	User / CO	No indicator

Table 9 - Non-Approved Services

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module using the approved integrity technique HMAC-SHA-384 is listed in the section 10.1.1. Integrity tests are performed as part of the Pre-Operational Self-Tests.

5.2 Initiate on Demand

The on demand pre-operational self-tests, including the integrity test on demand, are performed by powering the module off and powering it on again.

5.3 Executable Code

The executable code is defined by the firmware version 17.1.0.1. All code belonging to this firmware version is the executable code of the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

The module operates in a non-modifiable operational environment provided by F5 with a firmware version 17.1.01. Once the module is operational, it does not allow the loading of any additional firmware.

The module is a firmware validated at a Security Level 2 in Physical Security then there are no further requirements for this security area.

7 Physical Security

7.1 Mechanisms and Actions Required

The module tested in the platforms listed in Table 2 is enclosed in a hard-metallic production grade enclosure that provides opacity and prevents visual inspection of internal components. Each test platform is fitted with tamper evident labels to provide physical evidence of attempts to gain access inside the enclosure. The tamper evident labels shall be installed for the module to operate in approved mode of operation.

Physical Security Mechanism	Recommended Frequency of Inspection / Test	Inspection/Test Guidance Details
Production grade enclosure (SL1)	N/A	N/A
Opaque enclosure (SL2)	N/A	N/A
Tamper Evident Labels (SL2)	Once per month	The Crypto Officer checks the quality of the tamper-evident labels for any sign of removal, replacement, or tearing. If the tamper-evident labels require replacement, a kit providing 25 tamper labels is available for purchase (P/N: F5-ADD-BIG-FIPS140). The Crypto Officer shall be responsible for the storage of the label kits.

Table 10 - Physical Security Inspection Guidelines

7.2 Tamper Label Placement

The pictures below show the location of all tamper-evident labels for each hardware platform. Label application instructions are provided in Section 11.2.1 of the Crypto-Officer guidance below.

Hardware Appliance	Number of Tamper Labels	Number of opacity screen
r4800	5	1 (blank in PSU slot)
r5900	4	1 (blank in PSU slot)
r5920-DF	5	1 (blank in PSU slot)
r10900 r10920-DF r12900-DS	5	0
VELOS CX410 BX110	1	7 blanks in blade slot #2-8

Table 11 – Number of tamper evident labels and blanks/ filler panels per hardware appliance

The tamper labels are delineated with red circles in the pictures below.

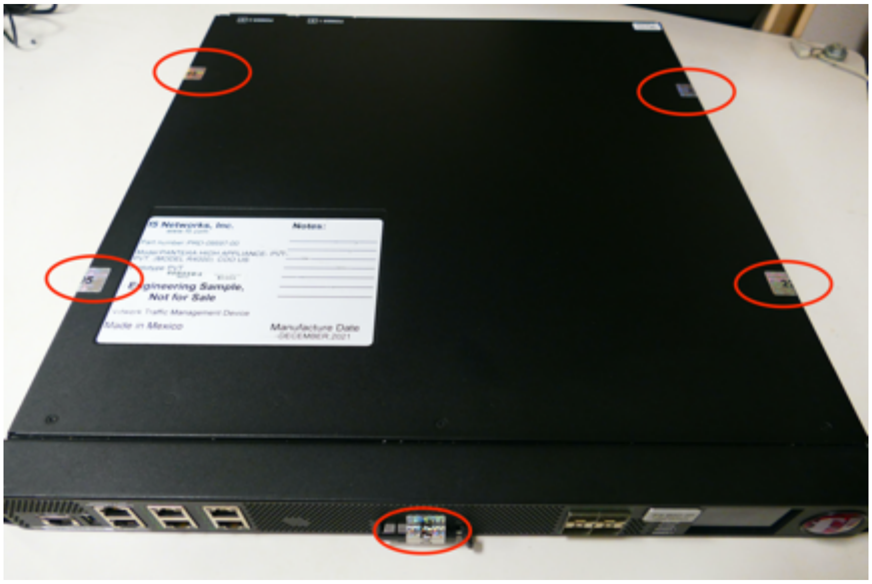

Figure 7 - Tamper labels on r4800 (5 of 5 tamper labels)

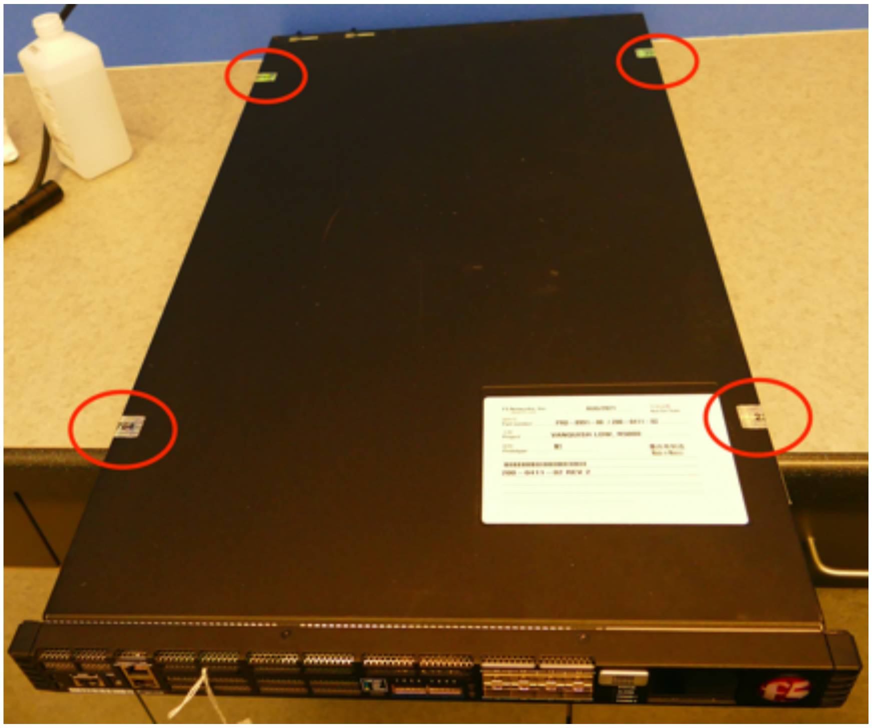

Figure 8 – Tamper labels on r5900 (4 of 4 tamper labels)

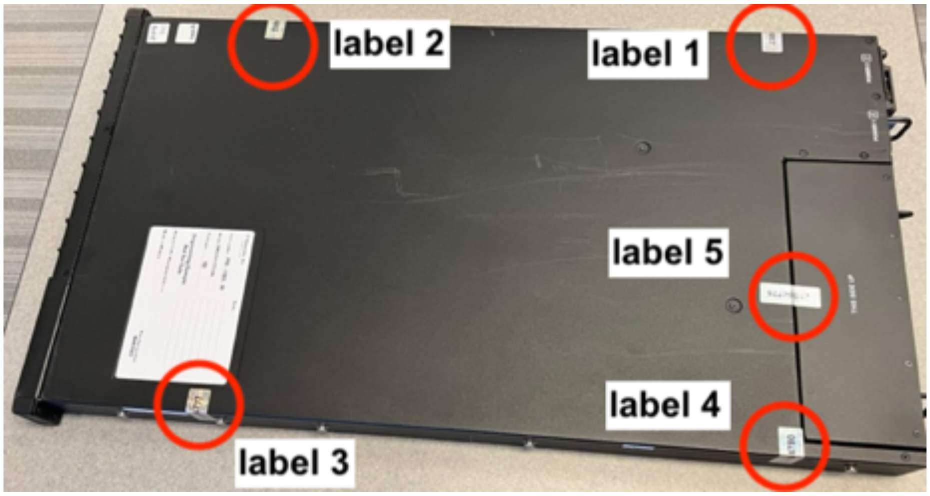

Figure 9 - Tamper labels on r5920-DF (5 of 5 tamper labels).
Labels are located on the lateral sides of the platform -labels 1,2,3 and 4. The tamper label 5 on the chassis / enclosure lid is covering the ventilation fan tray that allows access to SSD.

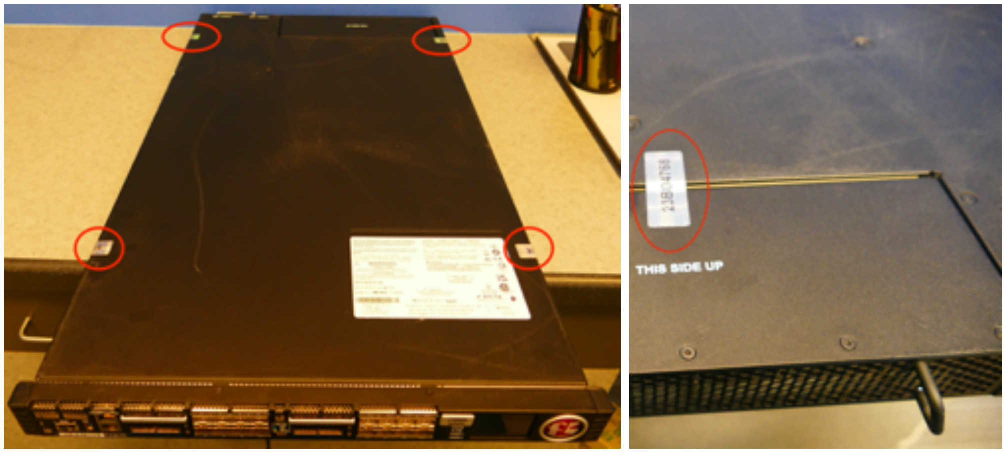

*Figure 10 – Tamper labels on r10900, r10920-DF and r12900-DS
(4 +1 of 5 tamper labels shown). Labels are located on the lateral sides of the platform -labels 1,2,3 and 4. The tamper label 5 on the chassis / enclosure lid is covering the ventilation fan tray that allows access to SSD.*

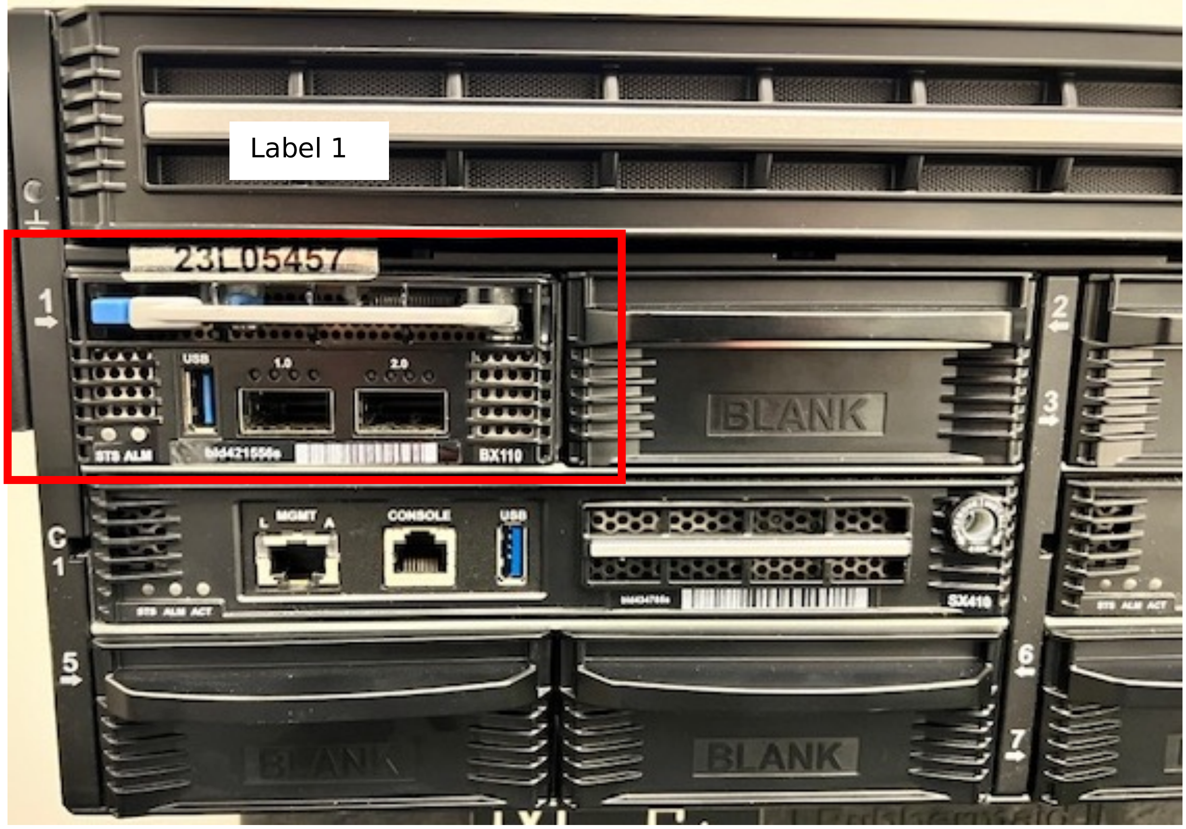

*Figure 11 – Tamper label on VELOS CX410 BX110 blade mounted on chassis.
The tamper label (1 of 1 tamper label shown) marked as "Label 1" affixed between blade and chassis is positioned to provide evidence if the tested blade in slot #1 is removed from the chassis.*

8 Non-Invasive Security

This section is N/A until non-Invasive security is defined in NIST SP800-140F that replaces the ISO/IEC 19790 Annex F requirements.

9 Sensitive Security Parameter Management

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
TLS RSA public key / PSP / asymmetric	112-bits and 150-bits	RSA A3729	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] Key generation method; random values are obtained using [SP 800-90Ar1] DRBG	Import: During protocol handshake Export: During protocol handshake	N/A	SSD	Zeroized when ssl key file is deleted with "Secure Erase" service at boot.	**Use**: Digital signature verification used in the TLS protocol **Related SSPs**: TLS RSA private key, DRBG internal states
TLS RSA private key / CSP / asymmetric				No import No export				**Use**: Digital signature generation used in the TLS protocol **Related SSPs**: TLS RSA public key, DRBG internal states
TLS ECDSA public key / PSP / asymmetric	128-bits and 192-bits	ECDSA A3729 A3730	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] ECDSA Key Generation method; random values are obtained using [SP 800-90Ar1] DRBG	Import: During protocol handshake Export: During protocol handshake	N/A	SSD	Zeroized when ssl key file is deleted with "Secure Erase" service at boot.	**Use**: Digital signature verification used in the TLS protocol **Related SSPs**: TLS ECDSA private key, DRBG internal states
TLS ECDSA private key / CSP				No import No export				**Use**: Digital signature generation used in the TLS protocol

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
/ asymmetric								**Related SSPs**: TLS ECDSA public key, DRBG internal states
TLS EC Diffie-Hellman public key / PSP / asymmetric	128-bits and 192-bits	EC Diffie-Hellman A3729 A3730	Generated conformant to SP800-133r2 (CKG) i.e. key generation method specified in [SP 800-56Ar3] using [FIPS 186-4] Key Generation; random values are obtained using [SP 800-90Ar1] DRBG	Import: During protocol handshake Export: During protocol handshake	N/A	RAM	Zeroized by closing TLS session or by "Reboot System" service	**Use**: TLS protocol key exchange **Related SSPs**: DRBG internal states, TLS pre-primary secret
TLS EC Diffie-Hellman private key / CSP/ asymmetric				No import, No export				
TLS Diffie-Hellman public key / PSP / asymmetric	112, 128, and 150-bits	Diffie-Hellman A3729 A3730	Generated using Safe primes key generation method specified in SP800-56Ar3; random values are obtained using [SP 800-90Ar1] DRBG	Import: During protocol handshake Export: During protocol handshake	N/A	RAM	Zeroized by closing TLS session or by "Reboot System" service	**Use**: Key Generation, TLS protocol key exchange **Related SSPs**: DRBG internal states, TLS pre-primary secret
TLS Diffie-Hellman private key / CSP				No import, No export				

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
/ asymmetric								
TLS pre-primary secret	Diffie-Hellman: 112, 128, 150-bits EC Diffie-Hellman: 128-bits and 192-bits	TLS KDF A3729 A3730	N/A	No import No export	SP800-56Ar3 KAS-ECC-SSC and KAS-FFC-SSC	RAM	Zeroized by closing TLS session or by "Reboot System" service	**Use**: TLS protocol **Related SSPs**: EC Diffie-Hellman public and private keys; TLS primary secret
TLS primary secret	256-bits	TLS KDF A3730 A3729	SP 800-135r1 TLS KDF	No import No export	N/A	RAM	Zeroized by closing TLS session or by "Reboot System" service	**Use**: TLS protocol **Related SSPs**: TLS pre-primary secret; TLS derived key
TLS derived session key (AES HMAC)	128 and 256-bits (AES) 112 and 256-bits (HMAC)	AES HMAC A3730 A3729	SP 800-135r1 TLS KDF	No import No export	N/A	RAM	Zeroized by closing TLS session or by "Reboot System" service.	**Use**: TLS protocol **Related SSPs**: TLS pre-primary secret, TLS primary secret
SSH ECDSA public key / PSP / asymmetric	128 and 192-bits	ECDSA A3729	Generated conformant to SP800-133r2 (CKG) i.e. key generation method	Import: During SSH session using the "Configure SSH user configura	N.A	SSD	Zeroized using SSH keyswap service or Secure Erase"	**Use**: SSH key-based authentication **Related SSPs**: DRBG internal states

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
			specified in [SP 800-56Ar3] using [FIPS 186-4] ECDSA Key generation method; random values are obtained using [SP 800-90Ar1] DRBG	tion" service. Export: During SSH session			service at boot.	
SSH ECDSA private key / CSP / asymmetric				No import No export				
SSH EC Diffie-Hellman public key / PSP / asymmetric	128 and 192-bits	EC Diffie-Hellman Shared Secret Computation A3729	Generated conformant to SP800-133r2 (CKG) using [FIPS 186-4] Key generation method; random values are obtained using [SP 800-90Ar1] DRBG	Import: During protocol handshake Export: During protocol handshake	N/A	RAM	Zeroized by closing SSH session or terminating the SSH application or "Reboot System" service	**Use**: SSH handshake **Related SSPs**: SSH shared secret, DRBG internal states
SSH EC Diffie-Hellman private key / CSP / asymmetric				No import No export				
SSH shared secret	128 and 192-bits	SSH KDF A3729	N/A	No import No export	SP800-56Ar3 KAS-ECC-SSC	RAM	Zeroized by closing SSH session or terminating the	**Use**: Key derivation; SSH shared secret; **Related SSPs**: EC Diffie-Hellman

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
							SSH application or "Reboot System" service	public and private keys; SSH derived session key
SSH derived session key (AES, HMAC)	128 and 256-bits (AES) 112 and 256-bits (HMAC)	AES HMAC A3729	SP 800-135r1 SSH KDF	No import No export	N/A	RAM	Zeroized by closing SSH session or terminating the SSH application or "Reboot System" service	**Use**: data encryption / decryption and MAC calculations in SSH protocol **Related SSPs**: SSH shared secret
Password	1/676,000,000 (see Table 7)	N/A	N/A	Input by the User or CO invoking "create additional user" or "Update own password" or "Update others password" services No export	N/A	SSD as a hashed value	Zeroized by "Secure Erase" service at boot	**Use**: SSH authentication, WebUI login **Related SSPs**: N/A
Entropy input / CSP (IG D.L)	256 bits	Entropy Source ESV Cert. #E74	Obtained from non-physical Entropy source	No import No export	N/A	RAM	Zeroized by "Reboot System" service	**Use**: random number generation **Related SSPs**: DRBG seed
DRBG seed / CSP	256 bits	CTR_DRBG A3729 A3730	Derived from the entropy string as	No import No export	N/A	RAM	Zeroized by "Reboot	**Use**: random number generation

Key/ SSP Name/ Type	Strength	Security Function / Cert. Number	Generation	Import /Export	Establishment	Storage	Zeroization	Use and related SSPs
(IG D.L)			defined by [SP 800-90Ar1]				System" service	**Related SSPs**: Entropy input, DRBG Internal states
DRBG internal states (V and key values) / CSP (IG D.L)	256 bits	CTR_DRBG A3729 A3730	Derived from the seed as defined by [SP 800-90Ar1]	No import No export	N/A	RAM	Zeroized by "Reboot System" service	**Use**: random number generation **Related SSPs**: Entropy input, DRBG seed

Table 12 - SSPs

9.1 Random Bit Generation - Entropy Source

The module employs a Deterministic Random Bit Generator (DRBG) based on [SP800-90Ar1] for the generation of random value used in asymmetric keys. The Approved DRBG provided by the module is the CTR_DRBG with AES-256. The module uses the SP800-90B compliant Entropy source specified in Table 13 to seed the DRBG.

The operator does not have the ability to modify the F5 entropy source (ES) configuration settings (see details in Public Use Document referenced in section 11.2). The F5 ES is tested in the OEs listed in Table 1.

Entropy Source	Minimum number of bits of entropy	Details
ESV #E74 (non-physical noise source)	256-bits	The CPU Jitter RNG version 3.4.0 entropy source uses jitter variations caused by executing instructions and memory accessed. The entropy source has been shown to provide full 256-bits of entropy at the output of the SHA3-256 vetted conditioning function (#A2621).

Table 13 - Non-Deterministic Random Number Generation Specification

9.2 SSP Generation

For generating RSA and ECDSA keys, the modules implements asymmetric key generation services compliant with [FIPS186-4]. A seed (i.e. the random value) used in asymmetric key generation is directly obtained from the SP800-90Ar1] DRBG.

The Diffie-Hellman generates keys using safe primes compliant with [SP800-56Arev3].

The public and private keys used in the EC Diffie-Hellman key agreement schemes are generated internally by the module using the ECDSA key generation method compliant with [FIPS186-4] and [SP800-56Arev3]

In accordance with FIPS 140-3 IG D.H, the cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per section 4 example 1 [SP800-133r2] (vendor affirmed).

The module does not implement symmetric key generation as an explicit service. The HMAC and AES symmetric keys are derived from shared secrets by applying [SP 800-135r1] as part of the TLS/ SSH protocols. The scenario maps to the [SP 800-133r2] section 6.2.1 *Symmetric keys generated using Key Agreement Scheme*.

9.3 SSP Establishment

The module provides the following key establishment services:

- EC Diffie-Hellman key agreement scheme compliant with SP800-56Ar3 and FIPS 140-3 IG D.F scenario 2 (path 2) is used as part of the TLS and SSH Protocols. The full EC Diffie-Hellman KAS implements a shared secret computation with the key derivation implemented by [SP 800-135r1] TLS KDFs and [SP 800-135r1] SSH KDFs.

  EC Diffie-Hellman key agreement provides 128 or 192-bits of encryption strength.

- Diffie-Hellman key agreement scheme compliant with SP800-56Ar3 and FIPS 140-3 IG D.F scenario 2 (path 2) is used as part of the TLS Protocols. The full Diffie-Hellman KAS implements a shared secret computation with the key derivation implemented by [SP 800-135r1] TLS KDFs.

  Diffie-Hellman key agreement provides between 112 and 150-bits of encryption strength.

- [SP 800-38F], IG D.G key wrapping in the context of TLS protocols where a key may be within a packet or message that is encrypted and authenticated using approved authenticated encryption mode or a combination method which includes approved symmetric encryption algorithm together with approved authentication method.

  [SP 800-38F] key wrapping using approved authenticated encryption mode (i.e. AES-GCM, AES-CCM) provides 128 or 256 bits of encryption strength (AES Certs. #A3729 and # A3730).

  [SP 800-38F] key wrapping using a combination of approved AES encryption and HMAC authentication method provides 128 or 256 bits of encryption strength (AES and HMAC Certs. #A3729 and # A3730).

- [SP 800-38F], IG D.G key wrapping in the context of SSH protocols where a key may be within a packet or message that is encrypted and authenticated using a combination method which includes approved symmetric encryption algorithm together with approved authentication method.

  [SP 800-38F] key wrapping using a combination of approved AES-CBC or AES-CTR encryption mode and HMAC authentication method provides 128 or 256 bits of encryption strength (AES and HMAC Cert. # A3729).

9.4 SSP Entry / Output

For TLS with EC Diffie-Hellman / Diffie-Hellman key exchange, the TLS pre-primary secret is established during key agreement and is not output from the module. Once the TLS session is established, any key or data transfer performed thereafter is protected by authenticated encryption mode using AES-GCM/ AES-CCM or by AES encryption and HMAC authentication through a mutually agreed AES and HMAC session keys derived by applying SP 800-135r1 TLS KDF.

For SSH with EC Diffie-Hellman key exchange, the SSH shared secret is established during key agreement and is not output from the module. SSH ECDSA public keys can be imported into the module by the CO and User role using the "Configure SSH user configuration" service. Once the SSH session is established, any key or data transfer performed thereafter is protected by AES encryption and HMAC authentication through a mutually agreed AES and HMAC session keys derived by applying SP 800-135r1 SSH KDF.

There are no encrypted SSPs that are directly entered.

9.5 SSP Storage

As shown in Table 12 the keys are stored in the volatile memory (RAM) in plaintext form.

The static SSPs are persistently stored in plaintext in the SSD that is part of the OE. The static SSPs will remain on the system across power cycle.

SSPs are only accessible to the authenticated operator, to which the SSPs are associated.

9.6 SSP Zeroization

The zeroization methods listed in Table 12, overwrites the memory occupied by keys with “zeros” or pre-defined values.

The zeroization of temporary values are performed when no longer needed.

The zeroization can be enforced by the Crypto Officer and Resource Manager role with the following services:

The zeroization can be enforced by the Crypto Officer with the following services:

- Calling Reboot System service will clear the SSPs present in volatile memory RAM memory.
- Using Secure Erase service (which can only be triggered by the crypto officer during reboot of the platform) will perform a single pass zeroization erasing the SSPs present in persistent memory.

10 Self-tests

10.1 Pre-Operational Self-Tests

The pre-operational self-tests are performed automatically whenever the module is powered on. At initialization the module performed pre-operational self-test (integrity test) and the conditional cryptographic algorithm tests (CASTs). The data output interface is inhibited and services are not available during the pre-operational self-tests and CASTs. On successful completion of the pre-operational and CASTs, the module enters operational mode and cryptographic services are available. If the module fails any of the tests, it will return an error code and enter into an error state.

10.1.1 Pre-operational Software/Firmware Integrity Test

The integrity of the module is verified by comparing the HMAC-SHA-384 checksum values of the installed binaries calculated at run time with the stored values computed at build time. If the values do not match the system enters the error state and the module will not be accessible. In order to recover from this state, the module needs to be reinstalled. The HMAC-SHA384 algorithm is self-tested prior to the integrity test being run.

10.2 Conditional Self-Tests

The conditional tests are performed without operator intervention, without any external controls, externally provided test vectors, output results and the determination of pass of fail is done by the module.

If one of the conditional self-tests fails, the module transitions to the error state and a corresponding error indication is given. The module becomes inoperable, and no services are available. Data output and cryptographic operations are inhibited while the module is in the error state.

10.2.1 Conditional Cryptographic Algorithm Self-Tests

The module performs cryptographic algorithm self-tests (CASTs) on all Approved cryptographic algorithms.

Algorithm	Test
Control Plane	
non-physical entropy source	SP800-90B health test (APT and RCT) classified as CAST: • at start-up: performed on 1,024 consecutive samples. • during runtime.
CTR_DRBG	CAST KAT with AES 256 bits with and without derivation function (SP800-90Ar1 section 11.3 health tests)
AES	CAST KAT of AES encryption / decryption separately with AES-GCM mode and 256-bit key CAST KAT of AES encryption / decryption separately with ECB mode and 128 bit-key
RSA	CAST KAT of RSA PKCS#1 v1.5 signature generation with 2048 bit key and SHA2-256

Algorithm	Test
	CAST KAT of RSA PKCS#1 v1.5 signature verification with 2048 bit key and SHA2-256
ECDSA	CAST KAT of ECDSA signature generation using P-256 and SHA2-256 CAST KAT of ECDSA signature verification using P-256 and SHA2-256
KAS-ECC-SSC	CAST KAT of shared secret computation with P-256 curve
KAS-FFC-SSC	CAST KAT of shared secret computation with 2048 modulus
HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512	CAST KAT of HMAC-SHA-1, CAST KAT of HMAC-SHA2-256 CAST KAT of HMAC-SHA2-384 (prior integrity test) CAST KAT of HMAC-SHA2-512
SHA-1, SHA2-256, SHA2-384, SHA2-512	CAST KATs for all SHA sizes are covered by the respective HMAC KATs (allowed per IG 10.3.B)
[SP800-135r1] KDF	SSH CAST KAT TLS1.2 CAST KAT
Data Plane	
AES	CAST KAT of AES encryption with GCM mode and 128-bit key CAST KAT of AES encryption /decryption performed separately with CBC mode and 128-bit key
RSA	CAST KAT of RSA PKCS#1 v1.5 signature generation with 2048 bit key and SHA2-256 CAST KAT of RSA PKCS#1 v1.5 signature verification with 2048 bit key and SHA2-256
ECDSA	CAST KATs of ECDSA signature generation and verification with P-256 curve, SHA2-256
KAS-ECC-SSC	CAST KAT of shared secret computation with P-256 curve
KAS-FFC-SSC	CAST KAT of shared secret computation with 2048 modulus
CTR_DRBG	Covered by Control Plane Self-Tests. (Data Plane makes use of the same DRBG implementation provided by Control Plane)
[SP800-135r1] KDF	TLS1.2 CAST KAT
HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512	CAST KAT of HMAC-SHA-1 CAST KAT of HMAC-SHA2-256 HMAC-SHA2-384 CAST KAT is covered by IG 10.3.A resolution 4. CAST KAT of HMAC-SHA2-512

Algorithm	Test
SHA-1, SHA2-256, SHA2-384, SHA2-512	CAST KATs for all SHA sizes are covered by respective HMAC KATs (allowed per IG 10.3.B)

Table 14 – Conditional Cryptographic Algorithm Self-Tests

10.2.2 Conditional Pairwise Consistency Self-Tests

A pairwise consistency test is run whenever asymmetric keys (RSA for Control Plane only, Diffie-Hellman, EC Diffie-Hellman, or ECDSA for both planes) are generated. PCT for ECDSA and RSA Key Pair Generation used for digital signatures is tested by the calculation and verification of a digital signature. PCT for Diffie-Hellman Key Pair Generation is performed following the SP 800-56Ar3 section 5.6.1 requirements. PCT for EC Diffie-Hellman Key Pair Generation in the Control Plane is covered by ECDSA PCT (IG 10.3.A). PCT for EC Diffie-Hellman Key Pair Generation used for key agreement in Data Plane is performed following the SP 800-56Ar3 section 5.6.2.1.4 requirements.

10.2.3 On Demand Self-Tests

On demand self-tests are performed by powering off the module and powering it on again. This service performs pre-operational self-tests and CASTs. During the execution of the on demand self-tests, crypto services are not available and no output through data output or cryptographic operations are possible.

10.3 Error States

Error State	Cause of Error	Status Indicator
error state	HMAC-SHA2-384 integrity test failure	Module will not load
	Failure of any of the Control Plane CAST KATs, and Data Plane CAST KATs	Module will not load
	Failure of any of the PCTs	Module will reboot
	Failure of the APT, RCT at runtime	Module will reboot (RCT, APT)
	Failure of the APT, RCT at restart	Module will not load

Table 15 - Error States

In any of the error states, any data output or cryptographic operations are prohibited. The module must reboot or re-load with a fresh image to clear the error condition.

All data output and cryptographic operations are inhibited when the module is in an error state.

11 Life-Cycle Assurance

11.1 Startup Procedures

The module is distributed as a part of a BIG-IP product which includes the hardware platform and an installed copy of firmware with a platform layer F5OS and the BIG-IP version 17.1.0.1. The hardware platforms are shipped directly from the hardware manufacturer/authorized subcontractor via trusted carrier and tracked by that carrier. The hardware is shipped in a sealed box that includes a packing slip with a list of components inside, and with labels outside printed with the product nomenclature, sales order number, and product serial number. Upon receipt of the hardware, the customer is required to perform the following verifications:

- Ensure that the shipping label exactly identifies the correct customer's name and address as well as the hardware model.
- Inspect the packaging for tampering or other issues.
- Ensure that the external labels match the expected delivery and the shipped product.
- Ensure that the components in the box match those on the documentation shipped with the product.
- Verify the hardware model with the model number given on the shipping label and marked on the hardware platform itself.

11.2 Administrator Guidance

The Crypto Officer should verify that the following specific configuration rules are followed to operate the module in the approved mode validated configuration.

The ESV Public Use Document (PUD) reference for non-physical entropy source is as follows: https://csrc.nist.gov/projects/cryptographic-module-validation-program/entropy-validations/certificate/74

11.2.1 Installing Tamper Evident Labels

Before the module is installed in the production environment, tamper-evident labels must be installed in the location identified for each test platforms in Section 7.2. The following steps should be taken when installing or replacing the tamper evident labels on the test platforms on which the module runs. The instructions are also included in *F5 Platforms: FIPS Kit Installation* provided with each hardware platforms.

- Use the provided alcohol wipes to clean the chassis cover and components of dirt, grease, or oil before you apply the tamper evidence seals.
- After applying the seal, run your finger over the seal multiple times using extra high pressure.
- The seals completely cure within 48 hours.

11.2.2 Installing F5OS

Follow the instructions in the " *Initial Configuration*" guide for the initial setup and configuration of the module.

- Run the Setup wizard "appliance-setup-wizard" using the CLI with the CO account and default credentials. The system will prompt you to change the password.
- LIcense the system from the WebUI. Guidance on Licensing the F5OS system can be found in https://techdocs.f5.com/en-us/hardware/f5-rseries-systems-getting-started/gs-system-initial-config.html#run-setup-wizard) and summarized as followed: Before you can activate

the license for the F5OS system, you must obtain a base registration key. The base registration key is pre-installed on new F5OS systems. When you power up the product and connect through the webUI, you can open the SYSTEM SETTINGS > Licensing page to display the registration key. Select "Automatic" for the license Activation Method to communicate with the F5 License Server. The F5 product generates a dossier which is an encrypted list of key characteristics used to identify the platform and activates the license.

11.2.3 Tenant image installation and deployment

The tenant inherits the license and VLANs of the rSeries VELOS host. The crypto officer must follow the following instructions to create a tenant from the Web-based management interface:

- Login with the CO account to the Tenant WebUI and select SYSTEM SETTINGS > Software Management to add image by uploading the qcow2 bundle file of the BIG-IP version 17.1.0.1.
- Under TENANT MANAGEMENT' > 'Tenant Deployments' and click 'Add'. Fill out the form and provide tenant Name, tenant Type (ie BIG-IP) and tenant Image (17.1.0.1.) and other information to fully deploy the tenant.
- Login to the new tenant via ssh or WebUI and configure as you would any BIG-IP system.

Tenant creation from CLI is detailed in the publicly available f5,com page (https://techdocs.f5.com/en-us/f5os-a-1-0-0/f5-rseries-systems-installation-upgrade/title-install-upgrade-software.html).

- Once the module is installed, licensed and configured, the Crypto Officer should confirm that the system is installed and licensed correctly.

11.2.3.1 Version Confirmation

The Crypto Officer should call the show version service (with commands "tmsh show sys version" and "tmsh show sys license"), then confirm that the provided version matches the validated version shown in Table 2. Any firmware loaded into the module other than version 17.1.0.1 is out of the scope of this validation and will mean that the module is not operating as a FIPS validated module.

11.2.3.2 License Confirmation

The FIPS validated module activation requires installation of the license referred as ‘FIPS license’. The Crypto Officer should call the show license service (with command "tmsh show sys license"), then verify that the list of license flags includes "FIPS 140-3”.

11.2.4 Additional Guidance

The Crypto Officer should verify that the following specific configuration rules are followed in order to operate the module in the FIPS validated configuration.

- All command shells other than tmsh are not allowed. For example, bash and other user-serviceable shells are excluded.
- Management of the module via the platform's LCD display is not allowed.
- Usage of f5-rest-node and iAppLX and provisioning of iRulesLX is not allowed.
- Only the provisioning of AFM and LTM is included.
- Remote access to the Lights Out / Always On Management capabilities of the system are not allowed.
- Serial port console and USB port should be disabled after the initial power on and communications setup of the module.

- Use of command run util fips-util -f init is not allowed. Running this command followed by a System Reboot service or restart will mean that the module is not operating as a FIPS validated module.
- The Single Diffie-Hellman use option should be turned ON for the platform GUI.

11.3 Non-Administrator Guidance

The approved and non-approved algorithms available to users are listed in section 2, the physical ports, and logical interfaces available to users are specified in section 3. The Approved and non-Approved modes of operation are specified in section 2.3. The algorithm-specific information is listed in sub-section below.

11.3.1 AES GCM IV

AES-GCM IV is constructed in accordance with SP800-38D in compliance with IG C.H scenario 1. e module does not support AES-GCM with external IV. The implementation of the nonce_explicit management logic inside the module ensures that when the IV exhausts the maximum number of possible values for a given session key, the module triggers a new handshake request to establish a new key. In case the module’s power is lost and then restored, the key used for the AES GCM encryption or decryption shall be re-distributed. The AES GCM IV generation follows [RFC 5288] and shall only be used for the TLS protocol version 1.2 to be compliant with [FIPS140-3_IG] IG C.H scenario 1; thus, the module is compliant with [SP800-52r2] section 3.3.1.

11.3.2 RSA SigGen/ SigVer

All the modulus sizes supported by the module have been ACVP tested (per IG C.F).

11.3.3 SP800-56Ar3 Assurances:

To comply with the assurances found in Section 5.6.2 of SP 800-56Ar3, the keys for KAS-FFC-SSC and KAS-ECC-SSC must be generated using the approved key generation services specified in section 2.9. The module performs full public key validation on the generated public keys. Additionally, the module performs full public key validation on the received public keys.

12 Mitigation of Other Attacks

The module does not implement security mechanisms to mitigate other attacks.

Appendix A. Glossary and Abbreviations

ADC	Application Delivery Controller
AES	Advanced Encryption Standard
API	Application Programming Interface
ACVP	Automated Cryptographic Validation Protocol
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CKG	Cryptographic Key Generation
CLI	Command Line Interface
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
ESV	Entropy Source Validation
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
KAS	Key Agreement Schema
KAT	Known Answer Test
KW	AES Key Wrap
KWP	AES Key Wrap with Padding
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
OFB	Output Feedback
PR	Prediction Resistance
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator

RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSD	Solid State Drive
SSH	Secure Shell
SSLO	Secure Sockets Layer (SSL) Orchestrator
TDES	Triple-DES
TLS	Transport Layer Security
XTS	XEX-based Tweaked-codebook mode with cipher text Stealing

Appendix B. References

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

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

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

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

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

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

PKCS#1
Public Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1
March 1998
https://datatracker.ietf.org/doc/html/rfc2313

RFC 5288
AES Galois Counter Mode (GCM) Cipher Suites for TLS
August 2008
https://www.ietf.org/rfc/rfc5288.txt

RFC 7627
Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension
September 2015
https://www.ietf.org/rfc/rfc7627.txt

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

SP800-38B
NIST Special Publication 800-38B - Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication
May 2005
http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf

SP800-38C NIST Special Publication 800-38C - Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality
May 2004
http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38c.pdf

SP800-38D NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC
November 2007
http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

SP800-38F NIST Special Publication 800-38F - Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping
December 2012
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38F.pdf

SP800-38G NIST Special Publication 800-38G - Recommendation for Block Cipher Modes of Operation: Methods for Format - Preserving Encryption
March 2016
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38G.pdf

SP800-52r2 Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations
August 2019
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r2.pdf

SP800-56Ar3 NIST Special Publication 800-56A Revision 3 - Recommendation for Pair Wise Key Establishment Schemes Using Discrete Logarithm Cryptography
April 2018
https://doi.org/10.6028/NIST.SP.800-56Ar3

SP800-90Ar1 NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators
June 2015
https://doi.org/10.6028/NIST.SP.800-90Ar1

SP800-90B NIST Special Publication 800-90B - Recommendation for the Entropy Sources Used for Random Bit Generation
January 2018
https://doi.org/10.6028/NIST.SP.800-90B

SP800-131Ar2 Transitioning the Use of Cryptographic Algorithms and Key Lengths
March 2019
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar2.pdf

SP800-133r2 NIST Special Publication 800-133 Revision 2 - Recommendation for Cryptographic Key Generation
June 2020
https://doi.org/10.6028/NIST.SP.800-133r2

SP800-135r1 NIST Special Publication 800-135 Revision 1 - Recommendation for Existing Application-Specific Key Derivation Functions
December 2011
http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-135r1.pdf

SP800-140B NIST Special Publication 800-140B - CMVP Security Policy Requirements
March 2020
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-140B.pdf