F5, Inc.

F5OS-C Cryptographic Module

FIPS 140-3 Non-Proprietary Security Policy

Document Version 1.0
2025-09-23

Prepared by:

Lightship Security

Applus

www.lightshipsec.com

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This non-proprietary FIPS 140-3 Security Policy for the *F5OS Cryptographic Module Version C-1.8.1 VELOS Chassis*, version 1.8.1 describes how the module meets the security requirements specified in FIPS 140-3 for an overall security level 2 module and outlines the security rules and operating procedures required to maintain compliance.

1.2 Security Levels

Section	Title	Security Level
1	General	2
2	Cryptographic module specification	2
3	Cryptographic module interfaces	2
4	Roles, services, and authentication	3
5	Software/Firmware security	2
6	Operational environment	2
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
	Overall Level	2

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

The *F5OS Cryptographic Module Version C-1.8.1 VELOS Chassis* (hereafter referred to as "the module") is a firmware module operating in a chassis platform which provides platform layer services components for the VELOS system controllers and chassis partitions.

Module Type: Firmware

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary: The cryptographic boundary for the module is defined as the F5OS-C image, which contains the operating system, all calling applications, and implements all approved services.

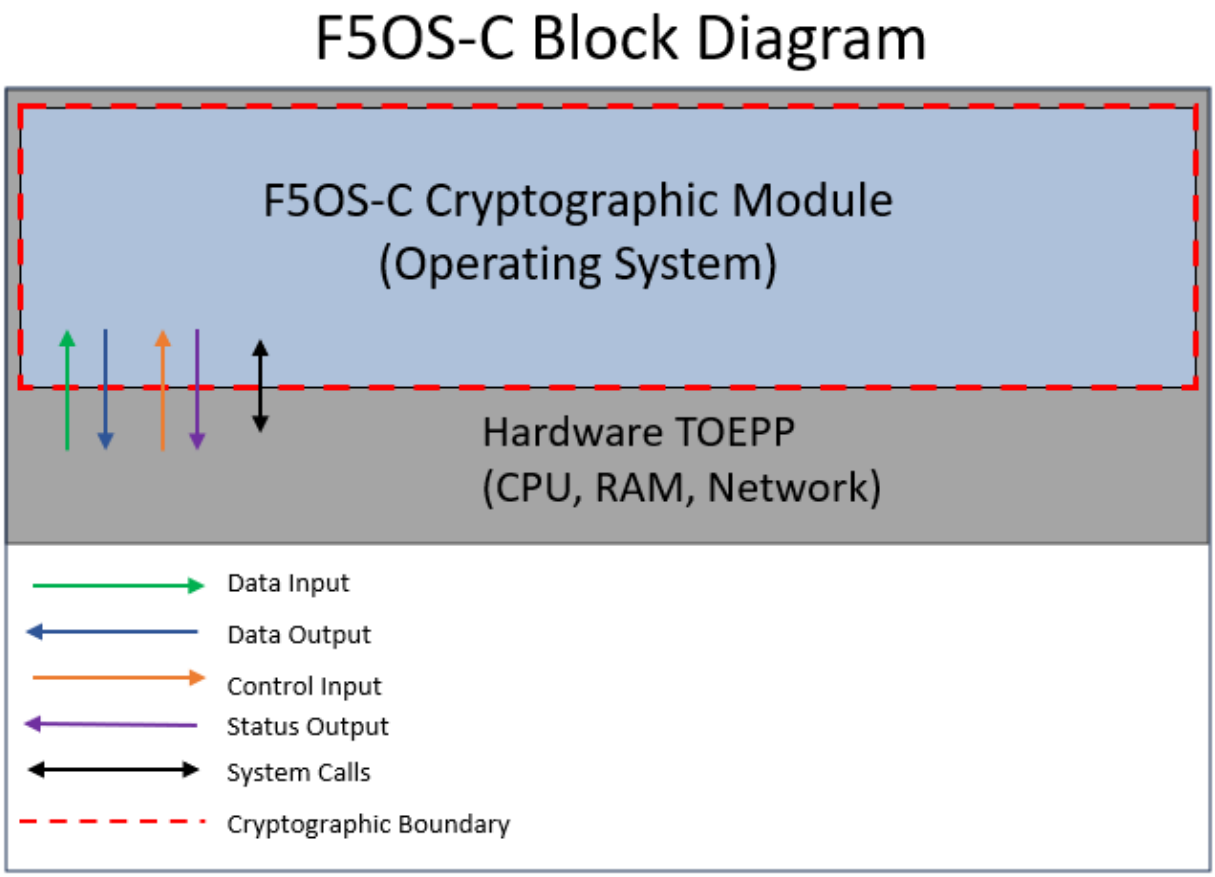

Figure 1: The F5OS-C Cryptographic Module block diagram

The Tested Operational Environment's Physical Perimeter (TOEPP) for the module is defined as the module outer metal chassis of the F5 CX1610 and CX410 models and all components within the chassis, excluding the traffic blades. The CX1610 and CX410 Chassis models are pictured below.

Figure 2: The F5 CX1610 Chassis, fully populated with 16 *BX520* traffic blades

Figure 3: The F5 CX410 Chassis, fully populated with 8 *BX110* traffic blades

2.2 Tested and Vendor Affirmed Module Version and Identification

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
F5OS-C	1.8.1	N/A	HMAC-SHA2-384

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

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
F5OS-C 1.8.1	CX1610 (chassis) with SX1610 (system controller)	Intel Denverton-NS C3958	Yes	N/A	1.8.1
F5OS-C 1.8.1	CX410 (chassis) with SX410 (system controller)	Intel Denverton-NS C3758	Yes	N/A	1.8.1

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

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Excluded Components

The module does not exclude any components within the cryptographic boundary.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	The approved mode of operation	Approved	Global ("FIPS Module: F5OS-C Cryptographic Module")

Table 4: Modes List and Description

The module only supports approved mode of operation. The operator can confirm that the module is operating in the approved mode by invoking the following commands:

“Show system security fips-module” (which will output the following:)

“System security fips-module state name “FIPS Module: F5OS-C Cryptographic Module”

“show system licensing” (which will output the following:)

“FIPS 140 Compliant Mode, <CX1610 / CX410>”

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A4783	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A4783	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - Yes Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
AES-GCM	A4783	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256 Tag Length - 104, 112, 120, 128, 32, 64, 96 IV Length - IV Length: 96 Payload Length - Payload Length: 128, 256, 104, 408 AAD Length - AAD Length: 128, 384, 160, 720, 0	SP 800-38D
Counter DRBG	A4783	Prediction Resistance - No Supports Reseed - Yes Mode - AES-256 Derivation Function Enabled - Yes Additional Input - Additional Input: 0 Entropy Input - Entropy Input: 256 Nonce - Nonce: 128 Personalization String Length - Personalization String Length: 0-256 Increment 256 Returned Bits - 512	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-5)	A4782	Curve - P-256, P-384 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA SigGen (FIPS186-5)	A4782	Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-5)	A4782	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA-1	A4783	MAC - MAC: 160 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1
HMAC-SHA2-256	A4783	MAC - MAC: 256 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1
HMAC-SHA2-384	A4783	MAC - MAC: 384 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A4782	Domain Parameter Generation Methods - P-256, P-384 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF SSH (CVL)	A4782	Cipher - AES-128, AES-256 Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-5)	A4782	Key Generation Mode - probable Modulo - 2048, 3072, 4096 p mod 8 - 0 Primality Tests - 2powSecStr q mod 8 - 0 Fixed Public Exponent - 010001 Info Generated By Server - No Private Key Format - standard Public Exponent Mode - fixed	FIPS 186-5
RSA SigGen (FIPS186-5)	A4782	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5	FIPS 186-5
RSA SigVer (FIPS186-5)	A4782	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5 Fixed Public Exponent - 010001 Public Exponent Mode - fixed	FIPS 186-5
SHA-1	A4783	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A4783	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A4783	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A4782	Hash Algorithm - SHA2-256, SHA2-384 Key Block Length - Key Block Length: 1024	SP 800-135 Rev. 1

Table 5: Approved Algorithms -

Entropy Source Conditioning Component

Algorithm	CAVP Cert	Properties	Reference
SHA3-256	A4093	Message Length - Message Length: 0-65536 Increment 8	FIPS 202

Table 6: Approved Algorithms - Entropy Source Conditioning Component

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	NIST SP800-133rev2 - Section 4, example 1

Table 7: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

The module does not implement any non-approved algorithms, allowed in the approved mode of operation.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

The module does not implement any non-approved algorithms, with no security claimed.

Non-Approved, Not Allowed Algorithms:

N/A for this module.

The module does not implement any non-approved, not allowed algorithms.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
RSA KeyGen	AsymKeyPair-KeyGen CKG	Generation of RSA public and private key		RSA KeyGen (FIPS186-5): (A4782) Counter DRBG: (A4783) CKG: () Key Type: Asymmetric
RSA SigGen	DigSig-SigGen	Generation of an RSA signature		RSA SigGen (FIPS186-5): (A4782)
RSA SigVer	DigSig-SigVer	Verification of an RSA signature		RSA SigVer (FIPS186-5): (A4782)
ECDSA KeyGen	AsymKeyPair-KeyGen CKG	Generation of ECDSA public and private key		ECDSA KeyGen (FIPS186-5): (A4782) Counter DRBG:

Name	Type	Description	Properties	Algorithms
				(A4783) CKG: () Key Type: Asymmetric
ECDSA SigGen	DigSig-SigGen	Generation of an ECDSA signature		ECDSA SigGen (FIPS186-5): (A4782)
ECDSA SigVer	DigSig-SigVer	Verification of an ECDSA signature		ECDSA SigVer (FIPS186-5): (A4782)
KAS-TLS	KAS-Full	Computation of a shared secret using EC Diffie-Hellman and derivation of TLS session keys	IG:D.F Scenario 2, path (2), split Key confirmation:no Key derivation:IG 2.4.B SP 800-135rev1 (CVL) Caveat:Key establishment methodology provides between 128 and 192 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A4782) Scheme: ephemeralUnified TLS v1.2 KDF RFC7627: (A4782)
KAS-SSH	KAS-Full	Computation of a shared secret using EC Diffie-Hellman and derivation of SSH session keys	IG:D.F Scenario 2, path (2), split Key confirmation:no Key derivation:IG 2.4.B SP 800-135rev1 (CVL) Caveat:Key establishment methodology provides between 128 and 192 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A4782) Scheme: ephemeralUnified KDF SSH: (A4782)
AES-CBC	BC-UnAuth	AES encryption and decryption		AES-CBC: (A4783) Key Length: 128, 192
AES-CTR	BC-UnAuth	AES encryption and decryption		AES-CTR: (A4783) Key Length: 128, 192

Name	Type	Description	Properties	Algorithms
AES-GCM	BC-Auth	Authenticated AES encryption and decryption, deterministic IV construction per NIST 800-38D 8.2.1		AES-GCM: (A4783) Key Length: 128, 192
HMAC	MAC	Message authentication code generation		HMAC-SHA-1: (A4783) HMAC-SHA2-256: (A4783) HMAC-SHA2-384: (A4783) SHA-1: (A4783) SHA2-256: (A4783) SHA2-384: (A4783)
DRBG Seed	ENT-Cond ENT-ESV	Generate DRBG Seed		SHA3-256: (A4093)

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

AES-GCM – IG C.H, Scenario 1:

TLS v1.2:

The Module is compliant with **TLS v1.2** and SP800-52 Rev2, Section 3.3.1. The Module supports TLS 1.2 GCM Cipher Suites for TLS, as described in RFC5288 and shall only be used for the TLS protocol version 1.2 to be compliant with FIPS 140-3 IG C.H, Scenario 1a.

No more than 2^{64}–1 AES-GCM encryptions may be performed in the same session and if the invocation counter reaches its maximum value 2^{64}–1, the next AES-GCM encryption is performed with the invocation counter set to 0.

When a session is terminated for any reason, the keys and IVs are zeroised and cannot be used again.

In case the module's power is lost and then restored, new keys and IVs for use with the AES-GCM are established.

SSHv2:

This module is compliant with RFCs 4252, 4253 and the rules for using AES-GCM documented in RFC 5647. The IV is only used in the context of the AES-GCM mode encryptions within the SSHv2 protocol.

The fixed field is 4-bytes in length and is derived form the SSH-KDF, ensuring a unique fixed field for each SSH session.

The invocation field is 8-bytes in length and is incremented each time the AES-GCM function is invoked. No more than 2^{64}–1 AES-GCM encryptions may be performed in the same session and if the invocation counter reaches its maximum value 2^{64}–1, the next AES-GCM encryption is performed with the invocation counter set to 0.

When a session is terminated for any reason, the keys and IVs are zeroised and cannot be used again.

In case the module's power is lost and then restored, new keys and IVs for use with the AES-GCM are established

2.8 RBG and Entropy

Cert Number	Vendor Name
E85	F5, inc.

Table 9: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
CPU Jitter RNG	Non-Physical	F5OS-C 1.8.1 on CX1610 (Controller) on Intel Denverton-NS C3958, Intel Denverton-NS C3758 on F5OS-C 1.8.1	256	Full entropy	A4093

Table 10: Entropy Sources

The module uses "CPU Jitter RNG, version 3.4.1" to seed the DRBG during the modules' boot process and to periodically reseed the DRBG. The entropy loaded into the approved SP800-90Arev1-compliant CTR_DRBG is 256 bits per call. The entropy source implements a vetted SHA3-256 post-conditioning component, which is applied to all output from the noise source.

2.9 Key Generation

The module implements asymmetric key generation services compliant with FIPS 186-5 to generate RSA and ECDSA keys. Seeds for these services are generated from the unmodified output of the module's approved DRBG.

2.10 Key Establishment

The module implements the following approved key agreement methods per 140-3 IG D.F:

Elliptic Curves

The module establishes EC DH shared secrets compliant to SP 800-56Arev3, using elliptic-curves specified in Appendix D of SP 800-186.

2.11 Industry Protocols

The module implements compliant key derivation functions (KDFs) as part of its implementations of the following protocols:

- TLS 1.2
- SSH

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

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
MGMT	Data Input Data Output Control Input Status Output	TLS/SSH protocol input/output messages, Configuration commands for interface management, API which controls system state (e.g. reset system, power off system), API which provides system status information
Power Interface (AC or DC)	Power	Power
Status LEDs	Status Output	System health information, Non-security relevant
LED Touchscreen	Status Output	Networking information, Non-security relevant
Backplane Connection	Data Input Data Output	Module configuration, Module SSPs
USB	None	None in the approved mode of operation
CONSOLE	None	None in the approved mode of operation

Table 11: Ports and Interfaces

The module contains the logical interfaces described in the table above.

The module does not implement a control output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
Password	Username and password pair	Operator Authentication (Password)	1 in 676,000,000	3 in 676,000,000
ECDSA Key	ECDSA Signature Verification	ECDSA SigVer	1 in 2^128	3 in 676,000,000

Table 12: Authentication Methods

The module implements the following identity-based authentication mechanisms:

Password: The minimum password length is 8 characters, enforced by the module. A password must contain at least 1 lower case letter, 1 upper case letter, and 2 numbers. Therefore, the weakest possible password consists of 1 upper case letter, 1 lower case letter, and 6 numbers.

Given the above, the probability of guessing the correct password in the worst-case scenario would be (1/10)^6*(1/26)*(1/26), or 1 in 676,000,000. If authentication by password fails 3 times, the account is locked out and must be unlocked by an Administrator (CO).

ECDSA key-pair: The smallest curve used by the module for key-based authentication in P-256, giving a security strength of 128 bits. Therefore, the chance of success for an authentication attempt using a randomly generated key is 1 in 2^128, which is smaller than 1 in 1,000,000. If key-based authentication fails, the user is prompted for a password. If authentication by password fails 3 times, the account is locked out and must be unlocked by an Administrator (CO).

User can authenticate to the module using the following interfaces:

CLI: The module offers a CLI connection to users over the SSHv2 protocol, using an ethernet connection.

GUI: The module offers a Web-based user interface to users over the HTTPS (TLS 1.2) protocol, using an ethernet connection.

The module does not maintain authenticated sessions upon power cycling. Power cycling the platform will require users to establish a new connection and reauthenticate to the module using one of the above methods.

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Administrator	Identity	CO	Password ECDSA Key
Resource Admin	Identity	User	Password
Operator	Identity	User	Password
User	Identity	User	Password

Table 13: Roles

The module provides the following roles:

- Administrator: The main administrator role for the cryptographic module and the defined crypto officer role. Has access to all administrative functions and services of the module.

- Resource Admin: An administrative user role that has access to a subset of Administrator services, such as viewing audit logs.
- Operator: A user role that has access to a subset of module services, such as read-only informational services, modifying the user's own password, and establishing connections to the module.
- User: A limited user role that has access to read-only informational services only.

The module supports a configurable concurrent user limit, which applies to all roles except for the Administrator (CO) role. The default number of concurrent user connections handled by the module is 10, but the limit can be modified by the Administrator (CO). Connections from the Administrator role (CO) contributes to the number of concurrent connections, but the Administrator role is not bound by the concurrent user limit. (i.e. The default limit is 10 total users + *n* Administrators)

4.3 Approved Services

The abbreviations of the access rights to SSPs have the following interpretation:

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 = Zeroize: The module zeroizes the SSP.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
List Users	Display list of all user accounts	Implicit, i.e. completion of service	Command	List of user accounts	None	Administrator Resource Admin Operator User
Create Additional User	Create additional user	Implicit, i.e. completion of service	Username, password, role	Confirmation of account creation	None	Administrator - Password: W
Modify Existing Users	Modify Existing Users	Implicit, i.e. completion of service	Username, modification (new username, role, password expiry data/tally count)	Confirmation of account creation	None	Administrator - Password: W
Delete User	Delete existing user	Implicit, i.e. completi	Username	Confirmation of deletion	None	Administrator

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
		on of service				
Unlock User	Remove lock from user who has exceeded login attempts	Implicit, i.e. completion of service	Username	Confirmation of unlock	None	Administrator
Update Own Password	Update own password	Implicit, i.e. completion of service	Username, password	Confirmation of password update	None	Administrator - Password: W Resource Admin - Password: W Operator - Password: W User - Password: W
Update Others Password	Update others password	Implicit, i.e. completion of service	Username, password	Confirmation of password update	None	Administrator - Password: W
Configure Password Policy	Set password policy features	Implicit, i.e. completion of service	New password policy	Confirmation of configuration change	None	Administrator
Create TLS Certificate	Self-signed certificate creation	Implicit, i.e. completion of service	Certificate identification information	Confirmation of certificate creation	None	Administrator - TLS RSA public key: E - TLS RSA private key: E - TLS ECDSA public key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - TLS ECDSA private key: E
Create TLS Key	Used for the SSL certificate key file	Implicit, i.e. completion of service	Key identification information	Confirmation of key creation	RSA KeyGen ECDSA KeyGen DRBG Seed	Administrator - TLS RSA public key: G - TLS RSA private key: G - TLS ECDSA public key: G - TLS ECDSA private key: G - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E
Delete TLS Certificate/ Key	Self-signed certificate/key deletion	Implicit, i.e. completion of service	Certificate/ Key identification information	Confirmation of certificate/ key deletion	None	Administrator - TLS RSA public key: W - TLS RSA private key: W - TLS ECDSA public key: W - TLS ECDSA private key: W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
List Certificate	List device TLS certificate	Implicit, i.e. completion of service	Command	List certificates	None	Administrator - TLS RSA public key: R - TLS ECDSA public key: R Resource Admin - TLS RSA public key: R - TLS ECDSA public key: R Operator - TLS RSA public key: R - TLS ECDSA public key: R User - TLS RSA public key: R - TLS ECDSA public key: R
List Private Keys	List device TLS private key	Implicit, i.e. completion of service	Command	List private keys	None	Administrator - TLS RSA private key: R - TLS ECDSA private key: R Resource Admin - TLS RSA private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						key: R - TLS ECDSA private key: R Operator - TLS RSA private key: R - TLS ECDSA private key: R User - TLS RSA private key: R - TLS ECDSA private key: R
View List of System Audit Logs	Display list of logs/files of configuration changes	Implicit, i.e. completion of service	Command	List of log file names	None	Administrator Resource Admin Operator User
View Contents of System Audit Logs	Display list of logs/files of configuration changes	Implicit, i.e. completion of service	Command	Content of log file	None	Administrator Resource Admin
Configure SSH Access	Enable/Disable SSH access	Implicit, i.e. completion of service	SSH access options	Confirmation of configuration of SSH access options	None	Administrator - SSH ECDSA public key: W
Configure SSH IP Address List	Configure IP address allow list	Implicit, i.e. completion of service	IP address list	Confirmation of configuration of SSH IP address list	None	Administrator Resource Admin

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Configure SSH User Configuration	Update ssh/authorized_keys file for user authentication	Implicit, i.e. completion of service	SSH ECDSA public key	Confirmation of configuration of SSH user configuration	None	Administrator - SSH ECDSA public key: W
Create a Tenant	Create tenant deployment	Implicit, i.e. completion of service	Tenant-console credentials	Confirmation of creation of tenant and tenant-console	None	Administrator Resource Admin
Reboot System	Restart system, Zeroize SSPs stored in RAM	Implicit, i.e. completion of service	Command	Confirmation of system reboot	None	Administrator - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z - SSH EC Diffie-Hellman public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						secret: Z - SSH derived session key: Z - Password: Z - Entropy input string: Z - DRBG seed: Z - DRBG internal state V: Z - DRBG internal state key: Z - AES-GCM IV: Z Resource Admin - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z - SSH EC Diffie-Hellman

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z - SSH derived session key: Z - Password: Z - Entropy input string: Z - DRBG seed: Z - DRBG internal state V: Z - DRBG internal state key: Z - AES-GCM IV: Z
Secure Erase	Full system zeroization	Implicit, i.e. completion of service	Command	Confirmation of zeroization	None	Administrator - TLS RSA public key: Z - TLS RSA private key: Z - TLS ECDSA public key: Z - TLS ECDSA private key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z - SSH ECDSA private key: Z - SSH EC Diffie-Hellman public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z - SSH derived session key: Z - Password: Z - Entropy input string: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- DRBG seed: Z - DRBG internal state V: Z - DRBG internal state key: Z
SSH Session Service	Establish and maintain SSH session	Implicit, i.e. completion of service	User, address, password, algorithm, key sizes, primary secret	Confirmation of SSH session establishment	ECDSA KeyGen ECDSA SigGen ECDSA SigVer KAS-SSH AES-CBC AES-CTR AES-GCM HMAC DRBG Seed	Administrator - SSH ECDSA public key: G,W - SSH ECDSA private key: G,W - SSH EC Diffie-Hellman public key: G,W - SSH EC Diffie-Hellman private key: G - SSH shared secret: G,E - SSH derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- Password: R,W - AES-GCM IV: G,E Resource Admin - SSH ECDSA public key: G,W - SSH ECDSA private key: G,W - SSH EC Diffie-Hellman public key: G,W - SSH EC Diffie-Hellman private key: G - SSH shared secret: G,E - SSH derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - Password: R,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- AES-GCM IV: G,E Operator - SSH ECDSA public key: G,W - SSH ECDSA private key: G,W - SSH EC Diffie-Hellman public key: G,W - SSH EC Diffie-Hellman private key: G,W - SSH shared secret: G,W - SSH derived session key: G - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - Password: R,W - AES-GCM IV: G,E User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH ECDSA public key: G,W - SSH ECDSA private key: G,W - SSH EC Diffie-Hellman public key: G,W - SSH EC Diffie-Hellman private key: G - SSH shared secret: G,E - SSH derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - Password: R,W - AES-GCM IV: G,E
Closing SSH Session	Close SSH session	Implicit, i.e. completion of service	Command	Confirmation of SSH session closure	None	Administrator - SSH EC Diffie-Hellman

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z - SSH derived session key: Z Resource Admin - SSH EC Diffie-Hellman public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z - SSH derived session key: Z Operator - SSH EC Diffie-Hellman public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH derived session key: Z User - SSH EC Diffie-Hellman public key: Z - SSH EC Diffie-Hellman private key: Z - SSH shared secret: Z - SSH derived session key: Z
TLS Session Service	Establish and maintain TLS session	Implicit, i.e. completion of service	Address, algorithms, keys	Confirmation of establishment of TLS session	RSA KeyGen RSA SigGen RSA SigVer ECDSA KeyGen ECDSA SigGen ECDSA SigVer KAS-TLS AES-CBC AES-GCM HMAC DRBG Seed	Administrator - TLS RSA public key: G,W - TLS RSA private key: G,W - TLS ECDSA public key: G,W - TLS ECDSA private key: G,W - TLS EC Diffie-Hellman public key: G,W - TLS EC Diffie-Hellman private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						key: G - TLS pre-primary secret: G,E - TLS primary secret: G,E - TLS derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - AES-GCM IV: G,E Resource Admin - TLS RSA public key: G,W - TLS RSA private key: G,W - TLS ECDSA public key: G,W - TLS ECDSA private key: G,W - TLS EC Diffie-Hellman public key: G,W

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS EC Diffie-Hellman private key: G - TLS pre-primary secret: G,E - TLS primary secret: G,E - TLS derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - AES-GCM IV: G,E Operator - TLS RSA public key: G,W - TLS RSA private key: G,W - TLS ECDSA public key: G,W - TLS ECDSA private key: G,W - TLS EC Diffie-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Hellman public key: G,W - TLS EC Diffie-Hellman private key: G - TLS pre-primary secret: G,E - TLS primary secret: G,E - TLS derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - AES-GCM IV: G,E User - TLS RSA public key: G,W - TLS RSA private key: G,W - TLS ECDSA public key: G,W - TLS ECDSA private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						key: G,W - TLS EC Diffie-Hellman public key: G,W - TLS EC Diffie-Hellman private key: G - TLS pre-primary secret: G,E - TLS primary secret: G,E - TLS derived session key: G,E - DRBG seed: E - DRBG internal state V: W,E - DRBG internal state key: W,E - AES-GCM IV: G,E
Closing TLS Session	Close TLS session	Implicit, i.e. completion of service	Command	Confirmation of TLS session closure	None	Administrator - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z Resource Admin - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z Operator - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS primary secret: Z - TLS derived session key: Z User - TLS EC Diffie-Hellman public key: Z - TLS EC Diffie-Hellman private key: Z - TLS pre-primary secret: Z - TLS primary secret: Z - TLS derived session key: Z
Show Version	Return the module name and version	Implicit, i.e. completion of service	Command	Version information and module name	None	Administrator Resource Admin Operator User
Show License	Return license information	Implicit, i.e. completion of service	Command	FIPS license information	None	Administrator Resource Admin Operator User
Show Status	Return the module status	Implicit, i.e. completion of service	Command	Status of the module	None	Administrator Resource Admin Operator User

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Self-Test	Execute integrity test and CASTs	Implicit, i.e. completion of service	Command	Results of the self-tests	HMAC	Administrator Resource Admin
Show Tenant	Lists tenant information	Implicit, i.e. completion of service	Command	List tenant information	None	Administrator Resource Admin
Firmware Update	Loads a new version of the module firmware	Implicit, i.e. completion of service	Command	Success or Fail message, Automatic reboot	RSA SigVer	Administrator

Table 14: Approved Services

As the module implements only approved services and has a global approved mode indicator, successful completion of any module service acts as the implicit *approved service* indicator.

4.4 Non-Approved Services

The module does not provide any non-approved services.

4.5 External Software/Firmware Loaded

The module supports the updating of the entire firmware image, which has been signed internally by F5. Firmware images are transferred to the module by the crypto officer from the administrative GUI over HTTPS. This GUI firmware loading menu only accepts ISO files in the proper format for the F5 device. On loading, the system validates the authenticity of the introduced ISO using an RSA 8192-bit signature verification. Successful verification will trigger installation of the new firmware image and an eventual reboot of the device.

Any firmware image that is not shown on the module certificate is out of scope of this validation and requires a separate FIPS 140-3 validation.

5 Software/Firmware Security

5.1 Integrity Techniques

The module uses HMAC-SHA2-384 as the approved integrity technique for the verification of firmware. During startup, the module computes the MAC values of the installed module firmware image and compares against the stored MAC value computed at build time. If the values do not match, the module enters the Error state.

If the module TOEPP contains 2 system controllers, each system controller contains an instance of the module firmware, and each instance separately performs the module integrity check on its own firmware. This ensures that no matter which system controller is currently active, the module firmware is checked.

5.2 Initiate on Demand

The on-demand pre-operational self-tests, including the approved integrity check and all module CASTs are performed as part of startup and can be initialized by rebooting the module.

Pairwise consistency tests can be invoked on demand by invoking a service that generates asymmetric keys within the module.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Limited

The module's operational environment is considered limited as additional firmware can be loaded to the device in the form of an F5OS-C build, which replaces the existing build, but not all firmware components within the device.

The module contains the entire operating system (OS) for the host device. Therefore, the module itself has control over all SSPs contained within the cryptographic boundary and all spawned processes. The module's modern OS supports and enforces memory separation between application processes.

7 Physical Security

7.1 Mechanisms and Actions Required

Mechanism	Inspection Frequency	Inspection Guidance
Tamper evident seals	Once per Quarter	Inspect tamper-evident seals for signs of tamper / unauthorized access

Table 15: Mechanisms and Actions Required

7.2 User Placed Tamper Seals – CX410 Chassis

Number: 6

Placement: Shown in figures 4-7

Surface Preparation: Yes

The surfaces should be cleaned with 70% Isoproyl alcohol to remove dirt and oil before applying the seals. Ensure that the surface is clean and dry before applying seals. Allow 72 hours for seals to adhere before operating the module.

Operator Responsible for Securing Unused Seals: Administrator (Crypto Officer)

Part Numbers: F5-UPG-FB-STICKER-1

Figure 4: The front side of the module shows the placement of seals # 1, 2, and 3.

Figure 5: The left side of the module shows the placement of seals # 4 and 5.

Figure 6: The rear side of the module shows the placement of seal #4, securing the fan unit to the chassis.

Figure 7: The right side of the module shows the placement of seal #6.

7.3 User Placed Tamper Seals – CX1610 Chassis

Number: 8

Placement: Shown in figures 8-11

Surface Preparation: Yes

The surfaces should be cleaned with 70% Isoproyl alcohol to remove dirt and oil before applying the seals. Ensure that the surface is clean and dry before applying seals. Allow 72 hours for seals to adhere before operating the module.

Operator Responsible for Securing Unused Seals: Administrator (Crypto Officer)

Part Numbers: F5-UPG-FB-STICKER-1

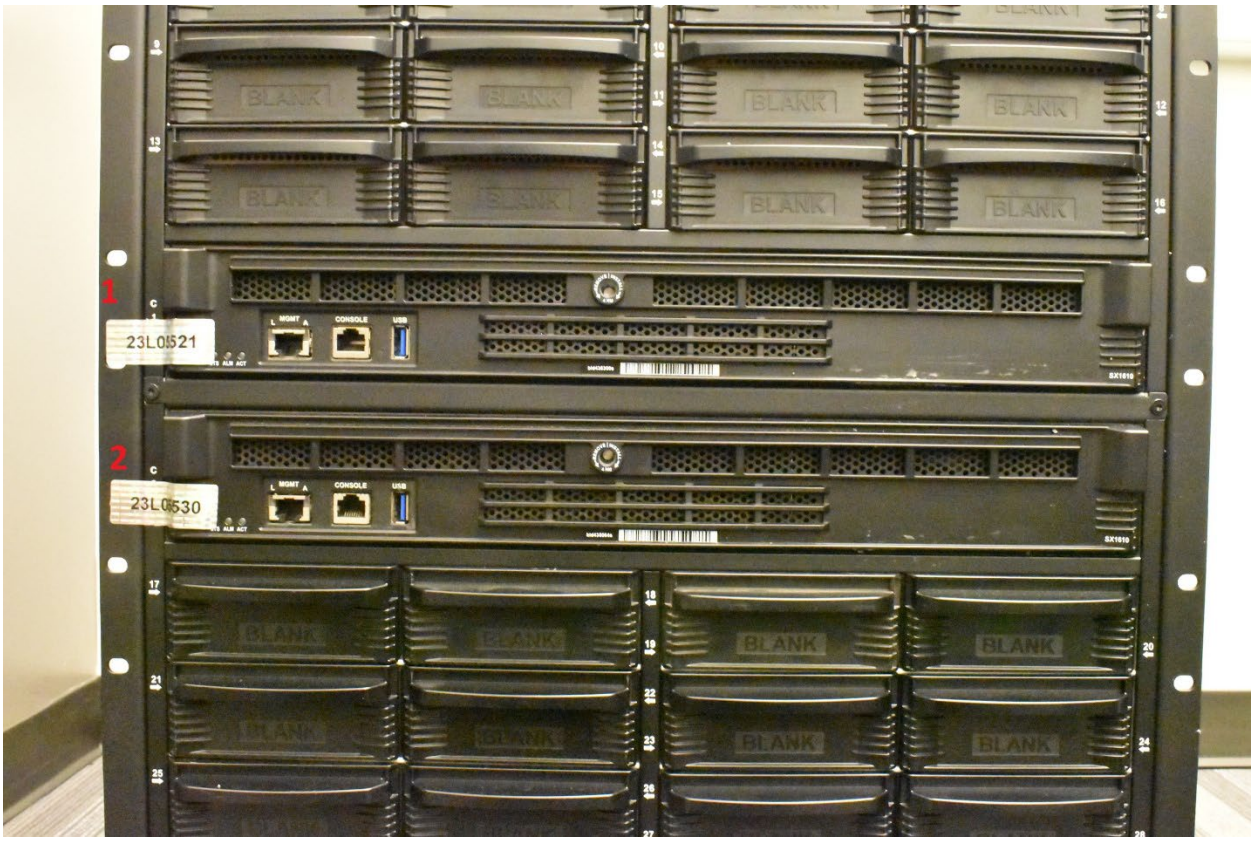

Figure 8: The front side of the module shows the placement of seals #1 and 2, securing the System Controller units to the module.

Figure 9: The left side of the module shows the placement of seal #3.

Figure 10: The right side of the module shows the placement of seals #4, 5, 6, 7, and 8.

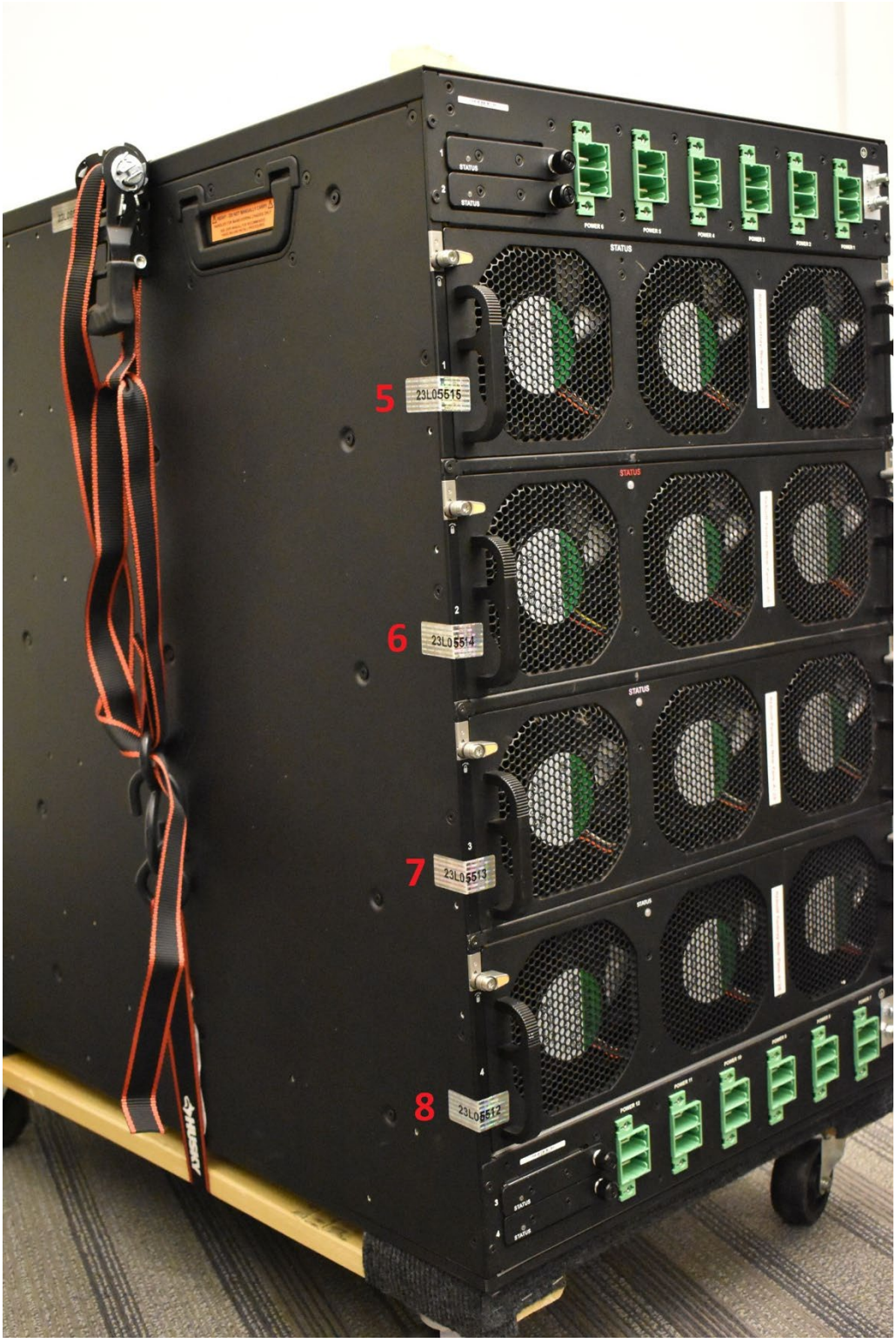

Figure 11: A rear-right angle shot of the module shows the placement of seals #5, 6, 7, and 8.

7.4 Filler Panels

While the module does include blank filler panels for the traffic blades bays, no security relevant components can be seen in either module chassis when the blank filler panels are not present. The module does not rely on the blank filler panels to meet physical security requirements, including opacity.

8 Non-Invasive Security

The Module does not implement any mitigation methods against non-invasive attacks.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	RAM of the module system controller	Dynamic
SSD	SSD storage of the module system controller	Static

Table 16: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Password	External	RAM	Plaintext	Manual	Electronic	
Module SSD to External Entity (TLS)	SSD	External	Encrypted	Automated	Electronic	KAS-TLS
Module SSD to External Entity	SSD	External	Plaintext	Automated	Electronic	
Module RAM to External entity	RAM	External	Plaintext	Automated	Electronic	
External Entity to Module	External	RAM	Plaintext	Manual	Electronic	

Table 17: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Secure Erase	Zeroizes all SSPs present in module	The CLI command overwrites the storage location keys with 0's, making them irretrievable.	Issue CLI command or select option in GUI menu
Close TLS Session	Zeroizes all TLS session keys	The protocol implementation zeroizes all session keys stored temporarily, in RAM	Close session, or remove power from connected device
Close SSH Session / Terminate SSH application	Zeroizes all SSH session keys	The protocol implementation zeroizes all session keys stored temporarily, in RAM	Close session, or remove power from connected device
Reboot System / Remove power	Reboot System / Remove power form host system	SSPs stored temporarily in RAM are zeroised and unretrievable	Reboot system (Command, GUI Option, Power Button), Remove power (Power button / Unplug)

Table 18: SSP Zeroization Methods

When the *Secure Erase*, *Close TLS Session*, and *Close SSH Session / Terminate SSH application* zeroization methods are used, the memory locations occupied by the zeroized SSPs are overwritten with zeros before freeing the memory location for use. Temporary values generated during the module integrity check are zeroized when no longer needed. Procedural zeroization of SSPs stored temporarily in module RAM, must be performed under control of the module operator.

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TLS RSA public key	Public portion of RSA key pair used by for establishing TLS (HTTPS) connection	2048, 3072, 4096-bits - 112-150-bits	Public Key - PSP	RSA KeyGen		RSA SigVer
TLS RSA private key	Private portion of RSA key pair used by for establishing TLS (HTTPS) connection	2048, 3072, 4096-bits - 112-150-bits	Private Key - CSP	RSA KeyGen		RSA SigGen
TLS ECDSA public key	Public portion of ECDSA key pair used by for establishing TLS (HTTPS) connection	P-256, P-384 - 128, 192-bits	Public Key - PSP	ECDSA KeyGen		ECDSA SigVer
TLS ECDSA private key	Private portion of ECDSA key pair used by for establishing TLS (HTTPS) connection	P-256, P-384 - 128, 192-bits	Private Key - CSP	ECDSA KeyGen		ECDSA SigGen
TLS EC Diffie-Hellman public key	Public portion of EC Diffie-Hellman key pair used for key agreement during TLS handshake	P-256, P-384 - 128, 192-bits	Public Key - PSP	ECDSA KeyGen		KAS-TLS
TLS EC Diffie-Hellman private key	Private portion of EC Diffie-Hellman key pair used for key agreement during TLS handshake	P-256, P-384 - 128, 192-bits	Private Key - CSP	ECDSA KeyGen		KAS-TLS

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TLS pre-primary secret	Shared secret computed as a result of EC Diffie-Hellman key agreement	256, 384 bits - 128, 192-bits	Secret - CSP		KAS-TLS	KAS-TLS
TLS primary secret	Secret value output from TLS KDF	384 bits - 192-bits	Secret - CSP	KAS-TLS		KAS-TLS
TLS derived session key	Encryption and MAC secret keys used for protecting TLS session communications	128-256-bits (AES), 112-192-bits (HMAC) - 128-256-bits (AES), 112-192-bits (HMAC)	Symmetric Key - CSP	KAS-TLS		AES-CBC AES-GCM HMAC
SSH ECDSA public key	Public portion of ECDSA key pair used by for establishing SSH connection	P-256, P-384 - 128, 192-bits	Public Key - PSP	ECDSA KeyGen		KAS-SSH
SSH ECDSA private key	Private portion of ECDSA key pair used by for establishing SSH connection	P-256, P-384 - 128, 192-bits	Private Key - CSP	ECDSA KeyGen		KAS-SSH
SSH EC Diffie-Hellman public key	Public portion of EC Diffie-Hellman key pair used for key agreement during SSH handshake	P-256, P-384 - 128, 192-bits	Public Key - PSP	ECDSA KeyGen		KAS-SSH
SSH EC Diffie-Hellman private key	Private portion of EC Diffie-Hellman key pair used for key agreement during SSH handshake	P-256, P-384 - 128, 192-bits	Private Key - CSP	ECDSA KeyGen		KAS-SSH

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
SSH shared secret	Shared secret computed as a result of EC Diffie-Hellman key agreement	256-bits, 384-bits - 128, 192-bits	Secret - CSP		KAS-SSH	KAS-SSH
SSH derived session key	Encryption and MAC secret keys used for protecting SSH session communications	128, 192 (CBC only), 256 bits (AES) 112-192-bits (HMAC) - 128, 192 (CBC only), 256-bits (AES) 112-192-bits (HMAC)	Symmetric Key - CSP	KAS-SSH		AES-CBC AES-CTR AES-GCM HMAC
Password	Secret value used by operator to authenticate to the module	Variable length, minimum 8 characters - 1 in 2.8x10^15	Authentication Data - CSP			
Entropy input string	Random bits obtained by entropy source	256-bits - 256-bits	Secret - CSP			RSA KeyGen ECDSA KeyGen
DRBG seed	String of bits used as input to DRBG	440-bits - 256-bits	Secret - CSP			RSA KeyGen ECDSA KeyGen
DRBG internal state V	"V" portion of the DRBG internal state, as defined by SP 800-90Ar1	128-bits - 128-bits	Secret - CSP			RSA KeyGen ECDSA KeyGen

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
DRBG internal state key	"Key" portion of the DRBG internal state, as defined by SP 800-90Ar1	256-bits - 256-bits	Secret - CSP			RSA KeyGen ECDSA KeyGen
AES-GCM IV	Initialization vector for AES-GCM within SSH and TLS protocols	96-bits - -	Initialization Vector - PSP	AES-GCM		AES-GCM (A4783)

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLS RSA public key	Module SSD to External Entity	SSD:Plaintext		Secure Erase	TLS RSA private key:Paired With
TLS RSA private key	Module SSD to External Entity (TLS)	SSD:Plaintext		Secure Erase	TLS RSA public key:Paired With
TLS ECDSA public key	Module SSD to External Entity	SSD:Plaintext		Secure Erase	TLS ECDSA private key:Paired With
TLS ECDSA private key	Module SSD to External Entity (TLS)	SSD:Plaintext		Secure Erase	TLS ECDSA public key:Paired With
TLS EC Diffie-Hellman public key	Module RAM to External entity External Entity to Module	RAM:Plaintext		Secure Erase Close TLS Session Reboot System / Remove power	TLS EC Diffie-Hellman private key:Paired With
TLS EC Diffie-Hellman private key		RAM:Plaintext		Secure Erase Close TLS Session Reboot System / Remove power	TLS EC Diffie-Hellman public key:Paired With
TLS pre-primary secret		RAM:Plaintext		Secure Erase Close TLS Session	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Reboot System / Remove power	
TLS primary secret		RAM:Plaintext		Secure Erase Close TLS Session Reboot System / Remove power	TLS pre-primary secret:Derived From
TLS derived session key		RAM:Plaintext		Secure Erase Close TLS Session Reboot System / Remove power	TLS primary secret:Derived From
SSH ECDSA public key	Module SSD to External Entity	SSD:Plaintext		N/A	SSH ECDSA private key:Paired With
SSH ECDSA private key		SSD:Plaintext		Secure Erase Close SSH Session / Terminate SSH application Reboot System / Remove power	SSH ECDSA public key:Paired With
SSH EC Diffie-Hellman public key	Module RAM to External entity External Entity to Module	RAM:Plaintext		Secure Erase Close SSH Session / Terminate SSH application Reboot System / Remove power	SSH EC Diffie-Hellman private key:Paired With
SSH EC Diffie-Hellman private key		RAM:Plaintext		Secure Erase Close SSH Session / Terminate SSH application	SSH EC Diffie-Hellman public key:Paired With

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Reboot System / Remove power	
SSH shared secret		RAM:Plaintext		Secure Erase Close SSH Session / Terminate SSH application Reboot System / Remove power	
SSH derived session key		RAM:Plaintext		Secure Erase Close SSH Session / Terminate SSH application Reboot System / Remove power	SSH shared secret:Derived From
Password	Password	RAM:Plaintext SSD:Obfuscated		Secure Erase Reboot System / Remove power	
Entropy input string		RAM:Plaintext		Secure Erase Reboot System / Remove power	
DRBG seed		RAM:Plaintext		Secure Erase Reboot System / Remove power	
DRBG internal state V		RAM:Plaintext		Secure Erase Reboot System / Remove power	
DRBG internal state key		RAM:Plaintext		Secure Erase Reboot System /	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
				Remove power	
AES-GCM IV		RAM:Plaintext		Secure Erase Reboot System / Remove power	TLS derived session key:Used With SSH derived session key:Used With

Table 20: SSP Table 2

9.5 Transitions

The module's SHA-1 implementation will be considered non-approved for all uses as of January 1, 2030.

10 Self-Tests

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-384 (A4783)	HMAC-SHA2-384	MAC	SW/FW Integrity	Console message	MAC verification (Compare computed HMAC value of entire software image to stored MAC)

Table 21: Pre-Operational Self-Tests

The module's integrity check is the only pre-operational self-test implemented in the module. This can be invoked on demand by power-cycling the module, or by the Administrator (CO), by inputting the following CLI command:

system security integrity-check controllers

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-GCM (A4783)	256-bit	KAT	CAST	Console message	Encrypt	On Startup
AES-CBC (A4783)	256-bit	KAT	CAST	Console message	Decrypt	On Startup
SHA-1 (A4783)	-	KAT	CAST	Console message	Hash	On Startup
SHA2-256 (A4783)	SHA2-256	KAT	CAST	Console message	Hash	On Startup
HMAC-SHA2-384 (A4783)	HMAC-SHA2-384	KAT	CAST	Console message	MAC verification	On Startup
ECDSA SigGen (FIPS186-5) (A4782)	P-256	KAT	CAST	Console message	Signature Generation	On Startup
ECDSA SigVer (FIPS186-5) (A4782)	P-256	KAT	CAST	Console message	Signature Verification	On Startup
RSA SigGen (FIPS186-5) (A4782)	2048-bit	KAT	CAST	Console message	Signature Generation	On Startup
RSA SigVer (FIPS186-5) (A4782)	2048-bit	KAT	CAST	Console message	Signature Generation	On Startup
TLS v1.2 KDF RFC7627 (A4782)	SHA2-256	KAT	CAST	Console message	Key Derivation	On Startup

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KDF SSH (A4782)	SHA-1	KAT	CAST	Console message	Key Derivation	On Startup
Counter DRBG (A4783)	AES-256	KAT	CAST	Console message	DRBG CAST exercises the following functions from NIST SP 800-90A: Instantiate (11.3.2), Generate (11.3.3) and Reseed (11.3.4).	On Startup
ECDSA KeyGen (FIPS186-5) (A4782)	P-256	PCT	PCT	Console message	Key Pair generation and validation	Generation of ECDSA Key Pair
RSA KeyGen (FIPS186-5) (A4782)	2048-bit	PCT	PCT	Console message	Key Pair generation and validation	Generation of RSA Key Pair
KAS-ECC-SSC Sp800-56Ar3 (A4782)	P-256	KAT	CAST	Console message	Shared Secret Computation	On Startup
KAS-ECC-SSC Sp800-56Ar3 - ECDH PCT	P-256	PCT	PCT	Console message	Signature Verification	Generation of ECDH Key Pair
Repetition Count Test (Entropy)	-	RCT	CAST	Console message	Repetition Count Test on entropy source	On Startup, Continuously on entropy source outputs
Adaptive Proportion Test	-	APT	CAST	Console message	Adaptive Proportion Test on entropy source	On Startup, Continuously on entropy source outputs
SHA3-256 (A4093)	256-bit	KAT	CAST	Console message	Hash	On Startup

Table 22: Conditional Self-Tests

The module's cryptographic algorithm self-tests (CAST) are run as part of module startup and can be invoked on demand by power-cycling the module.

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-384 (A4783)	MAC	SW/FW Integrity	On Demand	Manual, by module reboot

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-GCM (A4783)	KAT	CAST	On Demand	Manual, by module reboot
AES-CBC (A4783)	KAT	CAST	On Demand	Manual, by module reboot
SHA-1 (A4783)	KAT	CAST	On Demand	Manual, by module reboot
SHA2-256 (A4783)	KAT	CAST	On Demand	Manual, by module reboot
HMAC-SHA2-384 (A4783)	KAT	CAST	On Demand	Manual, by module reboot
ECDSA SigGen (FIPS186-5) (A4782)	KAT	CAST	On Demand	Manual, by module reboot
ECDSA SigVer (FIPS186-5) (A4782)	KAT	CAST	On Demand	Manual, by module reboot
RSA SigGen (FIPS186-5) (A4782)	KAT	CAST	On Demand	Manual, by module reboot
RSA SigVer (FIPS186-5) (A4782)	KAT	CAST	On Demand	Manual, by module reboot
TLS v1.2 KDF RFC7627 (A4782)	KAT	CAST	On Demand	Manual, by module reboot
KDF SSH (A4782)	KAT	CAST	On Demand	Manual, by module reboot
Counter DRBG (A4783)	KAT	CAST	On Demand	Manual, by module reboot
ECDSA KeyGen (FIPS186-5) (A4782)	PCT	PCT	On Key Pair Generation	-
RSA KeyGen (FIPS186-5) (A4782)	PCT	PCT	On Key Pair Generation	-
KAS-ECC-SSC Sp800-56Ar3 (A4782)	KAT	CAST	On Demand	Manual, by module reboot

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 - ECDH PCT	PCT	PCT	On Key Pair Generation	-
Repetition Count Test (Entropy)	RCT	CAST	On Demand, Continuous	-
Adaptive Proportion Test	APT	CAST	On Demand, Continuous	-
SHA3-256 (A4093)	KAT	CAST	On Demand	Manual, by module reboot

Table 24: Conditional Periodic Information

The module is security level 2 and not subject to periodic self-test requirements.

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Error	All data output and cryptographic services are inhibited in the error state.	If any self-test fails	Reboot module when all System Controllers are in Error mode	<name of test> self-test failed.

Table 25: Error States

The module transitions to an error state when any module self-test fails. All data output and cryptographic functions are inhibited when in the error state.

If both the primary (active) and secondary (standby) system controllers are active when the module enters the error state, the module will immediately pass control from the active to the standby system controller, and transition back to the normal operation state.

If there is only 1 active system controller, such as when the primary controller has already failed and passed control to the secondary controller, then module must be rebooted to clear the error condition.

10.5 Operator Initiation of Self-Tests

All pre-operational and cryptographic algorithm self-tests can be invoked on demand by the module operator by power-cycling the module.

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

- The hardware platform (TOEPP) is shipped directly from the Vendor to the customer via a professional carrier and the shipment tracked by that carrier.
- The Crypto Officer should inspect the product packaging for signs of tamper during transit.
- The Crypto Officer should verify the product matches the description and pictures in this Security Policy and ensure that all components are present in the package.
- The Crypto Officer should follow the instructions found in the "Install and Upgrade Software" here: https://techdocs.f5.com/en-us/velos-1-5-0/velos-systems-installation-upgrade/title-install-upgrade-software.html#ch-title-install-sw
- The Crypto Officer should follow the instructions found in the "F5 VELOS system initial configuration" guide, Section "License the system automatically from the system controller webUI" here: https://techdocs.f5.com/en-us/hardware/velos-systems-getting-started/gs-system-initial-config.html#velos-setup-wizard-overview
- The device will reboot and return to operation in the Approved mode.
- The administrator should verify that the module is operating in the approved mode of operation, with the FIPS license installed at any time by issuing the commands listed in Section 2.4: Modes of Operation, of this security policy and verifying that the output matches the expected output.

11.2 Administrator Guidance

The administrator may verify that the module is operating in the approved mode of operation, with the FIPS license installed at any time by issuing the commands listed in Section 2.4: Modes of Operation, of this security policy and verifying that the output matches the expected output. The services available to administrators are listed in Section 4.3: Approved Services, of this security policy.

11.3 Non-Administrator Guidance

There is no specific non-administrator for proper use of the module once configured into the approved mode of operation. The services available to non-administrators are listed in Section 4.3: Approved Services, of this security policy.

12 Mitigation of Other Attacks

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