F5, Inc.

FIPS 140-3 Non-Proprietary Security Policy

BIG-IP Tenant Cryptographic Module

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

Prepared for:
F5, Inc.
801 Fifth Ave
Seattle, WA 98104
www.f5.com

Table of Contents

List of Tables

List of Figures

Copyrights and Trademarks

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

Intel® Xeon® and Intel® Atom® processors are Registered trademarks of Intel Corporation.

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy that contains the security rules under which the BIG-IP Tenant Cryptographic 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 an Overall Security Level 2 module.

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	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
	Overall Level	2

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

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. Solutions built on this platform are load balancers. They are full proxies that give visibility into, and the power to control—inspect and encrypt or decrypt—all the traffic that passes through your network.

Purpose and Use:

The BIG-IP Tenant Cryptographic Module is 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. BIG-IP hardware and the firmware components form 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.

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.

Module Type: Firmware

Module Embodiment: Multi-Chip Standalone

Cryptographic Boundary: The cryptographic boundary of the module is defined by the red dotted line in Figure 1. The block diagram below shows the module, its interfaces with the operational environment and the delimitation of its cryptographic boundary. Figure 1 also depicts the flow of status output (SO), control input (CI), data input (DI) and data output (DO). Description of the ports and interfaces can be found in Table- Ports and Interfaces.

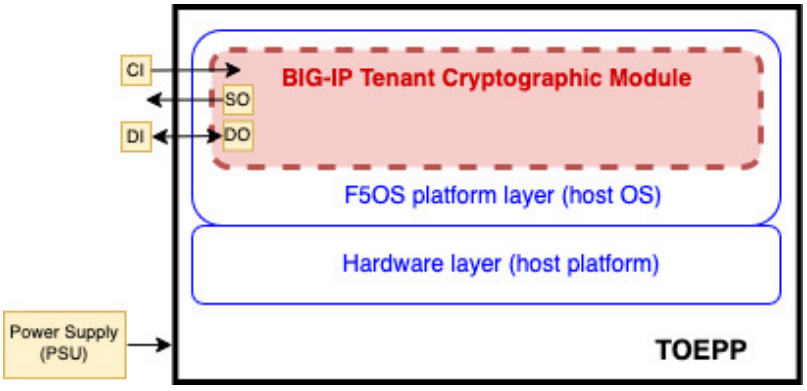

Figure 1: Block Diagram

Tested Operational Environment’s Physical Perimeter (TOEPP):
The TOEPP defined by the tested platforms listed in Table 2 is delineated by the black rectangle in Figure -Block Diagram. The TOEPP is represented by the enclosure of each of the tested platforms pictured below.

Figure 2 – r4800

Figure 3 – r5900

Figure 4 – r5920-DF

Figure 5 – r10900, r10920-DF and r12900
(same chassis for the test three platforms)

Figure 6 – VELOS CX410-BX110
with 7 filler panels and one blade. The tested blade VELOS CX410-BX110 in slot #1 delineated with red rectangle.

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

N/A for this module.

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

Package or File Name	Software/ Firmware Version	Features	Integrity Test
BIGIP-17.1.0.1-0.0.4.ALL-F5OS.qcow.zip.bundle	17.1.0.1	N/A	HMAC-SHA-2-384

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

The executable code is defined by the firmware version indicated in the table -*Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets).* All code belonging to this firmware version is the executable code of the module.

Tested Module Identification – Hybrid Disjoint Hardware:

N/A for this module.

Tested Operational Environments - Software, Firmware, Hybrid:

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
BIG-IP 17.1.0.1 Tenant	r4800	Intel® Atom® P5342 Snow Ridge	No	F5OS-A 1.5.1	17.1.0.1
BIG-IP 17.1.0.1 Tenant	r5900	Intel® Xeon® Silver 4314 Ice Lake	No	F5OS-A 1.5.1	17.1.0.1
BIG-IP 17.1.0.1 Tenant	r5920-DF	Intel® Xeon® Silver 4314 Ice Lake	No	F5OS-A 1.5.1	17.1.0.1
BIG-IP 17.1.0.1 Tenant	r10900	Intel® Xeon® Gold 6312U Ice Lake	No	F5OS-A 1.5.1	17.1.0.1
BIG-IP 17.1.0.1 Tenant	r10920-DF	Intel® Xeon® Gold 6312U Ice Lake	No	F5OS-A 1.5.1	17.1.0.1
BIG-IP 17.1.0.1 Tenant	r12900-DS	Intel® Xeon® Platinum 8351N Ice Lake	No	F5OS-A 1.7.0	17.1.0.1
BIG-IP 17.1.0.1 Tenant	VELOS CX410 BX110	Intel® Xeon® D-2177NT Skylake	No	F5OS-C 1.6.0	17.1.0.1

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

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

N/A for this module.

2.3 Excluded Components

None

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved mode	Automatically entered whenever an approved service is requested	Approved	Equivalent to the indicator of the requested service (Service Indicator: Approved, SSH connection successful)
Non-Approved mode	Only non-approved security functions can be used	Non-Approved	No service indicator required for non-approved services per IG 2.4.C

Table 4: Modes List and Description

Mode Change Instructions and Status:

The module enters the Approved Mode after the pre-operational self-tests and conditional algorithms self-tests (CASTs) have completed successfully.

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.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A3729	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CBC	A3730	Direction - Decrypt, Encrypt Key Length - 128, 256	SP 800-38A
AES-CCM	A3729	Key Length - 128, 192, 256 Tag Length - 112, 128, 32, 48, 64, 80, 96 IV Length - IV Length: 56, 64, 80, 88, 96, 104 Payload Length - Payload Length: 0-256 Increment 8 AAD Length - AAD Length: 0, 524288	SP 800-38C
AES-CCM	A3730	Key Length - 128, 256 Tag Length - 112, 128, 32, 48, 64, 80, 96 IV Length - IV Length: 56, 64, 80, 88, 96, 104 Payload Length - Payload Length: 0-256 Increment 8 AAD Length - AAD Length: 0, 524288	SP 800-38C
AES-CTR	A3729	Direction - Decrypt, Encrypt Key Length - 128, 192, 256 Payload Length - Payload Length: 8-128 Increment 8 Supports Counter larger than maximum value - No Incremental Counter - Yes Counter Tests Performed - Yes	SP 800-38A
AES-GCM	A3729	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
AES-GCM	A3730	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 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

Algorithm	CAVP Cert	Properties	Reference
Counter DRBG	A3729	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - AES-256 Derivation Function Enabled - No, Yes Additional Input - Additional Input: 0 Entropy Input - Entropy Input: 256, Entropy Input: 384 Nonce - Nonce: 128 Personalization String Length - Personalization String Length: 0-256 Increment 8, Personalization String Length: 0-384 Increment 8 Returned Bits - 512	SP 800-90A Rev. 1
Counter DRBG	A3730	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 8 Returned Bits - 512	SP 800-90A Rev. 1
ECDSA KeyGen (FIPS186-4)	A3729, A3730	Curve - P-256, P-384 Secret Generation Mode - Testing Candidates	FIPS 186-4
ECDSA KeyGen (FIPS186-5)	A3729, A3730	Curve - P-256, P-384 Secret Generation Mode - testing candidates	FIPS 186-5
ECDSA KeyVer (FIPS186-4)	A3729, A3730	Curve - P-256, P-384	FIPS 186-4
ECDSA KeyVer (FIPS186-5)	A3729, A3730	Curve - P-256, P-384	FIPS 186-5
ECDSA SigGen (FIPS186-4)	A3729, A3730	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigGen (FIPS186-5)	A3729, A3730	Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 Component - No	FIPS 186-5
ECDSA SigVer (FIPS186-4)	A3729, A3730	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-5)	A3729, A3730	Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA2-512	FIPS 186-5
HMAC-SHA-1	A3729	MAC - MAC: 160 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1
HMAC-SHA2-256	A3729, A3730	MAC - MAC: 256 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1

Algorithm	CAVP Cert	Properties	Reference
HMAC-SHA2-384	A3729, A3730	MAC - MAC: 384 Key Length - Key Length: 8, 16, 64, 128, 1024	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A3729, A3730	Domain Parameter Generation Methods - P-256, P-384 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KAS-FFC-SSC Sp800-56Ar3	A3729, A3730	Domain Parameter Generation Methods - ffdhe2048, ffdhe3072, ffdhe4096 Scheme - dhEphem - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF SSH (CVL)	A3729	Cipher - AES-128, AES-256 Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1
RSA KeyGen (FIPS186-4)	A3729	Key Generation Mode - B.3.3 Modulo - 2048, 3072, 4096 Primality Tests - Table C.2 Info Generated By Server - No Public Exponent Mode - Fixed Fixed Public Exponent - 010001 Private Key Format - Standard	FIPS 186-4
RSA KeyGen (FIPS186-5)	A3729	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 (FIPS186-4)	A3729	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA2-256	FIPS 186-4
RSA SigGen (FIPS186-4)	A3730	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA2-256	FIPS 186-4
RSA SigGen (FIPS186-5)	A3729	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5	FIPS 186-5
RSA SigGen (FIPS186-5)	A3730	Hash Pair - Hash Algorithm - SHA2-256 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Mask Function - mgf1	FIPS 186-5

Algorithm	CAVP Cert	Properties	Reference
RSA SigVer (FIPS186-4)	A3729	Signature Type - PKCS 1.5 Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA-1 Salt Length - 20 Public Exponent Mode - Fixed Fixed Public Exponent - 010001	FIPS 186-4
RSA SigVer (FIPS186-4)	A3730	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA-1 Salt Length - 20 Public Exponent Mode - Fixed Fixed Public Exponent - 010001	FIPS 186-4
RSA SigVer (FIPS186-5)	A3729	Hash Pair - Hash Algorithm - SHA-1 Salt Length - 20 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5 Fixed Public Exponent - 010001 Public Exponent Mode - fixed	FIPS 186-5
RSA SigVer (FIPS186-5)	A3730	Hash Pair - Hash Algorithm - SHA-1 Salt Length - 20 Modulo - 2048, 3072, 4096 Signature Type - pkcs1v1.5, pss Fixed Public Exponent - 010001 Public Exponent Mode - fixed Mask Function - mgf1	FIPS 186-5
Safe Primes Key Generation	A3729, A3730	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096	SP 800-56A Rev. 3
Safe Primes Key Verification	A3729, A3730	Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096	SP 800-56A Rev. 3
SHA2-256	A3729, A3730	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A3729, A3730	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
TLS v1.2 KDF RFC7627 (CVL)	A3729, A3730	Hash Algorithm - SHA2-256, SHA2-384 Key Block Length - Key Block Length: 1024	SP 800-135 Rev. 1

Table 5: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
Cryptographic Key Generation (CKG)	Key Type:asymmetric	N/A	Section 4 example 1 of SP800-133r2

Table 6: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

Name	Use and Function
HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2-512	Message authentication TLS
Triple-DES, Camellia, SEED	Symmetric encryption and decryption TLS
HMAC-SHA2-256, HMAC-SHA2-512, AES-GCM	Message authentication in IPsec/ IKEv2 protocol
PKCS #1 v1.5 scheme with modulus other than 2048, 3072 or 4096 bits, for all SHA sizes; PKCS #1 v1.5 and PSS schema with modulus size 2048, 3072, 4096 bits with SHA-1, SHA2-224, SHA2-512; with ANS X9.31 standard;	RSA signature generation and verification
ECDSA with curves P-256/ P-384 with SHA-1/SHA2-224, ECDSA using curves other than P-256 and P-384 with all SHA sizes	ECDSA signature generation and verification
RSA with modulus sizes up to 16384 bits	RSA encrypt / decrypt
DSA with all key and SHA sizes	DSA domain parameter generation, domain parameter verification, key pair generation, signature generation and verification
Diffie-Hellman using MODP1024, MODP2048 groups	Shared secret computation in IPsec/IKE protocol
MD5/ SHA-1/ SHA2-224 / SHA2-512	Key Derivation function in the context of TLS KDF
EdDSA with Ed25519	EdDSA digital signature
SHA-1, AES-ECB, RSA- signature verification	SNMP
TLS ciphersuites implemented by f5-rest-node	TLS used in SSL Orchestrator (SSLO)
RSA keypair with 2048, 3072 and 4096 (REST API)	iControl representation state transfer (REST) access
EC Diffie-Hellman Ephemeral Unified with curves other than P-256, P-384. EC Diffie-Hellman using onePassDH / StaticUnified schemes. Diffie-Hellman using groups other than ffdhe2048, ffdhe3072, ffdhe4096	Shared secret computation
Triple-DES, AES-GCM-128, AES-192, AES-256	Symmetric encryption and decryption in IPsec /IKEv2

Table 7: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
Key Wrapping/Unwrapping with encryption and authentication in TLS	KTS-Wrap	Key Wrapping, Key Unwrapping in the context of TLS	Standard:SP 800-38F, FIPS 197 Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG D.G:Combination of AES-CBC and HMAC	AES-CBC: (A3729, A3730) HMAC-SHA2-256: (A3729, A3730) HMAC-SHA2-384: (A3729, A3730)
Key Wrapping/Unwrapping with encryption and authentication in SSH	KTS-Wrap	Key Wrapping, Key Unwrapping in the context of SSH	Standard:SP 800-38F, FIPS 197 Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG D.G:Combination of AES-CBC/AES-CTR and HMAC	AES-CTR: (A3729) AES-CBC: (A3729) HMAC-SHA-1: (A3729) HMAC-SHA2-256: (A3729)
Key Wrapping/Unwrapping with authenticated encryption	KTS-Wrap	Key Wrapping, Key Unwrapping	Standard:SP 800-38F, FIPS 197 Caveat:Key establishment methodology provides between 128 and 256 bits of security strength IG D.G:Using approved authenticated symmetric encryption with AES-GCM or AES-CCM	AES-CCM: (A3729, A3730) AES-GCM: (A3729, A3730)
Random Number Generation in Control Plane	DRBG	Generate random bytes	Compliance:Compliant with SP800-90ARev1 Properties:with / without derivation function, prediction resistance disabled / enabled	Counter DRBG: (A3729)
Signature generation	DigSig-SigGen	Generate a digital signature		ECDSA SigGen (FIPS186-5): (A3729, A3730) ECDSA SigGen (FIPS186-4): (A3729, A3730)

Name	Type	Description	Properties	Algorithms
				RSA SigGen (FIPS186-5): (A3729, A3730) RSA SigGen (FIPS186-4): (A3729, A3730)
Signature verification	DigSig-SigVer	Verify a digital signature		ECDSA SigVer (FIPS186-5): (A3729, A3730) ECDSA SigVer (FIPS186-4): (A3729, A3730) RSA SigVer (FIPS186-5): (A3729, A3730) RSA SigVer (FIPS186-4): (A3729, A3730)
Key pair generation	AsymKeyPair-KeyGen CKG	Generate an ECDSA, ECDH, DH or RSA key pair		ECDSA KeyGen (FIPS186-5): (A3729, A3730) ECDSA KeyGen (FIPS186-4): (A3729, A3730) RSA KeyGen (FIPS186-5): (A3729) RSA KeyGen (FIPS186-4): (A3729) Safe Primes Key Generation: (A3730, A3729) Cryptographic Key Generation (CKG): ()
Key pair verification	AsymKeyPair-KeyVer	Verify an ECDSA or ECDH or DH key pair		ECDSA KeyVer (FIPS186-5): (A3729, A3730) ECDSA KeyVer (FIPS186-4): (A3729, A3730) Safe Primes Key Verification: (A3729, A3730)
Message Digest	SHA	Compute a message digest		SHA2-256: (A3729, A3730) SHA2-384: (A3729, A3730)

Name	Type	Description	Properties	Algorithms
TLS Handshake (ECC)	KAS-Full	Key agreement	Key derivation:IG 2.4.B SP 800-135rev1 CVL IG D.F:Scenario 2 (path 2) Key confirmation:no Caveat:Key establishment methodology provides between 128 and 192 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A3729, A3730) TLS v1.2 KDF RFC7627: (A3729, A3730)
SSH Handshake	KAS-Full	Key agreement	Caveat:Key establishment methodology provides between 128 and 192 bits of key strength IG D.F:Scenario 2 (path 2) Key confirmation:no Key derivation:IG 2.4.B SP 800-135rev1 CVL	KAS-ECC-SSC Sp800-56Ar3: (A3729) KDF SSH: (A3729)
Random Number Generation in Data Plane	DRBG	Generate random bytes	Compliance:Compliant with SP800-90ARev1 Properties:AES 256 in CTR mode, with derivation function, prediction resistance disabled	Counter DRBG: (A3730)
TLS Handshake (FFC)	KAS-Full	Key agreement	Key derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology provides between 112 to 150-bits of security strength IG D.F:Scenario 2 (path 2) Key confirmation:no	KAS-FFC-SSC Sp800-56Ar3: (A3729, A3730) TLS v1.2 KDF RFC7627: (A3729, A3730)

Table 8: Security Function Implementations

2.7 Algorithm Specific Information

AES-GCM: The IV for AES-GCM is constructed in compliance with IG C.H scenario 1a (TLS 1.2).

For TLS 1.2, the module offers the AES-GCM implementation and uses the context of Scenario 1a of IG C.H. The module is compliant with SP800-52r2 section 3.3.1 and the mechanism for IV generation is compliant with RFC5288.

The module's implementation of AES-GCM is compliant to IG C.H option i) where module implements TLS protocol. The design of the TLS protocol implicitly ensures that the counter (the nonce_explicit part of the IV) does not exhaust the maximum number of possible values for a given session key. In the event the module's power is lost and restored, the consuming application must ensure that a new key for use with the AES-GCM key encryption or decryption under this scenario shall be established.

RSA modulus sizes (IG C.F): RSA Signature Generation uses modulus sizes greater than or equal to 2048 bits. All the modulus sizes supported by the module have been ACVP tested.

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.

FIPS 186-4, has been superseded by FIPS 186-5 on February 4, 2024. For the current module context, FIPS 186-4 can still be used in the approved mode as the module was submitted before the transition date. See IG C.K for details. One operational environment, tested later than the others, received CAVP certificates with FIPS 186-5.

2.8 RBG and Entropy

Cert Number	Vendor Name
E74	F5, Inc.

Table 9: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
CPU Jitter Entropy Source for F5 cryptographic module (SHA_3) version 3.4.0	Non-Physical	BIG-IP 17.1.0.1 Tenant with Intel® Atom® P5342 Snow Ridge NS, BIG-IP 17.1.0.1 Tenant with Intel® Xeon® Silver 4314 Ice Lake-SP, BIG-IP 17.1.0.1 Tenant with Intel® Xeon® Gold 6312U Ice Lake-SP, BIG-IP 17.1.0.1 Tenant with Intel® Xeon® Platinum 8351N Ice Lake-SP, and BIG-IP 17.1.0.1 Tenant with Intel® Xeon® D-2177NT Skylake-D	256 bits	Full entropy	SHA3-256 (CAVP cert. #A3769)

Table 10: Entropy Sources

The module employs a Deterministic Random Bit Generator (DRBG) based on [SP800-90Ar1] for the generation of random values used in asymmetric keys, and for providing a RNG service to calling applications. The approved DRBG provided by the module is the CTR_DRBG with AES-256.

The output of entropy source provides full entropy to seed and reseed SP800-90Ar1 DRBG during initialization (seed) and reseeding (reseed).

In accordance with FIPS 140-3 IG D.L, the 'Entropy input string', 'seed', 'DRBG internal state (V and key values)' are considered CSPs by the module.

No non-DRBG functions or instances are able to access the DRBG internal state.

2.9 Key Generation

The module implements asymmetric key generation methods according to SP 800-133r2 section 5. The key generation methods are specified in the Security Function Implementations table.

2.10 Key Establishment

The module implements SSP agreement, compliant with IG D.F scenario 2 (path 2). Additionally, the module implements SSP transport, compliant with IG D.G. The Key Establishment methods are specified in the *Security Function Implementations* table.

2.11 Industry Protocols

GCM with internal IV generation in the approved mode is compliant with version 1.2 of the TLS protocol (RFC 5288) and shall only be used in conjunction with the TLS protocol.

Additionally, the module implements the TLS 1.2 and SSH key derivation functions for use in the TLS protocol and SSH protocol (RFC 4253 updated by RFC 6668).

The TLS 1.2 and SSHv2 protocols have not been reviewed or tested by the CAVP or CMVP.

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	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

Table 11: Ports and Interfaces

The logical interfaces are the commands through which the 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.

The physical ports are interpreted to be the physical ports of the hardware platform on which it runs.

The module does not implement Control Output interface.

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
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.	Password-based authentication	1/676,000,000	3/676,000,000
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.	PLACEHOLDER	1/(2^128)	3/676,000,000

Table 12: Authentication Methods

The module supports 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:

- Command Line Interface (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).

4.2 Roles

Name	Type	Operator Type	Authentication Methods
Administrator	Role	Crypto Officer (CO)	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
Auditor	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
Certificate Manager	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
Firewall Manager	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
iRule Manager	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
Operator	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)
Resource Manager	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)

Name	Type	Operator Type	Authentication Methods
User Manager	Role	User	Role-based authentication with Password (CLI or WebUI) Role-based authentication with SSH ECDSA key-pair (CLI only)

Table 13: Roles

At initialization of the module, the CO is the only available role. Only the CO can create the user roles.

4.3 Approved Services

The service indicator gets recorded in the high speed logging /var/log remote.log file after the service is executed. For approved services, the indicator is identified with the log message 'Service Indicator: Approved' and for non-approved services the log message includes 'Service Indicator: Not Approved'.

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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
List users	Display all User accounts	None	None	List of all user accounts including self	None	Administrator User Manager Resource Manager Auditor
Create additional User	Create additional User	None	Username / password	Confirmation of account creation	None	Administrator - Password: W User Manager - Password: W
Modify existing Users	Modify existing Users	None	Username	Confirmation of account modification	None	Administrator User Manager
Delete User	Delete User	None	Username	Confirmation of deletion	None	Administrator User Manager
Unlock User	Remove lock from user who has exceeded login attempts	None	Username	Confirmation of unlock	None	Administrator User Manager

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Update own password	Update own password	None	Own password	Confirmation of update of password	None	Administrator - Password: W Auditor - Password: W Certificate Manager - Password: W Firewall Manager - Password: W iRule Manager - Password: W Operator - Password: W Resource Manager - Password: W User Manager - Password: W
Update others password	Update others password	None	Username / password	Confirmation of update	None	Administrator - Password: W User Manager - Password: W
Configure Password Policy	Set password policy features	None	New password policy	Confirmation of configuration change	None	Administrator
Create TLS Certificate	Self-signed certificate creation	Service Indicator: Approved	Certificate identification information	Confirmation of certificate creation	Signature generation	Administrator - TLS RSA private key: E - TLS ECDSA private key: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Certificate Manager - TLS RSA private key: E - TLS ECDSA private key: E Resource Manager - TLS RSA private key: E - TLS ECDSA private key: E
Create TLS Key	Used for the SSL Certificate key file	Service Indicator: Approved	Key identification information	Confirmation of key creation	Key pair generation Random Number Generation in Control Plane Random Number Generation in Data Plane	Administrator - TLS RSA private key: G - TLS RSA public key: G - TLS ECDSA private key: G - TLS ECDSA public key: G - DRBG seed : E - DRBG internal state (V and key values) : E,W - Entropy input string: E Resource Manager - TLS RSA private key: G - TLS RSA public key: G

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS ECDSA private key: G - TLS ECDSA public key: G - DRBG seed : E - DRBG internal state (V and key values) : E,W - Entropy input string: E Certificate Manager - TLS RSA private key: G - TLS RSA public key: G - TLS ECDSA private key: G - TLS ECDSA public key: G - DRBG seed : E - DRBG internal state (V and key values) : E,W - Entropy input string: E
Delete TLS Certificate /Key	Self-signed certificate / key deletion	None	Key identification information	Confirmation of key / certificate deletion	None	Administrator - TLS RSA private key: Z - TLS RSA public key: Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS ECDSA private key: Z - TLS ECDSA public key: Z Resource Manager - TLS RSA private key: Z - TLS RSA public key: Z - TLS ECDSA private key: Z - TLS ECDSA public key: Z Certificate Manager - TLS RSA private key: Z - TLS RSA public key: Z - TLS ECDSA private key: Z - TLS ECDSA public key: Z
List Certificate	Display expiration/creation dates, certificate key size, generator of installed certificates	None	List of certificates to display	Certificate expiration information	None	Administrator Auditor Certificate Manager Resource Manager
List Private Keys	List private key information (name, size, type)	None	List of private keys to display	TLS private key information	None	Administrator Auditor Certificate

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Manager Resource Manager
Establish SSH session	SSH session Key authentication, Key Exchange	SSH connection successful	User / address / password / algorithms / key sizes / key derivation	Confirmation of SSH session authentication, Confirmation of SSH session key exchange	Signature generation Signature verification SSH Handshake	Administrator - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key: E Auditor - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key: E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Certificate Manager - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key: E Firewall Manager - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key: E iRule Manager

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key : E Operator - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key : E Resource Manager - SSH ECDSA public key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key : E User Manager - SSH ECDSA public key: E - Password: W,E - SSH EC Diffie-Hellman public key: G,R,W,E - SSH EC Diffie-Hellman private key: G,R,W,E - SSH shared secret: G - SSH derived session key : E
Maintain SSH Session	SSH data encryption, decryption, integrity	SSH connection successful	SSH Derived Session key	SSH session information	Key Wrapping/Unwrapping with encryption and authentication in SSH	Administrator - SSH derived session key : E Auditor

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- SSH derived session key : E Certificate Manager - SSH derived session key : E Firewall Manager - SSH derived session key : E iRule Manager - SSH derived session key : E Operator - SSH derived session key : E Resource Manager - SSH derived session key : E User Manager - SSH derived session key : E
Establish TLS Session	TLS session signature generation and verification, key exchange	Service Indicator: Approved	Address / algorithms/ keys	Confirmation of digital signature verification of TLS session, Confirmation of establishment of TLS session	Message Digest Signature verification TLS Handshake (ECC) TLS Handshake (FFC)	Administrator - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G Auditor - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						secret : E,G - TLS derived session key: G - TLS primary secret: G Certificate Manager - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G Firewall Manager - TLS RSA public key: R - TLS ECDSA

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G iRule Manager - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key:

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G Operator - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G Resource

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Manager - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G User Manager - TLS RSA public key: R - TLS ECDSA public key: R - TLS EC Diffie-Hellman private key: E - TLS EC Diffie-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Hellman public key: W - TLS Diffie-Hellman private key: E - TLS Diffie-Hellman public key: W - TLS pre-primary secret : E,G - TLS derived session key: G - TLS primary secret: G
Maintain TLS Session	TLS data encryption, authentication	Service Indicator: Approved	TLS Derived Session key	TLS session information	Key Wrapping/Unwrapping with authenticated encryption Key Wrapping/Unwrapping with encryption and authentication in TLS	Administrator - TLS derived session key: E Auditor - TLS derived session key: E Certificate Manager - TLS derived session key: E Firewall Manager - TLS derived session key: E iRule Manager - TLS derived session key: E Operator - TLS

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						derived session key: E Resource Manager - TLS derived session key: E User Manager - TLS derived session key: E
Delete ssh-keyswap	Utility service delete ssh keys	None	SSH key to delete	Confirmation of SSH key deletion	None	Administrator - SSH ECDSA public key: Z - SSH ECDSA private key: Z Resource Manager - SSH ECDSA public key: Z - SSH ECDSA private key: Z
Reboot System	Restart the cryptographic module	Module reboots	None	Confirmation of system reboot	None	Administrator - TLS primary secret: Z - TLS derived session key: Z
Zeroization	Full system zeroization	Module end of life	Selection option	Confirmation of full system zeroization	None	Administrator - SSH ECDSA private key: Z - SSH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						ECDSA public key: Z - TLS RSA private key: Z - TLS ECDSA private key: Z - TLS ECDSA public key: Z - TLS RSA public key: Z - Password: Z
Show version	Return the HW and FW versions and the module's name	N/A	N/A	Module name and version	None	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
Show Status	Return the module status	N/A	N/A	Module status	None	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
Close TLS / SSH session	Closing TLS / SSH session	N/A	N/A	Confirmation of TLS/SSH	None	Administrator - TLS EC Diffie-

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
				session closure		Hellman private key: Z - TLS EC Diffie-Hellman public key: Z - TLS Diffie-Hellman public key: Z - TLS Diffie-Hellman private key: Z - TLS pre-primary secret : Z - TLS primary secret: Z - TLS derived session key: Z - SSH shared secret: Z - SSH derived session key : Z - SSH EC Diffie-Hellman private key: Z - SSH EC Diffie-Hellman public key: Z Auditor - TLS EC Diffie-Hellman private key: Z - TLS EC Diffie-Hellman

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						public key: Z - TLS pre-primary secret : Z - TLS primary secret: Z - TLS derived session key: Z - SSH shared secret: Z - SSH derived session key : Z - SSH EC Diffie-Hellman private key: Z - SSH EC Diffie-Hellman public key: Z - TLS Diffie-Hellman public key: Z - TLS Diffie-Hellman private key: Z Certificate Manager - TLS EC Diffie-Hellman private key: Z - TLS EC Diffie-Hellman public 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 - SSH shared secret: Z - SSH derived session key : Z - SSH EC Diffie-Hellman private key: Z - SSH EC Diffie-Hellman public key: Z - TLS Diffie-Hellman public key: Z - TLS Diffie-Hellman private key: Z Firewall Manager - 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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						session key: Z - SSH shared secret: Z - SSH derived session key : Z - SSH EC Diffie-Hellman private key: Z - SSH EC Diffie-Hellman public key: Z - TLS Diffie-Hellman public key: Z - TLS Diffie-Hellman private key: Z iRule Manager - 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 shared secret: Z

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

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

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

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Hellman private key: Z
Self-tests	Execute integrity test. Execute the CASTs	Integrity test, CASTs from section 10	N/A	Pass or fail	Random Number Generation in Control Plane Signature generation Signature verification Key pair verification Random Number Generation in Data Plane	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
Show license	Return license indication	N/A	N/A	FIPS license information	None	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
Import TLS Certificate	Import TLS Certificate	None	Certificate to import	Confirmation of import of certificate	None	Administrator - TLS RSA public key: W - TLS ECDSA public key: W Resource Manager - TLS RSA public key: W - TLS ECDSA public key: W Certificate Manager

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS RSA public key: W - TLS ECDSA public key: W
Export Certificate File	Export Certificate File	None	Certificate to export	Exported Certificate file	None	Administrator - TLS ECDSA public key: R - TLS RSA public key: R Resource Manager - TLS ECDSA public key: R - TLS RSA public key: R Certificate Manager - TLS RSA public key: R - TLS ECDSA public key: R
Create ssh-keyswap	Utility service create ssh keys	Service Indicator: Approved	SSH key to create	Confirmation of SSH key creation	Key pair generation	Administrator - SSH ECDSA private key: G - SSH ECDSA public key: G Resource Manager - SSH ECDSA private key: G - SSH

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						ECDSA public key: G
Configure Firewall	Set policy rules, and address lists for use by firewall rules	None	Policy rules, address lists	Confirmation of policy configuration	None	Administrator Firewall Manager Resource Manager
Configure Firewall Users	Set firewall user	None	Firewall user and configuration information	Confirmation of configuration	None	Administrator Firewall Manager User Manager
Show firewall state	Display the current system-wide state of firewall rules	None	N/A	Display the current system wide state of the firewall rules.	None	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
Shows statistics	Shows statistics of firewall rules on the BIG-IP system	None	N/A	List of statistics of firewall rules	None	Administrator Auditor Certificate Manager Firewall Manager iRule Manager Operator Resource Manager User Manager
View System Audit Log	Display logs/files of configuration changes	None	N/A	Display of system audit logs	None	Administrator Auditor iRule Manager

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Resource Manager
Export Analytics Logs System	Export Analytics Logs System	None	N/A	Display System Analytics Logs	None	Administrator User Manager
Enable/ Disable Audit	Enable/ Disable Audit	None	N/A	Confirmation of enabling or disabling of audit	None	Administrator Resource Manager
Configure Boot Options	Enable Quiet boot, Manage boot locations	None	Boot options	Confirmation of configuration of boot options	None	Administrator Resource Manager
Configure SSH access options	Enable / Disable SSH access, Configure IP address allow list	None	SSH access / IP address list	Confirmation of configuration of SSH access options	None	Administrator Resource Manager
Configure SSH user configuration	Update ssh/ authorized_keys file for user authentication	None	ssh/ authorized_keys file	Confirmation of configuration of SSH user configuration	None	Administrator - SSH ECDSA public key: W
Modify nodes and pool members	Enable / Disable nodes and pool members	None	Which nodes and pool members to modify	Confirmation of modification of nodes and pool members	None	Administrator Resource Manager
Configure nodes	Create, delete nodes	None	List of nodes to create / view / delete	Confirmation of creation / deletion of nodes	None	Administrator Resource Manager
Configure iRules	Create, delete, iRules	None	List of iRules to create / delete	Confirmation of creation / deletion of iRules	None	Administrator iRule Manager Resource Manager

Table 14: Approved Services

For the above table, the convention below applies when specifying the access permissions (types) that the service has for each SSP.

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.

4.4 Non-Approved Services

Name	Description	Algorithms	Role
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	CO / User
SSLO Configuration and usage	Management of the module protected by iApplx authentication	TLS ciphersuites implemented by f5-rest-node	CO / User
iControl REST access	Access to the system through REST API	RSA keypair with 2048, 3072 and 4096 (REST API)	CO / User
IPsec /IKEv2	Protocol configuration	HMAC-SHA2-256, HMAC-SHA2-512, AES-GCM Diffie-Hellman using MODP1024, MODP2048 groups Triple-DES, AES-GCM-128, AES-192, AES-256	CO / User
Simple network management protocol (SNMP)	Protocol configuration	SHA-1, AES-ECB, RSA- signature verification	CO / User
Establish TLS session	Signature generation and verification, Key Exchange	PKCS #1 v1.5 scheme with modulus other than 2048, 3072 or 4096 bits, for all SHA sizes; PKCS #1 v1.5 and PSS schema with modulus size 2048, 3072, 4096 bits with SHA-1, SHA2-224, SHA2-512; with ANS X9.31 standard; ECDSA with curves P-256/ P-384 with SHA-1/SHA2-224, ECDSA using curves other than P-256 and P-384 with all SHA sizes RSA with modulus sizes up to 16384 bits EdDSA with Ed25519 MD5/ SHA-1/ SHA2-224 / SHA2-512 EC Diffie-Hellman Ephemeral Unified with curves other than P-256, P-384. EC Diffie-Hellman using onePassDH / StaticUnified schemes. Diffie-Hellman using groups other than ffdhe2048, ffdhe3072, ffdhe4096	CO / User

Table 15: Non-Approved Services

5 Software/Firmware Security

5.1 Integrity Techniques

The integrity of the module is using the approved integrity technique HMAC-SHA-384. The HMAC key used for integrity check is 384 bits in length and is stored within the module.

Integrity tests are performed as part of the Pre-Operational Self-Tests.

5.2 Initiate on Demand

The on demand integrity test is performed as part of the Pre-Operational Self-Tests by rebooting 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 17.1.0.1. The module is a firmware validated at a Security Level 2 in Physical Security. Once the module is operational, it does not allow the loading of any additional firmware.

There are no further requirements for this security area.

Type of Operational Environment: Non-Modifiable

How Requirements are Satisfied: Once the module is operational, it does not allow the loading of any additional firmware.

7 Physical Security

7.1 Mechanisms and Actions Required

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

Table 16: Mechanisms and Actions Required

7.2 User Placed Tamper Seals

Number:

Hardware Appliance	# of Tamper Labels
r4800	5
r5900	4
r5920-DF	5
r10900 r10920-DF r12900-DS	5
VELOS CX410-BX110	1

Placement: The pictures below show the location of all tamper evident labels for each 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 / 4 tamper labels)

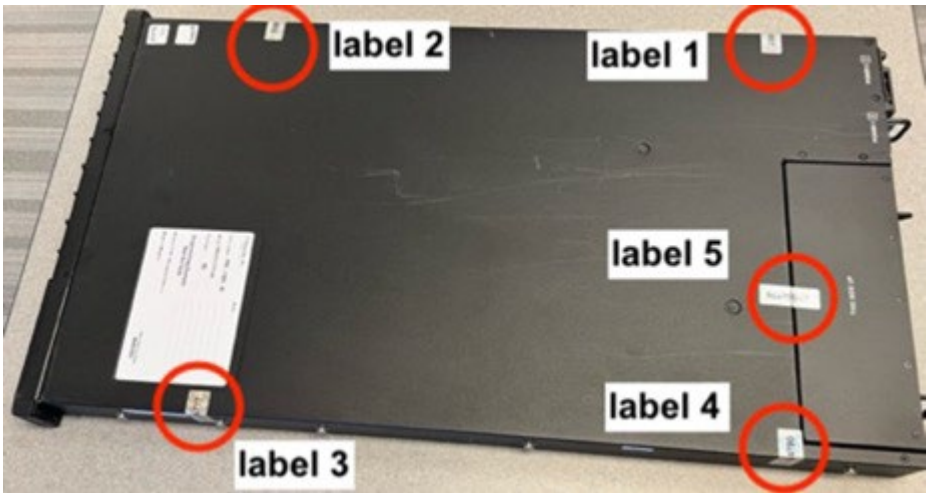

Figure 9 – Tamper labels on r5920-DF.
(5 of 5 tamper labels). Labels are located on the lateral sides of the platform -labels 1,2,3 and 4. The tamper label 5 on the chassis / enclosure lid is covering the ventilation fan tray that allows access to SSD.

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.

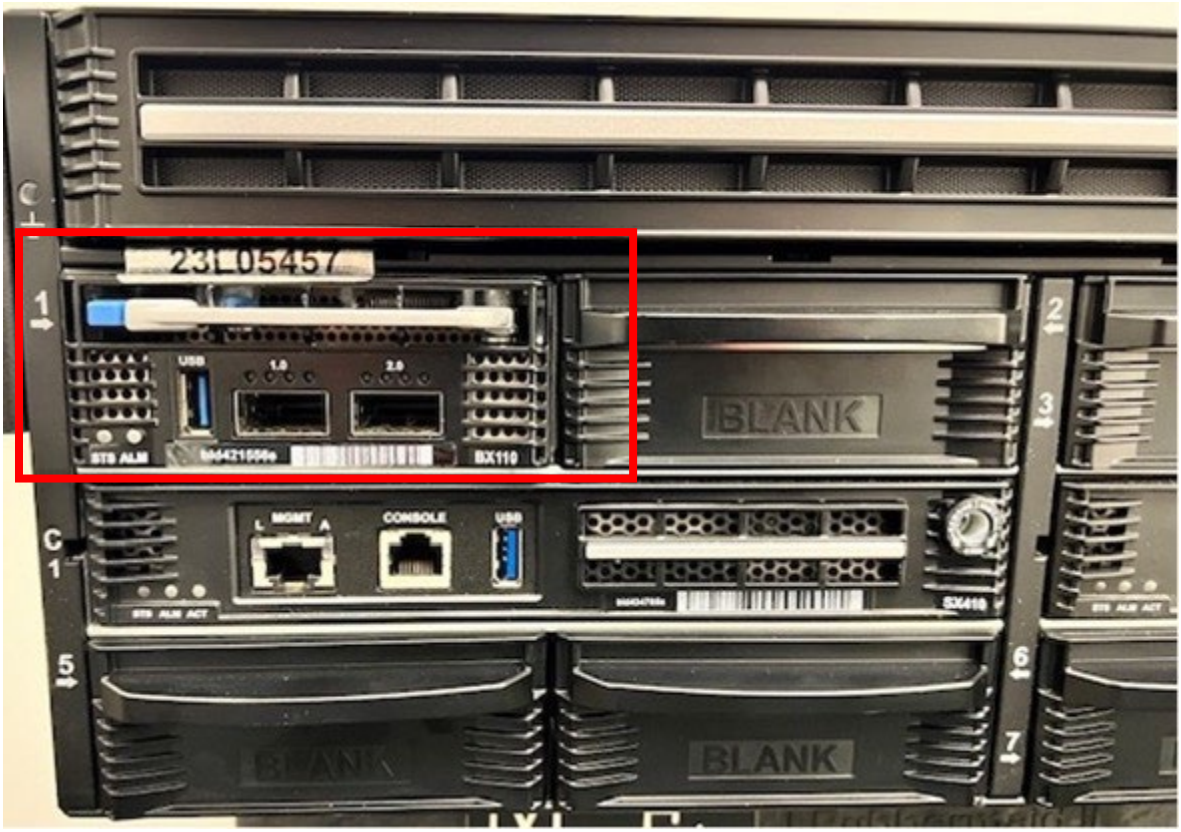

Figure 11 – Tamper labels on VELOS CX410-BX110 blade mounted on chassis.
The tamper label affixed between blade and chassis is positioned to provide evidence if the tested blade in slot #1 is removed from the chassis.

Surface Preparation: Before the module is installed in the production environment, tamper-evident labels must be installed in the location identified for each test platforms below. The tamper evident labels shall be installed for the module to operate in approved mode of operation. The following steps should be taken when installing or replacing the tamper evident labels on the module. The instructions are also included in *F5 Platforms: FIPS Kit Installation* provided with each hardware platform.

- 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 24 hours.

Operator Responsible for Securing Unused Seals: The Crypto Officer shall be responsible for the storage of the label kits.

Part Numbers: P/N: F5-ADD-BIG-FIPS140

7.3 Filler Panels

Hardware Appliance	# of Filler Panels
r4800	1 (blank in PSU slot)

r5900	1 (blank in PSU slot)
r5920-DF	1 (blank in PSU slot)
r10900 r10920-DF r12900-DS	0
VELOS CX410-BX110	7 blanks in blade slot #2-8

8 Non-Invasive Security

Per IG 12.A: Until requirements of SP 800-140F are defined, non-invasive mechanisms fall under ISO / IEC 19790:2012 Section 7.12 Mitigation of other attacks.

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
RAM	The keys are stored in plaintext form and are only accessible to the authenticated operator, to which the SSPs are associated	Dynamic
SSD	The keys stored in plaintext will remain on the system across power cycle and are only accessible to the authenticated operator to which the SSPs are associated	Static

Table 17: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Public key output during protocol handshake	Module	User	Plaintext	Automated	Electronic	
Public key input during protocol handshake	User	Module	Plaintext	Automated	Electronic	
Key Wrapping/Unwrapping with encryption and authentication in TLS	User	Module	Encrypted	Automated	Electronic	Key Wrapping/Unwrapping with encryption and authentication in TLS
Key Wrapping/Unwrapping with encryption and authentication in SSH	User	Module	Encrypted	Automated	Electronic	Key Wrapping/Unwrapping with encryption and authentication in SSH
Key Wrapping/Unwrapping with authenticated encryption	User	Module	Encrypted	Automated	Electronic	Key Wrapping/Unwrapping with authenticated encryption

Table 18: SSP Input-Output Methods

The module only allows entry/output of public keys in plaintext from outside of the module's TOEPP as part of protocol handshake process.
Once TLS/SSH session is established any key or data transfer performed thereafter is protected by authenticated encryption provided by the respective protocol

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Zeroization	Single pass zeroization erasing the entire module	All SSPs present in the module are erased including the one in the non-volatile memory	The Crypto Officer is calling the Zeroization service

Zeroization Method	Description	Rationale	Operator Initiation
Reboot System	Clear the SSPs present in RAM memory	Volatile memory used by the module is overwritten within nanoseconds when power is removed.	The Crypto Officer is calling Reboot System service
Delete SSH keyswap	Destruction of the selected SSH authentication keys	Zeroization service overwrites the memory occupied by keys with "zeros" or pre-defined values.	The Administrator or Resource Manager are calling the Delete SSH keyswap service
Closing TLS Connection / SSH Session	Zeroization of temporary values	SSP temporary values generated during key generation services are zeroized by the module	Closing TLS Connection / SSH Session

Table 19: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
TLS RSA private key	RSA private key used for RSA signature generation in TLS protocol	Modulus N: 2048 and 4096-bit keys - 112 and 150-bits	Asymmetric - CSP	Key pair generation		Signature generation
TLS RSA public key	RSA public key used for RSA signature verification in TLS protocol	Modulus N: 2048 and 4096-bit keys - 112 and 150-bits	Asymmetric - PSP	Key pair generation		Signature verification
TLS ECDSA private key	ECDSA private key used for EC signature generation, shared secret computation in TLS protocol	Curve size: P-256, P-384 - 128 and 192-bits	Asymmetric - CSP	Key pair generation		Signature generation
TLS ECDSA public key	ECDSA public key used for EC signature verification, shared secret computation in TLS protocol	Curve size:P-256, P-384 - 128 and 192-bits	Asymmetric - PSP	Key pair generation		Signature verification Key pair verification
TLS EC Diffie-Hellman private key	TLS EC Diffie-Hellman private key used for EC signature generation, shared secret	P-256, P-384 - 128 and 192-bits	Asymmetric - CSP	Key pair generation		TLS Handshake (ECC)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	computation in TLS protocol					
TLS EC Diffie-Hellman public key	EC Diffie-Hellman public key used for signature verification, shared secret computation in TLS protocol	Curve size: P-256, P-384 - 128 and 192-bits	Asymmetric - PSP	Key pair generation		Key pair verification TLS Handshake (ECC)
TLS Diffie-Hellman public key	TLS Diffie-Hellman private key used for shared secret computation in TLS protocol	Curve size: ffdhe2048, ffdhe3072, ffdhe4096 - 112 to 150-bits	Asymmetric - PSP	Key pair generation		TLS Handshake (FFC)
TLS Diffie-Hellman private key	TLS Diffie-Hellman public key used for shared secret computation in TLS protocol	Curve size: ffdhe2048, ffdhe3072, ffdhe4096 - 112 to 150-bits	Asymmetric - CSP	Key pair generation		TLS Handshake (FFC)
TLS pre-primary secret	TLS pre-primary secret used for deriving the TLS primary secret	Curve size: P-256, P-384 - 128 and 192-bits	Asymmetric - CSP		TLS Handshake (ECC) TLS Handshake (FFC)	TLS Handshake (ECC) TLS Handshake (FFC)
TLS primary secret	TLS primary secret used for deriving the TLS derived key	256-bits - 256-bits	Pre-primary secret - CSP	TLS Handshake (ECC) TLS Handshake (FFC)		TLS Handshake (ECC) TLS Handshake (FFC)
TLS derived session key	TLS derived session key from TLS primary secret	Key length: 128 and 256-bits (AES) 112 and 256-bits (HMAC) - 128 and 256-bits (AES) 112 and 256-bits (HMAC)	Symmetric Key - CSP	TLS Handshake (ECC) TLS Handshake (FFC)		TLS Handshake (ECC) TLS Handshake (FFC)
SSH shared secret	SSH shared secret used for deriving the SSH key	Curve size: P-256, P-384 - 128 and 192-bits	Symmetric Key - CSP		SSH Handshake	SSH Handshake
SSH derived	SSH derived session key	Key Length: 128 and 256-bits (AES) 112 and 256-bits (HMAC) -	Shared secret - CSP		SSH Handshake	SSH Handshake

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
session key		128 and 256-bits (AES) 112 and 256-bits (HMAC)				
Entropy input string	Entropy input string used to seed the DRBG	384 bits - 384 bits	Random number generation - CSP	Random Number Generation in Control Plane Random Number Generation in Data Plane		Random Number Generation in Control Plane Random Number Generation in Data Plane
DRBG seed	DRBG seed derived from entropy input as defined in SP 800-90Ar1	384 bits - 384 bits	Random number generation - CSP	Random Number Generation in Control Plane Random Number Generation in Data Plane		Random Number Generation in Control Plane Random Number Generation in Data Plane
DRBG internal state (V and key values)	Internal state of CTR_DRBG	384 bits - 384 bits	Random number generation - CSP	Random Number Generation in Control Plane Random Number Generation in Data Plane		Random Number Generation in Control Plane Random Number Generation in Data Plane
SSH ECDSA private key	Used for key-based authentication in SSH protocol	Curve size: P-256, P-384 - 128 and 192-bits	Asymmetric - CSP	Key pair generation		Signature generation
SSH ECDSA public key	Used for key-based authentication in SSH protocol	Curve size:P-256, P-384 - 128 and 192-bits	Asymmetric - PSP	Key pair generation		Signature verification
SSH EC Diffie-Hellman private key	Used for key exchange in SSH protocol	Curve size: P-256, P-384 - 128 and 192-bits	Asymmetric - CSP	Key pair generation		SSH Handshake
SSH EC Diffie-	Used for key exchange in SSH protocol	Curve size:P-256, P-384 - 128 and 192-bits	Asymmetric - PSP	Key pair generation		Key pair verification

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Hellman public key						SSH Handshake
Password	Password input by the User or CO during creation of a new user or updating an existing password	8 characters - 1/676,000,000	Password - CSP			

Table 20: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
TLS RSA private key		SSD:Plaintext	From handle creation until freeing the cipher handle	Zeroization	TLS RSA public key:Paired With
TLS RSA public key	Public key input during protocol handshake Public key output during protocol handshake	SSD:Plaintext	From handle creation until freeing the cipher handle	Zeroization	TLS RSA private key:Paired With
TLS ECDSA private key		SSD:Plaintext	From handle creation until freeing the cipher handle	Zeroization	TLS ECDSA public key:Paired With
TLS ECDSA public key	Public key input during protocol handshake Public key output during protocol handshake	SSD:Plaintext	From handle creation until freeing the cipher handle	Zeroization	TLS ECDSA private key:Paired With
TLS EC Diffie-Hellman private key		RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	TLS EC Diffie-Hellman public key:Paired With
TLS EC Diffie-Hellman public key	Public key input during protocol handshake Public key output during protocol handshake	RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	TLS EC Diffie-Hellman private key:Paired With
TLS Diffie-Hellman public key	Public key output during protocol handshake Public key input during protocol handshake	SSD:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	
TLS Diffie-Hellman private key		SSD:Plaintext	From handle creation until	Reboot System Closing TLS	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
			freeing the cipher handle	Connection / SSH Session	
TLS pre-primary secret		RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	TLS primary secret:Used With
TLS primary secret		RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	TLS pre-primary secret :Derived From
TLS derived session key	Public key input during protocol handshake	RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	TLS primary secret:Derived From
SSH shared secret		RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	SSH derived session key :Used With
SSH derived session key	Public key input during protocol handshake	RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	SSH shared secret:Derived From
Entropy input string		RAM:Plaintext	Storage duration during the usage of the CSP	Reboot System	DRBG seed :Used With
DRBG seed		RAM:Plaintext	Storage duration during the usage of the CSP	Reboot System	DRBG internal state (V and key values) :Used With
DRBG internal state (V and key values)		RAM:Plaintext	Storage duration during the usage of the CSP	Reboot System	DRBG seed :Used With
SSH ECDSA private key		SSD:Plaintext	Persistent	Zeroization Delete SSH keyswap	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
SSH ECDSA public key	Public key input during protocol handshake	SSD:Plaintext	From handle creation until freeing the cipher handle	Zeroization Delete SSH keyswap	
SSH EC Diffie-Hellman private key		RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	
SSH EC Diffie-Hellman public key	Public key output during protocol handshake Public key input during protocol handshake	RAM:Plaintext	From handle creation until freeing the cipher handle	Reboot System Closing TLS Connection / SSH Session	
Password	Key Wrapping/Unwrapping with encryption and authentication in TLS Key Wrapping/Unwrapping with encryption and authentication in SSH Key Wrapping/Unwrapping with authenticated encryption	RAM:Plaintext	From handle creation until freeing the cipher handle	Closing TLS Connection / SSH Session	

Table 21: SSP Table 2

10 Self-Tests

At power-up the module performs the pre-operational self-tests (the integrity test) and the conditional cryptographic algorithm self-tests (CASTs). Both the pre-operational tests and 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. Services are not available during the pre-operational self-test and CASTs and the data output interface is inhibited.

If the module fails any of the tests, 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.

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
HMAC-SHA2-384 (A3729)	HMAC key: 384-bits	Message Authentication	SW/FW Integrity	Module becomes operational	Integrity of the module is verified by comparing the HMAC-SHA2-384 value calculated at runtime with the HMAC-SHA2-384 value stored in the module that was computed at build time
HMAC-SHA2-384 (A3730)	HMAC key: 384-bits	Message Authentication	SW/FW Integrity	Module becomes operational	Integrity of the module is verified by comparing the HMAC-SHA2-384 value calculated at runtime with the HMAC-SHA2-384 value stored in the module that was computed at build time

Table 22: Pre-Operational Self-Tests

The pre-operational self-tests are performed automatically when the module is powered on.

Services are not available during the pre-operational self-test and the data output interface is inhibited. On successful completion of the pre-operational self-tests, the module enters operational mode and cryptographic services are available.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
Counter DRBG (A3729)	AES-256 in CTR mode, with and without derivation function, prediction resistance disabled	KAT	CAST	Module becomes operational	SP800-90ARev1 section 11.3 health tests	Test runs at power on
AES-CBC (A3729)	128-bit key	KAT	CAST	Module becomes operational	Encryption / decryption	Test runs at power on
AES-GCM (A3729)	128-bit key	KAT	CAST	Module becomes operational	Encryption / decryption	Test runs at power on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
RSA SigGen (FIPS186-5) (A3729)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
RSA SigGen (FIPS186-4) (A3729)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
RSA SigVer (FIPS186-5) (A3729)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
RSA SigVer (FIPS186-4) (A3729)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
RSA KeyGen (FIPS186-5) (A3729)	Requested modulus size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
RSA KeyGen (FIPS186-4) (A3729)	Requested modulus size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
ECDSA SigGen (FIPS186-5) (A3729)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
ECDSA SigGen (FIPS186-4) (A3729)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
ECDSA KeyGen (FIPS186-5) (A3729)	Requested curve size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
ECDSA KeyGen (FIPS186-4) (A3729)	Requested curve size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
KAS-ECC-SSC Sp800-56Ar3 (A3729)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power on
HMAC-SHA-1 (A3729)	HMAC-SHA-1	KAT	CAST	Module becomes operational	MAC	Test runs at power on
HMAC-SHA2-256 (A3729)	HMAC-SHA2-256	KAT	CAST	Module becomes operational	MAC	Test runs at power on

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
TLS v1.2 KDF RFC7627 (A3729)	SHA-256	KAT	CAST	Module becomes operational	Key derivation used in the TLS protocol	Test runs at power on
KDF SSH (A3729)	SHA-256	KAT	CAST	Module becomes operational	Key derivation used in the SSH protocol	Test runs at power on
HMAC-SHA2-384 (A3730)	HMAC-SHA-384	KAT	CAST	Module becomes operational	MAC	Test runs at power on
AES-CBC (A3730)	128-bit key	KAT	CAST	Module becomes operational	Encryption / decryption	Test runs at power on
AES-GCM (A3730)	128-bit key	KAT	CAST	Module becomes operational	Encryption / decryption	Test runs at power on
KAS-ECC-SSC Sp800-56Ar3 (A3730)	P-256	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power on
ECDSA SigGen (FIPS186-5) (A3730)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
ECDSA SigGen (FIPS186-4) (A3730)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
Safe Primes Key Generation (A3730)	Requested curve size	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
KAS-FFC-SSC Sp800-56Ar3 (A3730)	fdhe2048	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power on
HMAC-SHA2-256 (A3730)	HMAC-SHA-384	KAT	CAST	Module becomes operational	MAC	Test runs at power on
HMAC-SHA2-384 (A3729)	HMAC-SHA-384	KAT	CAST	Module becomes operational	MAC	Test runs at power on
ECDSA KeyGen (FIPS186-5) (A3730)	Requested curve size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
ECDSA KeyGen (FIPS186-4) (A3730)	Requested curve size, SHA2-256	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
ECDSA SigVer (FIPS186-5) (A3730)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
ECDSA SigVer (FIPS186-4) (A3730)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
ECDSA SigVer (FIPS186-5) (A3729)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
ECDSA SigVer (FIPS186-4) (A3729)	P-256 and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
KAS-FFC-SSC Sp800-56Ar3 (A3729)	ffdhe2048	KAT	CAST	Module becomes operational	Shared secret computation	Test runs at power on
Safe Primes Key Generation (A3729)	Requested curve size	PCT	PCT	Asymmetric algorithm is performed	Calculation and verification of a digital signature	Key generation
TLS v1.2 KDF RFC7627 (A3730)	SHA-256	KAT	CAST	Module becomes operational	Key derivation used in the TLS protocol	Test runs at power on
Counter DRBG (A3730)	AES-256 in CTR mode, with derivation function, prediction resistance disabled	KAT	CAST	Module becomes operational	SP800-90ARev1 section 11.3 health tests	Test runs at power on
RSA SigGen (FIPS186-5) (A3730)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
RSA SigGen (FIPS186-4) (A3730)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature generation	Test runs at power on
RSA SigVer (FIPS186-5) (A3730)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on
RSA SigVer (FIPS186-4) (A3730)	2048 bit key and SHA2-256	KAT	CAST	Module becomes operational	Signature verification	Test runs at power on

Table 23: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
HMAC-SHA2-384 (A3729)	Message Authentication	SW/FW Integrity	Determined by the operator	Module is powered-off and on
HMAC-SHA2-384 (A3730)	Message Authentication	SW/FW Integrity	Determined by the operator	Module is powered-off and on

Table 24: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Counter DRBG (A3729)	KAT	CAST	On Demand	Manually
AES-CBC (A3729)	KAT	CAST	On Demand	Manually
AES-GCM (A3729)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-5) (A3729)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3729)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-5) (A3729)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3729)	KAT	CAST	On Demand	Manually
RSA KeyGen (FIPS186-5) (A3729)	PCT	PCT	On Demand	Manually
RSA KeyGen (FIPS186-4) (A3729)	PCT	PCT	On Demand	Manually
ECDSA SigGen (FIPS186-5) (A3729)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3729)	KAT	CAST	On Demand	Manually
ECDSA KeyGen (FIPS186-5) (A3729)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3729)	PCT	PCT	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
KAS-ECC-SSC Sp800-56Ar3 (A3729)	KAT	CAST	On Demand	Manually
HMAC-SHA-1 (A3729)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3729)	KAT	CAST	On Demand	Manually
TLS v1.2 KDF RFC7627 (A3729)	KAT	CAST	On Demand	Manually
KDF SSH (A3729)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3730)	KAT	CAST	On Demand	Manually
AES-CBC (A3730)	KAT	CAST	On Demand	Manually
AES-GCM (A3730)	KAT	CAST	On Demand	Manually
KAS-ECC-SSC Sp800-56Ar3 (A3730)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-5) (A3730)	KAT	CAST	On Demand	Manually
ECDSA SigGen (FIPS186-4) (A3730)	KAT	CAST	On Demand	Manually
Safe Primes Key Generation (A3730)	PCT	PCT	On Demand	Manually
KAS-FFC-SSC Sp800-56Ar3 (A3730)	KAT	CAST	On Demand	Manually
HMAC-SHA2-256 (A3730)	KAT	CAST	On Demand	Manually
HMAC-SHA2-384 (A3729)	KAT	CAST	On Demand	Manually
ECDSA KeyGen (FIPS186-5) (A3730)	PCT	PCT	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A3730)	PCT	PCT	On Demand	Manually
ECDSA SigVer (FIPS186-5) (A3730)	KAT	CAST	On Demand	Manually

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
ECDSA SigVer (FIPS186-4) (A3730)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-5) (A3729)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A3729)	KAT	CAST	On Demand	Manually
KAS-FFC-SSC Sp800-56Ar3 (A3729)	KAT	CAST	On Demand	Manually
Safe Primes Key Generation (A3729)	PCT	PCT	On Demand	Manually
TLS v1.2 KDF RFC7627 (A3730)	KAT	CAST	On Demand	Manually
Counter DRBG (A3730)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-5) (A3730)	KAT	CAST	On Demand	Manually
RSA SigGen (FIPS186-4) (A3730)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-5) (A3730)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-4) (A3730)	KAT	CAST	On Demand	Manually

Table 25: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Halt Error	Module is no longer operational. The data output is inhibited.	HMAC-SHA2-384 KAT failure or HMAC-SHA2-384 integrity test failure Failure of any of the CASTs	The module must be re-loaded	For integrity, CAST and health test failures the module will not load. For PCT failures the module will halt and enter the error state

Table 26: Error States

The module must be re-loaded with a fresh image to clear the error condition.

10.5 Operator Initiation of Self-Tests

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

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

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 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 device itself.

Installation Process: Follow the instructions in the "*BIG-IP System: 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. Installing the FIPS license for the host system is required for module activation. Guidance on Licensing the BIG-IP system can be found in https://support.f5.com/csp/article/K7752 and summarized as followed: Before you can activate the license for the BIG-IP system, you must obtain a base registration key. The base registration key is pre-installed on new BIG-IP systems. When you power up the product and connect to the Configuration utility, the Licensing page opens and displays the registration key. After a license activation method is selected (activation method specifies how you want the system 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. If the automated activation method is selected, the BIG-IP system automatically connects to the F5 License Server and activates the license. If the manual method is selected, the Crypto Officer shall go to the F5 Product Licensing page at secure.f5.com, paste the dossier in the “Enter Your Dossier” box which produces a license. The Crypto Officer will then copy and paste it into the “License” box in the Configuration Utility. The BIG-IP system then reloads the configuration and is ready for additional system configuration.
- Once the device is installed, licensed and configured, the Crypto Officer should confirm that the system is installed and licensed correctly as follows:

  Version Confirmation: The Crypto Officer should call the show version service (with command "tmsh show sys version" and "tmsh show sys hardware"), then confirm that the provided version matches the validated version shown in Table - Tested Module Identification. 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.

  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”.

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 appliance'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 hardware.
- 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 should be turned ON for the platform GUI.
- The server ssl profile shall be configured with "cert none" and "key none" option that disables client authentication.

11.2 Administrator Guidance

The Crypto Officer should confirm that version, license are provided according to the documentation in section 11.1. The Crypto Officer should follow the additional guidance in section 11.1 to operate the module in the approved 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

11.3 Non-Administrator Guidance

N/A

11.4 Design and Rules

- AES GCM IV see Section 2.7
- SP800-56Ar3 assurances see Section 2.7
- RSA modulus size see Section 2.7

11.5 End of Life

Secure sanitization of the module consists of using the Zeroization service that will perform erasing all the SSPs bringing the module to the factory default state. The service can only be triggered by the administrator using “tmsh load sys config default” command via CLI and must be followed by reboot of the module.

12 Mitigation of Other Attacks

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

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CMVP	Cryptographic Module Validation Program
CSP	Critical Security Parameter
CTR	Counter Mode
DES	Data Encryption Standard
DF	Derivation Function
DSA	Digital Signature Algorithm
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
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
MAC	Message Authentication Code
NDF	No Derivation Function
NIST	National Institute of Science and Technology
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PCT	Pairwise Consistency Test
PR	Prediction Resistance
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSH	Secure Shell
TDES	Triple-DES
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)** August 2015 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
FIPS186-5	**Digital Signature Standard (DSS)** February 2023 https://doi.org/10.6028/NIST.FIPS.186-5
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
FIPS202	**SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions** August 2015 http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf
PKCS#1	**Public Key Cryptography Standards (PKCS) #1: RSA Cryptography** Specifications Version 2.1 February 2003 http://www.ietf.org/rfc/rfc3447.txt
RFC3394	**Advanced Encryption Standard (AES) Key Wrap Algorithm** September 2002 http://www.ietf.org/rfc/rfc3394.txt
RFC5649	**Advanced Encryption Standard (AES) Key Wrap with Padding Algorithm** September 2009 http://www.ietf.org/rfc/rfc5649.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
SP800-38E	**NIST Special Publication 800-38E - Recommendation for Block Cipher Modes of Operation: The XTS AES Mode for Confidentiality on Storage Devices** January 2010 http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.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
SP800-56Ar3	**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
SP800-56Cr2	**Recommendation for Key Derivation through Extraction-then-Expansion** August 2020 https://doi.org/10.6028/NIST.SP.800-56Cr2
SP800-57	**NIST Special Publication 800-57 Part 1 Revision 4 - Recommendation for Key Management Part 1: General** January 2016 http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf
SP800-67	**NIST Special Publication 800-67 Revision 1 - Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher** January 2012 http://csrc.nist.gov/publications/nistpubs/800-67-Rev1/SP-800-67-Rev1.pdf
SP800-90Ar1	**NIST Special Publication 800-90A - Revision 1 - Recommendation for Random Number Generation Using Deterministic Random Bit Generators** June 2015 http://dx.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
SP800-131r2	**Transitioning the Use of Cryptographic Algorithms and Key Lengths** March 2019 https://doi.org/10.6028/NIST.SP.800-131Ar2

SP800-132	**NIST Special Publication 800-132 - Recommendation for Password-Based Key Derivation - Part 1: Storage Applications** December 2010 http://csrc.nist.gov/publications/nistpubs/800-132/nist-sp800-132.pdf
SP800-133r2	**NIST Special Publication 800-133 - 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
SP800-140Br1	**NIST Special Publication 800-140B - CMVP Security Policy Requirements** November 2023 https://doi.org/10.6028/NIST.SP.800-140Br1