Corporation

SR-OS Cryptographic Module (SRCM)

FIPS 140-2 Non-Proprietary Security Policy

FIPS Security Level:1

Document Version: 3.4

June 16, 2022

TABLE OF CONTENTS

LIST OF FIGURES

GLOSSARY

AES-128, AES-256	*Advanced Encryption Standard*
BGP	*Border Gateway Protocol*
CBC	*Cipher Block Chaining*
CFM	*Control / Forwarding Module*
CLI	*Command Line Interface*
CMVP	*Cryptographic Module Validation Program*
CPM	*Control Processor Module*
CSP	*Critical Security Parameter*
CVL	*Component Validation List*
ESP	*Encapsulating Security Payload*
FIPS	*Federal Information Processing Standard*
GRE	*Generic Routing Encapsulation*
HMAC	*Hashed Message Authentication Code*
ICMP	*Internet Control Message Protocol*
ICV	*Integrity Check Value*
IGMP	*Internet Group Management Protocol*
IP	*Internet Protocol*
IPsec	*IP Security*
LDP	*Label Distribution Protocol*
LSP	*Label Switched Path*
MPLS	*Multi-protocol label switching*
NDRNG	*Non-Deterministic RNG*
NIST	*National Institute of Standards and Technology*
OSPF	*Open Shortest Path First*
PFS	*Perfect Forward Secrecy*
RNG	*Random Number Generator*
SA	*Security Association*
SAM	*Service Aware Manager*
SFM	*Switch Fabric Module*
SHA	*Secure Hash Algorithm*
SSH	*Secure Shell*
SPI	*Security Parameter Index*
TLS	*Transport Layer Security*

TM	*Traffic Management*
VPLS	*Virtual Private LAN Service*

Table 1 - Glossary

1. INTRODUCTION

1.1 Purpose

This document describes the non-proprietary SR-OS (Service Router Operating System) Cryptographic Module (SRCM) Security Policy for the 7950 XRS, 7750 SR and the 7450 ESS product families. These are referenced in the document as either 7x50 or XRS/SR/ESS.

This security policy provides the details for configuring and running the 7x50 products in a FIPS-140-2 mode of operation and describes how the module meets the Level 1 requirements of FIPS 140-2. Please see the references section for a full list of FIPS 140-2 requirements.

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

Table 2 - Security Level per FIPS 140-2 Section

1.2 Versions Available for FIPS

The following platforms and models of the 7x50 products are FIPS tested and validated for running SRCM in a FIPS approved mode:

Platform	Model(s)
7750 Service Router (SR)	SR-7, SR-14s, SR-7s, SR-2s, SR-1s, SR-1, SR-2e, SR-a4
7950 Extensible Routing System (XRS)	XRS-20, XRS-16c

Table 3 – FIPS Validated Platforms and Models

The following platforms and models are vendor affirmed for FIPS compatibility:

Platform	Model(s)
7750 Service Router (SR)	SR-12e, SR-12, SR-3e, SR-1e, SR-a8
7450 Ethernet Services Switch (ESS)	ESS-12, ESS-7

Table 4 – Vendor Affirmed Platforms and Models

2. SR-OS CRYPTOGRAPHIC MODULE OVERVIEW

The section provides an overview of the SR-OS Cryptographic Module (SRCM) and the FIPS validated cryptographic algorithms used by services requiring those algorithms. The SRCM does not implement any services or protocols directly. Instead, it provides the cryptographic algorithm functions needed to allow SR-OS to implement cryptography for those services and protocols that require it.

2.1 SRCM Characteristics

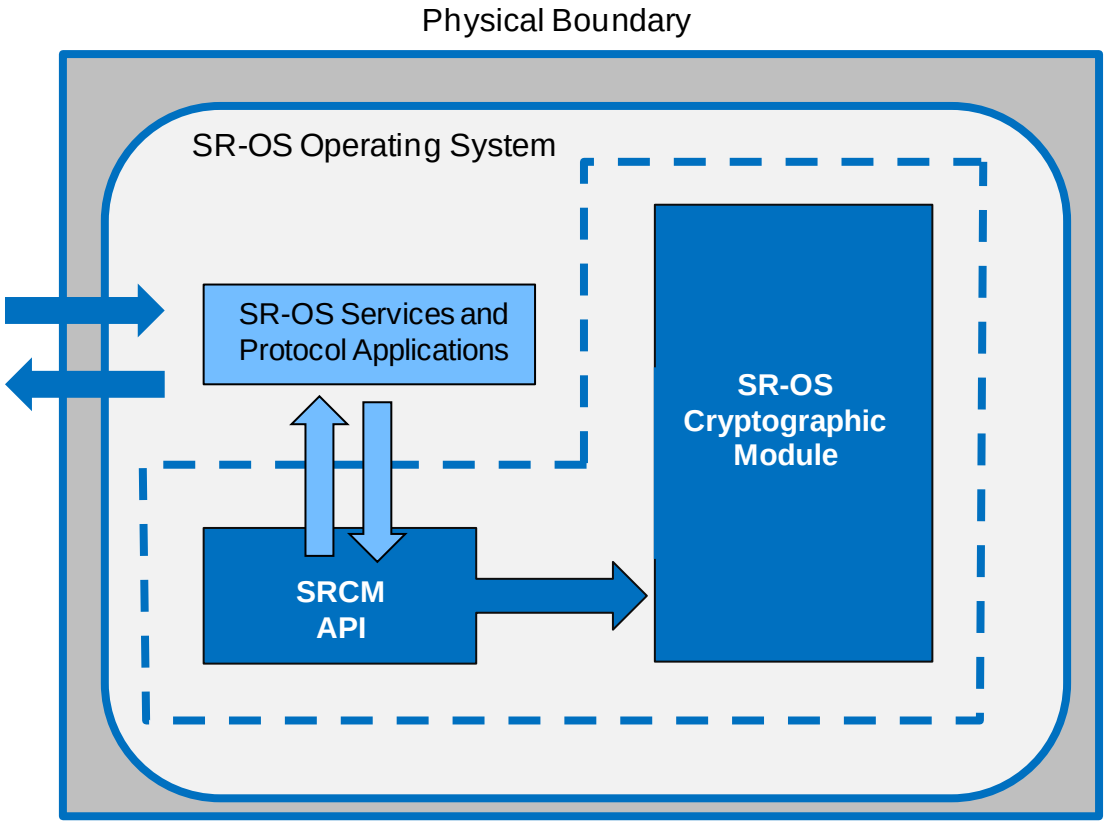

Figure 1 - SRCM Diagram of Logical and Physical Boundaries

The SRCM logical and physical properties and boundary considerations are illustrated in Figure 1. The solid blue line represents the physical boundary of the cryptographic module that represents the hardware system on which SR-OS is running and hence where SRCM is also running. The dashed blue line indicates the logical cryptographic boundary of the SRCM within SR-OS. The SRCM is available as a cryptographic service for any SR-OS services or protocols that require cryptographic operations.

The SRCM is part of a single SR-OS binary file (cpm.tim) that is used to run the full SR-OS application. SRCM is classified as a multi-chip standalone firmware module and SRCM is included within the SR-OS application code. The firmware version used to validate the SRCM is SR-OS 20.10.R4.

2.2 SRCM Approved Algorithms

The SRCM uses the following FIPS approved algorithms:

Algorithm	Certificate #20.10R4
AES CBC (e/d; 128, 192, 256); CFB128 (e/d;128); CTR (ext only; 128, 192, 256) CMAC; ECB* (e/d; 128); CCM* (e/d; 128); XTS* (e/d; 128, 256)	C2084, C2075, C2074
Triple-DES (TCBC) (keying option (K1 != K2 != K3))	C2084, C2075, C2074
RSA FIPS186-2: ANSI X9.31 1024-bit & 1536-bit& 2048-bit & 3072-bit & 4096-bit signature verification FIPS186-2: RSASSA PKCS v1.5 1024-bit & 1536-bit& 2048-bit & 3072-bit & 4096-bit Signature Verification FIPS186-2: RSASSA PSS 1024-bit & 1536-bit& 2048-bit & 3072-bit & 4096-bit Signature Verification FIPS186-4: ANSI X9.31 2048-bit & 3072-bit Signature Generation FIPS186-4: 2048-bit Key Pair Generation [FIPS186-4_Fixed_e (10001); PGM (ProbRandom: (2048) PPTT:(C.2)] ANSI X9.31 1024-bit, 2048-bit & 3072-bit Signature Verification RSASSA-PKCS#1-v1.5: 2048-bit & 3072-bit Signature Generation RSASSA-PSS: 2048-bit & 3072-bit Signature Generation RSASSA-PKCS#1-v1.5: 1024-bit, 2048-bit & 3072-bit Signature Verification RSASSA-PSS: 1024-bit, 2048-bit & 3072-bit Signature Verification	C2084, C2075, C2074
HMAC (HMAC-SHA-1, HMAC-SHA2-224, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512)	C2084, C2075, C2074
AES-GCM • **Direction: Decrypt, Encrypt** • **IV Generation: External** • **Key Length: 128, 192, 256** • **Tag Length: 32, 64, 96, 104, 112, 120, 128** • **IV Length: 8, 96, 1024** • **Payload Length: 8, 128, 256, 408** • **AAD Length: 0, 8, 128, 256, 408** **AES-GMAC**	C2084, C2075, C2074

• **Direction: Decrypt, Encrypt** • **IV Generation: External** • **Key Length: 128, 192, 256** • **Tag Length: 32, 64, 96, 104, 112, 120, 128** • **IV Length: 8, 96, 1024** • **AAD Length: 0, 8, 128, 256, 408** **AES-KW** • **Direction: Decrypt, Encrypt** • **Cipher: Cipher** • **Key Length: 128, 256** • **Payload Length: 128, 192, 448, 1024, 4096** **Prerequisites:** • **C2084 (AES)** **AES-KWP** • **Direction: Decrypt, Encrypt** • **Cipher: Cipher** • **Key Length: 128, 256** • **Payload Length: 4096** **Prerequisites:** • **C2084 (AES)** **KTS** • **Direction: Decrypt, Encrypt** • **Cipher: Cipher** • **Key Length: 128, 256** • **Payload Length: 128, 192, 448, 1024, 4096** **Prerequisites:** **C2084 (AES)** **Key establishment methodology provides 128 or 256 bits of encryption strength**	
CMAC AES-128 & AES-256 Generation ○ **Capabilities:** ▪ **Direction: Generation**	C2084, C2075, C2074

▪ **Key Length: 128** ▪ **MAC: 64, 96, 128** ▪ **Message Length: 0, 128, 320, 480, 512, 524288** ○ **Capabilities:** ▪ **Direction: Generation** ▪ **Key Length: 256** ▪ **MAC: 64, 96, 128** ▪ **Message Length: 0, 128, 320, 480, 512, 524288** ○ **Capabilities:** ▪ **Direction: Verification** ▪ **Key Length: 128** ▪ **MAC: 64, 96, 128** ▪ **Message Length: 0, 128, 320, 480, 512, 524288** ○ **Capabilities:** ▪ **Direction: Verification** ▪ **Key Length: 256** ▪ **MAC: 64, 96, 128** ▪ **Message Length: 0, 128, 320, 480, 512, 524288**	
SHA (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512)	C2084, C2075, C2074
DRBG CTR_DRBG (AES-256)	C2084, C2075, C2074
DSA FIPS186-4: 2048-bit & 3072-bit PQG Generation & Verification FIPS186-4: 2048-bit & 3072-bit Key Pair Generation [(2048,256); (3072,256)] 2048-bit & 3072-bit Signature Generation 1024-bit, 2048-bit & 3072-bit Signature Verification	C2084, C2075, C2074
ECDSA FIPS186-4: 186-4 Key Pair Generation (P-256, P-384, P-521) Public Key Validation CURVES (ALL-P ALL-K ALL-B) Signature Generation CURVES (P-256: (SHA-256, 384, 512) P-384: (SHA-256, 384, 512) P-521: (SHA-256, 384, 512))	C2084, C2075, C2074

Signature Verification CURVES(P-192: (SHA-1, 224, 256, 384, 512) P-224: (SHA-1, 224, 256, 384, 512) P-256: (SHA-1, 224, 256, 384, 512) P-384: (SHA-1, 224, 256, 384, 512) P-521: (SHA-1, 224, 256, 384, 512) K-163: (SHA-1, 224, 256, 384, 512) K-233: (SHA-1, 224, 256, 384, 512) K-283: (SHA-1, 224, 256, 384, 512) K-409: (SHA-1, 224, 256, 384, 512) K-571: (SHA-1, 224, 256, 384, 512) B-163: (SHA-1, 224, 256, 384, 512) B-233: (SHA-1, 224, 256, 384, 512) B-283: (SHA-1, 224, 256, 384, 512) B-409: (SHA-1, 224, 256, 384, 512) B-571: (SHA-1, 224, 256, 384, 512))	
CVL SSH KDF: Cipher: AES-128, AES-192, AES-256, TDES Hash Algorithm: SHA-1, SHA2-256, SHA2-512 TLS KDF: TLS Version: v1.2 Hash Algorithm: SHA2-256	C2084, C2075, C2074

Table 5 – Approved Algorithm Implementations

Note: Algorithms marked by asterisk (*) are implemented but not used by any services implemented by the module in Approved mode of operation.

2.3 SRCM non-Approved but Allowed Algorithms

The module supports the following non-FIPS approved algorithms which are:

- Diffie-Hellman (key agreement; key establishment methodology provides between 112 and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)
- NDRNG

2.4 SRCM Interfaces

The physical ports used by SRCM within SR-OS are the same as those available on the system which is running SR-OS per the platforms specified in the previous section. The logical interface is a C-language application program interface (API).

The Data Input interface consists of the input parameters of the API procedures and includes plaintext and/or cipher text data.

The Data Output interface consists of the output parameters of the API procedures and includes plaintext and/or cipher text data.

The Control Input interface consists of API functions that specify commands and control data used to control the operation of the module. The API may specify other functions or procedures as control input data.

The Status Output includes the return status, data and values associated with the status of the module.

The module provides logical interfaces to the other services within SR-OS and those other SR-OS services use the following logical interfaces for cryptographic functions: data input, data output, control input, and status output.

Interface	Description
Data Input	API input parameters including plaintext and/or cipher text data
Data Output	API output parameters including plaintext and/or cipher text data
Control Input	API procedure calls that may include other function calls as input, or input arguments that specify commands and control data used to control the operation of the module.
Status Output	API return code describing the status of SRCM

Table 6 – FIPS 140-2 Logical Interface Mappings

3. SRCM ROLES AND SERVICES

The SRCM meets all FIPS 140-2 level 1 requirements for Roles and Services, implementing support for both the Crypto Officer and User roles within the SRCM. The support for both Crypto Officer and User roles within the SRCM is classed as a process. As allowed by FIPS 140-2, the SRCM does not support user authentication for these roles, which is handled by the SR-OS system implementing SRCM. Only one role may be using the SRCM at a time and the module does not allow concurrent operators to access the SRCM.

The User and Crypto Officer roles are implicitly assumed by the entity accessing the services implemented by the SRCM:

- Installation and initialization of the SRCM which is embedded in the SR-OS image and installed on the SR-OS platforms is assumed implicitly as the Crypto Officer when installation and initialization occurs.

The services available by the SRC in FIPS mode to the Crypto Officer and User roles consist of the following:

Services	Access	Critical Security Parameters	Crypto Officer	User
Encryption	Execute	Symmetric keys AES, Triple-DES	X	X
Decryption	Execute	Symmetric keys AES, Triple-DES	X	X
Hash (HMAC)	Execute	HMAC SHA keys	X	X
Key generation	Write/execute	Symmetric key AES, Triple-DES, Asymmetric RSA, DSA, ECDSA, Diffie-Hellman public and private keys	X	X
Key agreement	Execute	DH public/private key	X	X
Perform Self-Tests	Execute/read	NA	X	X
DRBG	Execute	Seed input	X	X
Show Status	Execute	NA	X	X
Signature signing	Execute	Asymmetric private key DSA, RSA, ECDSA	X	X
Signature verification	Execute	Asymmetric public key DSA, RSA, ECDSA	X	X
Zeroization	Execute	Symmetric key, asymmetric key, HMAC-SHA keys, seed key, seed	X	X
Module Initialization	Execute	All CSPs	X	

Table 7 – Module Services

4. PHYSICAL SECURITY

The module obtains its physical security from any platform running SR-OS with production grade components and standard passivation as allowed by FIPS 140-2 level 1.

5. OPERATIONAL ENVIRONMENT

The SRCM was tested on the following platforms that represent the required HW components that runs SR-OS and the SRCM.

Hardware Running SRCM	Processor	Platforms
1.5 GHz 10 core CPU on 7950 XRS-20 CPM-x20	Cavium OCTEON II CN6645	7950 XRS-20
1.5 GHz 10 core CPU on 7950 XRS-16c CPM-x16	Cavium OCTEON II CN6645	7950 XRS-16c
1.5 GHz 10 core CPU on 7750 SR CPM5	Cavium OCTEON II CN6645	SR-7
800 MHz 6 core CPU on 7750 SR-a CPM-a	Cavium OCTEON II CN6635	7750 SR-a4
1.3 GHz 10 core CPU on 7750 SR-e CPM-e	Cavium OCTEON II CN6645	7750 SR-2e
1.8 GHz 16 core CPU on SR-1 CPM-1	Cavium OCTEON III CN7360	7750 SR-1
1.8 GHz 16 core CPU on SR-1s CPM-1s	Cavium OCTEON III CN7360	7750 SR-1s
1.8 GHz 16 core CPU on SR-2s CPM-2s	Cavium OCTEON III CN7360	7750 SR-2s
1.5 GHz 10 core CPU on SR-s CPM-s	Cavium OCTEON II CN6645	7750 SR-14s, SR-7s

Table 8 – Hardware and Platforms Used to Test Module

6. KEY TABLE

6.1 Keys/CSPs Algorithms In FIPS-140-2 Mode

The following keys and CSPs are available when running in FIPS-140-2 mode for the SRCM:

Key or CSP	Usage (Service)	Storage	Generation/Input	Zeroization	Access Role (R,W,X)
Triple DES-CBC	SSHv2, TLS, AA Local List File	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-256-CTR	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-192-CTR	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-128-CTR	SSHv2, TLS Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-128-CBC	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
Triple DES-CBC	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-192-CBC	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-256-CBC	SSHv2, TLS, Secure Copy	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
HMAC-SHA-1	OSPF, IS-IS, RSVP, SSHv2, TLS, Software Integrity	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Command	R, W
HMAC-SHA-256	OSPF, IS-IS, SSHv2, Python Script Authentication	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Command	R, W
HMAC-SHA-512	SSHv2 Authentication	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Command	R, W
AES-128-CFB	SNMP Encryption	Non-Volatile memory (Obfuscated)	Operator – Manually	Command	R, W
RSA Public Key	SSHv2, TLS,	Non-Volatile memory	Approved DRBG, API parameter	Reboot, Command	R, W, X

		(Obfuscated)			
RSA Private Key	SSHv2, TLS, Certificate Signing Request generation, CMPv2	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Reboot, Command	R, W, X
DSA Public Key	SSHv2, TLS	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Reboot, Command	R, W, X
DSA Private Key	SSHv2, TLS, Certificate Signing Request generation, CMPv2	Non-Volatile memory (Obfuscated)	Approved DRBG, API parameter	Reboot, Command	R, W, X
ECDSA Public Key	SSHv2	Non-Volatile memory (Obfuscated)	Operator - Manually	Reboot, Command	R, W, X
ECDSA Private Key	Certificate Signing Request generation, CMPv2	Non-Volatile memory (Obfuscated	Approved DRBG, API parameter	Reboot, Command	R, W, X
Diffie-Hellman Private Key	SSHv2, TLS	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
Diffie-Hellman Public Key	SSHv2, TLS	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
DRBG Seed	Key generation	DRAM (plaintext)	Internally Generated	Reboot	R, W
DRBG Entropy	Key generation	DRAM (plaintext)	Internally Generated	Reboot	R, W
DRBG ‘V’ Value	Key generation	DRAM (plaintext)	Internally Generated	Reboot	R, W
DRBG ‘Key’ Value	Key generation	DRAM (plaintext)	Internally Generated	Reboot	R, W
AES-CMAC	MACsec	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X
AES-GCM	MACsec	DRAM (plaintext)	Approved DRBG, API parameter	Reboot, Command	R, W, X

Table 9 – Cryptographic Keys and CSPs

The module's AES-GCM implementation conforms to IG A.5 scenario #1 for TLS. The module is compatible with TLSv1.2 and supports the acceptable GCM cipher suites from Section 3.3.1 of SP 800-52 Rev1. The counter portion of the IV is set by the module within its cryptographic boundary. When the nonce explicit part of the IV exhausts the maximum number of possible values for a given session key, either party (the client or the server) that encounters this condition triggers a handshake to establish a new encryption key.

The TLS and SSH protocols govern the generation of their respective Triple-DES keys as specified in RFC 5246 (TLS) and RFC 4253 (SSH). The user is responsible for ensuring the module limits the number of encryptions with the same Triple-DES key to 2^20.

Access roles include "R"- Read, "W" – Write, and "X" – Execute.

The SSH, TLS, IPv6 and SNMP protocols have not been reviewed or tested by the CAVP or CMVP.

KDF protocols have been tested for SSH and TLS.

7. EMC/EMI (FCC COMPLIANCE)

The XRS/SR/ESS chassis where the CPM, SR-OS and SRCM runs were tested and found conformant to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class A (business use).

8. SELF-TESTS

8.1 Self-Tests on the CPM

When FIPS-140-2 mode is enabled, the node performs the following startup tests (Note: KATs marked by asterisk (*) are implemented but not used by any services implemented by the module in Approved mode of operation):

- Firmware integrity check on startup using HMAC-SHA-256 (Key Length: 512 bits)
- DRBG KAT and health test (Key Length: 128, 192, 256 bits)
- Triple-DES (encrypt/decrypt) KAT (Key Length: 192 bits)
- AES ECB (encrypt/decrypt) KAT* (Key Length: 128 bits)
- AES CCM (encrypt/decrypt) KAT* (Key Length: 128 bits)
- AES XTS (encrypt/decrypt) KAT* (Key Length: 128 bits, 256 bits)
- AES GCM (encrypt/decrypt) KAT (Key Length: 256 bits)
- HMAC SHA-1 KAT, HMAC SHA-224 KAT, HMAC-SHA-256 KAT, HMAC SHA-384 KAT, HMAC SHA-512 KAT (Key Length: 160 bits)
- AES CMAC KAT (Key Length: 128, 192, 256 bits)
- Triple-DES CMAC KAT* (Key Length: 192 bits)
- SHA-1 KAT, SHA-256 KAT, SHA-512 KAT
- RSA encrypt KAT, decrypt KAT, SigGen and SigVer KAT (Key Length: 2048 bits, Signature Type: PKCS and PSS)
- ECDSA SigGen and SigVer KAT (Key Length: 256 bits)
- DSA SigGen and SigVer KAT (Key Length: 2048 bits)

The DRBG health tests include the Repetition Count Test on the entropy source and the Continuous Random Number Generator Test.

Should any of these tests fail, the SRCM does not allow the node to continue booting the image. An error is displayed on the console port that indicates the failed test and the SRCM forces a reboot to attempt the self-tests again.

8.1.1 Cryptographic DRBG Startup Test

A known answer test is used by the DRBG on startup (by using a known seed). If the startup test fails, then an error message is printed on the console and the node will attempt the boot sequence again.

8.1.2 RSA Startup test

SRCM performs an initial startup test with a known public key, a known digital signature and a test that verifies it can perform a proper verification of the known signature with the known public key. If the SRCM fails to successfully perform this startup test, then a message is printed on the console, the SRCM causes the node to reboot and tries to perform all the startup tests successfully again from the beginning.

8.2 Conditional Test on the CPM

When FIPS-140-2 mode is enabled, the node performs the following conditional self-tests during normal operation of the node:

- Manual Key Entry Tests
- Pairwise Consistency Test for RSA, DSA and ECDSA
- Continuous Random Number Generator Test (CRNGT)

Descriptions of the tests are described in the following sections.

SRCM Failure

When a Conditional Test (e.g., the pairwise consistency tests or the CRNGT test) fails, then the SRCM is considered as failed. The node will print a message on the console that indicates that the SRCM has failed.

Manual Key Entry Tests

Cryptographic key or key components manually entered into the cryptographic module are entered using duplicate entries. If the duplicate entries do not match, the test shall fail.

Pairwise Consistency Test for RSA, DSA and ECDSA

The Pairwise Consistency Test is performed whenever public or private keys are generated. The consistency of RSA/DSA keys is tested by the calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail.

An additional test is performed on RSA key pairs. A plaintext value is encrypted by the RSA public key. The resulting ciphertext value is compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key is used to decrypt the ciphertext and the resulting value is compared to the original plaintext value. If the two values are not equal, the test shall fail.

Continuous Random Number Generator Test (CRNGT)

The CRNGT is performed for every RNG call. Each call to an RNG produces blocks of 128 bits. The first 128-bit block generated after power-up, initialization, or reset is not used, but is saved for comparison with the next 128-bit block to be generated. Each subsequent generation of a 128-bit block is compared with the previously generated block. The test shall fail if any two compared 128-bit blocks are equal.

9. FIPS-140 USER GUIDANCE

The following sections described the SR-OS user guidance for configuring the XRS/SR/ESS systems where the SRCM is embedded and accessed by SR-OS.

9.1 FIPS-140-2 Mode Configuration

To enable FIPS-140-2 on the XRS/SR/ESS a configurable parameter is available in the bof.cfg file. When configured in the bof.cfg, the node boots in FIPS-140-2 mode and the following behaviors are enabled on the node:

- Only FIPS-140-2 approved algorithms (except for two-key Triple-DES and Diffie-Hellman with key sizes less than 2048 bits) are available for encryption and authentication for any cryptographic function on the CPM where SR-OS and the SRCM reside
- Diffie-Hellman with non-compliant key sizes must not be used in FIPS mode; otherwise, the module will enter a non-FIPS mode.
- Startup tests are executed on the CPM when the node boots
- Conditional tests are executed when required during normal operation (e.g., manual key entry test, pairwise consistency checks and RNG tests)

The current state of the bof and the parameters used for booting can be verified with the following CLI commands:

```
*A:bkvm12>show bof
*A:bkvm12>show bof booted
```

The output of "show bof booted" would show "fips-140-2" instead of "no fips-140-2". Note the FIPS-140-2 parameter in the bof.cfg does not take effect until the node has been rebooted. When running in FIPS mode the system will display a value in the system command that indicates this is the case.

9.2 Configurations Not Allowed when running in FIPS-140-2 Mode

When the node is configured in FIPS-140-2 mode the following disallowed algorithms are present in CLI. Procedurally, the User must not configure the following algorithms and functions when running in FIPS-140-2 mode or reverse the configuration steps in Section 9.1:

- MD5
  - SNMP, OSPF, BGP, LDP, NTP authentication, multi-chassis redundancy
- HMAC-MD5

  SNMP, IS-IS, RSVP
- HMAC-MD5-96
  - SNMP
- HMAC-SHA-1-96

- SNMP, OSPF, BGP, LDP

- AES-128-CMAC-96
  - BGP, LDP

9.3 Non-FIPS-140-2 Mode

To disable FIPS-140-2 on the XRS/SR/ESS, the User must configure the bof with "no fips-140-2" and reboot the system to transition to the non-FIPS-140-2 mode. The User must delete persistent keys before switching modes.

The non-approved mode of operation implements all services as in the approved mode, as well as the following services with non-approved algorithms and key lengths:

Services	Non-Approved Algorithms/Key Lengths
BGP	Hash: MD5 MAC: HMAC-SHA-1-96, AES-128-CMAC-96
CMPv2	Hash: MD5 Asymmetric: RSA key length < 2048 bits, DSA: (L, N) ≠ (2048, 224), (2048, 256) or (3072, 256)
Certificate Signing Request generation	Hash: MD5 Asymmetric: RSA key length < 2048 bits, DSA: (L, N) ≠ (2048, 224), (2048, 256) or (3072, 256)
IPv6 Secure Neighbor Discovery	Asymmetric: RSA 1024-bit key
IS-IS	MAC: HMAC-MD5
LDP	Hash: MD5 MAC: HMAC-SHA-1-96, AES-128-CMAC-96
Multi-chassis Redundancy	Hash: MD5
NTP	Hash: MD5
OSPF	Hash: MD5 MAC: HMAC-SHA-1-96
RSVP	MAC: HMAC-MD5
SSHv1	Symmetric: DES, Blowfish Asymmetric: 1024-bit RSA, 1024-bit Diffie-Hellman
SSHv2	Symmetric: Blowfish, CAST-128, RC4, Rijndael MAC: HMAC-MD5, HMAC-MD5-96, HMAC-SHA-1-96, HMAC-RIPEMD-160 Asymmetric: 1024-bit Diffie-Hellman
SNMP	Hash: MD5 MAC: HMAC-MD5, HMAC-MD5-96, HMAC-SHA-1-96
TLS	Asymmetric: RSA key sizes < 2048 bits (digital signature generation)

Table 10 - Services with Non-Approved Algorithms and Key Lengths

The module supports the Crypto Officer and User roles while in the non-Approved mode of operation.

10.REFERENCES

[FIPS 140-2] FIPS PUB 140-2, *Security Requirements for Cryptographic Modules*, May 25, 2001, CHANGE NOTICES (12-03-2002). http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

[FIPS 140-2 DTR] Derived Test Requirements for FIPS PUB 140-2, *Security Requirements for Cryptographic Modules*, January 4, 2011 Draft. http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402DTR.pdf

[FIPS 140-2 IG] Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program, December 3, 2019. http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf