FIPS 140-2 Non-Proprietary Security Policy

Rubrik Cryptographic Library

Software Version 1.0

Document Version 1.1

June 1, 2016

Prepared For:

Rubrik, Inc.
299 South California Avenue, Suite 250
Palo Alto, CA 94306
www.rubrik.com

Prepared By:

SafeLogic Inc.
459 Hamilton Ave, Suite 306
Palo Alto, CA 94301
www.safelogic.com

Abstract

This document provides a non-proprietary FIPS 140-2 Security Policy for the Rubrik Cryptographic Library.

Table of Contents

1 Introduction 4
1.1 About FIPS 140 4
1.2 About this Document 4
1.3 External Resources 4
1.4 Notices 4
1.5 Acronyms 5

2 Rubrik Cryptographic Library 6
2.1 Cryptographic Module Specification 6
2.1.1 Validation Level Detail 6
2.1.2 Approved Cryptographic Algorithms 6
2.1.3 Non-Approved Cryptographic Algorithms 9
2.1.4 Non-Approved Mode of Operation 9
2.2 Module Interfaces 11
2.3 Roles, Services, and Authentication 12
2.3.1 Operator Services and Descriptions 12
2.3.2 Operator Authentication 13
2.4 Physical Security 13
2.5 Operational Environment 13
2.6 Cryptographic Key Management 14
2.6.1 Random Number Generation 15
2.6.2 Key/Critical Security Parameter (CSP) Authorized Access and Use by Role and Service/Function 16
2.6.3 Key/CSP Storage 16
2.6.4 Key/CSP Zeroization 16
2.7 Self-Tests 16
2.7.1 Power-On Self-Tests 16
2.7.2 Conditional Self-Tests 17
2.7.3 Cryptographic Function 18
2.8 Mitigation of Other Attacks 18

3 Guidance and Secure Operation 19
3.1 Crypto Officer Guidance 19
3.1.1 Software Installation 19
3.1.2 Additional Rules of Operation 19
3.2 User Guidance 19
3.2.1 General Guidance 19

List of Tables

Table 1 – Acronyms and Terms 5
Table 2 – Validation Level by FIPS 140-2 Section 6
Table 3 – FIPS-Approved Algorithm Certificates 8
Table 4 – Logical Interface / Physical Interface Mapping 12
Table 5 – Module Services, Roles, and Descriptions 13
Table 6 – Module Keys/CSPs 15
Table 7 – Power-On Self-Tests 17
Table 8 – Conditional Self-Tests 18

List of Figures

Figure 1 – Module Boundary and Interfaces Diagram 11

1 Introduction

1.1 About FIPS 140

Federal Information Processing Standards Publication 140-2 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic modules to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Communications Security Establishment Canada (CSEC) Cryptographic Module Validation Program (CMVP) run the FIPS 140-2 program. The NVLAP accredits independent testing labs to perform FIPS 140 testing; the CMVP validates modules meeting FIPS 140-2 validation. *Validated* is the term given to a module that is documented and tested against the FIPS 140-2 criteria.

More information is available on the CMVP website at http://csrc.nist.gov/groups/STM/cmvp/index.html.

1.2 About this Document

This non-proprietary Cryptographic Module Security Policy for the Rubrik Cryptographic Library from Rubrik, Inc. provides an overview of the product and a high-level description of how it meets the security requirements of FIPS 140-2. This document contains details on the module's cryptographic keys and critical security parameters. This Security Policy concludes with instructions and guidance on running the module in a FIPS 140-2 mode of operation.

The Rubrik Cryptographic Library may also be referred to as the "module" in this document.

1.3 External Resources

The Rubrik website (www.rubrik.com) contains information on Rubrik services and products. The Cryptographic Module Validation Program website contains links to the FIPS 140-2 certificate and Rubrik contact information.

1.4 Notices

1.5 Acronyms

The following table defines acronyms found in this document:

Acronym	Term
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
CMVP	Cryptographic Module Validation Program
CO	Crypto Officer
CSEC	Communications Security Establishment Canada
CSP	Critical Security Parameter
DES	Data Encryption Standard
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
EC	Elliptic Curve
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
GPC	General Purpose Computer
GUI	Graphical User Interface
HMAC	(Keyed-) Hash Message Authentication Code
KAT	Known Answer Test
MAC	Message Authentication Code
MD	Message Digest
NIST	National Institute of Standards and Technology
OS	Operating System
PKCS	Public-Key Cryptography Standards
PRNG	Pseudo Random Number Generator
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, and Adleman
SHA	Secure Hash Algorithm
SSL	Secure Sockets Layer
Triple-DES	Triple Data Encryption Algorithm
TLS	Transport Layer Security
USB	Universal Serial Bus

Table 1 – Acronyms and Terms

2 Rubrik Cryptographic Library

2.1 Cryptographic Module Specification

The Rubrik Cryptographic Library provides cryptographic functions for the Rubrik Hybrid Appliances.

The module's logical cryptographic boundary is the shared library files and their integrity check HMAC files. The module is a multi-chip standalone embodiment installed on a General Purpose Device.

All operations of the module occur via calls from host applications and their respective internal daemons/processes. As such there are no untrusted services calling the services of the module.

2.1.1 Validation Level Detail

The following table lists the level of validation for each area in FIPS 140-2:

FIPS 140-2 Section Title	Validation Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	N/A
Operational Environment	1
Cryptographic Key Management	1
Electromagnetic Interference / Electromagnetic Compatibility	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A

Table 2 – Validation Level by FIPS 140-2 Section

2.1.2 Approved Cryptographic Algorithms

The module’s cryptographic algorithm implementations have received the following certificate numbers from the Cryptographic Algorithm Validation Program:

Algorithm	CAVP Certificate
AES **ECB** (e/d; 128 , 192 , 256) **CBC** (e/d; 128 , 192 , 256) **CFB1** (e/d; 128 , 192 , 256) **CFB8** (e/d; 128 , 192 , 256) **OFB** (e/d; 128 , 192 , 256) **CTR** (ext only; 128 , 192 , 256) **CCM** (KS: 128 , 192 , 256) **CMAC** (Generation/Verification) (KS: 128, 192, 256) **GCM** (KS: AES_128(e/d), AES_192(e/d), AES_256(e/d)) **GMAC_Supported** **XTS** ((KS: XTS_128((e/d) (f/p)) KS: XTS_256((e/d) (f/p))	2273
HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC- SHA-384, HMAC-SHA-512	1391
DSA, DSA 2 FIPS 186-2 **PQR Ver**: Sig Ver- 1024-bit FIPS 186-4 **PQG Gen**: 2048 & 3072 (using SHA-2) **PQG Ver**: 1024, 2048 & 3072 (using SHA-1 and SHA-2) **Key Pair**: 2048-bit & 3072-bit **Sig Gen**: 2048-bit & 3072-bit (using SHA-2) **Sig Ver**: 1024-bit. 2048-bit & 3072-bit (using SHA-1 & SHA-2)	709
ECDSA, ECDSA2 FIPS 186-2 **PKG**: Curves (P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409 & B-571) **PKV**: Curves All P, K & B FIPS 186-4 **PKG**: Curves (P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409 & B-571) **PKV**: Curves All P, K & B **Sig Gen**: Curves (P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409 & B-571) (SHA-2) **Sig Ver**: Curves (P-192, P-224, P-256, P-384, P-521, K-163, K-233, K-283, K-409, K-571, B-163, B-233, B-283, B-409 & B-571) (any SHA size)	368
RSA (X9.31, PKCS #1.5, PSS)	1166

FIPS 186-2 **ANSIX9.31** Key Gen: 2048-bit, 3072-bit & 4096-bit Sig Gen: 2048-bit, 3072-bit & 4096 bit (any SHA size) Sig Ver: 1024-bit, 1536-bit, 2048-bit, 3072-bit & 4096-bit (any SHA size) **PKCS1 V1 5** Sig Gen: 2048-bit, 3072-bit & 4096-bit (any SHA size) Sig Ver: 1024-bit, 1536-bit, 2048-bit, 3072-bit & 4096-bit (any SHA size) **PSS** Sig Gen: 2048-bit, 3072-bit & 4096-bit (any SHA size) Sig Ver: 1024-bit, 1536-bit, 2048-bit, 3072-bit & 4096-bit (any SHA size) FIPS 186-4 **ANSIX9.31** Sig Gen: 2048-bit (using SHA-2) Sig Ver: 1024-bit, 2048-bit, & 3072-bit (any SHA size) **PKCS1 V1 5** Sig Gen: 2048-bit & 3072-bit (using SHA-2) Sig Ver: 1024-bit, 2048-bit, & 3072-bit (any SHA size) **PSS** Sig Gen: 2048-bit & 3072-bit (using SHA-2) Sig Ver: 1024-bit, 2048-bit, & 3072-bit (any SHA size)	
SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	1954
Triple-DES **TECB**(KO 1 e/d, KO 2 d only) **TCBC**(KO 1 e/d, KO 2 d only) **TCFB1**(KO 1 e/d, KO 2 d only) **TCFB8**(KO 1 e/d, KO 2 d only) **TCFB64**(KO 1 e/d, KO 2 d only) **TOFB**(KO 1 e/d, KO 2 d only) **CMAC**(KS: 3-Key; Generation/Verification; Block Size(s): Full / Partial)	1420
SP 800-90 DRBG (Hash_DRBG, HMAC_DRBG, CTR_DRBG)	281
CVL (ECC CDH KAS)	44

Table 3 – FIPS-Approved Algorithm Certificates

2.1.3 Non-Approved Cryptographic Algorithms

The module supports the following non-FIPS 140-2 approved but allowed algorithms:

- RSA (key wrapping; key establishment methodology provides between 112 and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength)
- EC Diffie-Hellman (key agreement; key establishment methodology provides between 112 and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength)

2.1.4 Non-Approved Mode of Operation

The module supports a non-approved mode of operation. The algorithms listed in this section are not to be used by the operator in the FIPS Approved mode of operation.

The following algorithms are disallowed as of January 1, 2016 per the NIST SP 800-131A algorithm transitions:

- Random Number Generator Based on ANSI X9.31 Appendix A.2.4
- Two-Key Triple DES Encryption

The following algorithm is disallowed as of October 29, 2015 per the removal from NIST SP 800-90A:

- Dual EC DRBG

The following algorithms are disallowed as of January 1, 2014 per the NIST SP 800-131A algorithm transitions:

- FIPS 186-2 DSA (using SHA-1): PQG Gen- 1024-bit
  Key Gen- 1024-bit
  Sig Gen- 1024-bit
- FIPS 186-4 DSA PQG Gen, 1024-bit (any SHA size)
  Key Gen, 1024-bit
  Sig Gen, 1024-bit (any SHA size), 2048-bit & 3072-bit using SHA-1
- FIPS 186-2 RSA **ANSIX9.31**
  Key Gen 1024 & 1536

  ANSIX9.31
  Sig Gen 1024 & 1536 (any SHA size); 2048, 3072 & 4096 using SHA-1

  PKCSI V1 5
  Sig Gen 1024 & 1536 (any SHA size); 2048, 3072 & 4096 using SHA-1

  PSS
  Sig Gen 1024 & 1536 (any SHA size); 2048, 3072 & 4096 using SHA-1

- FIPS 186-4 RSA **ANSIX9.31**
  Sig Gen 1024 using SHA-1

  PKCSI V1 5
  Sig Gen 1024 using SHA-1

  PSS
  Sig Gen 1024 using SHA-1

- FIPS 186-2 ECDSA **PKG: Curves** P-192, K-163 & B-163
  Sig Gen Curves All P, K & B

- FIPS 186-4 ECDSA **PKG: Curves** P-192, K-163 & B-163

  Sig Gen Curves P-224, P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409 & B-571) (using SHA-1)
  P-192-, K-163 & B-163 (any SHA size)

- CVL (ECC CDH KAS) (non-compliant less than 112 bits of encryption strength)

2.2 Module Interfaces

The figure below shows the module's physical and logical block diagram:

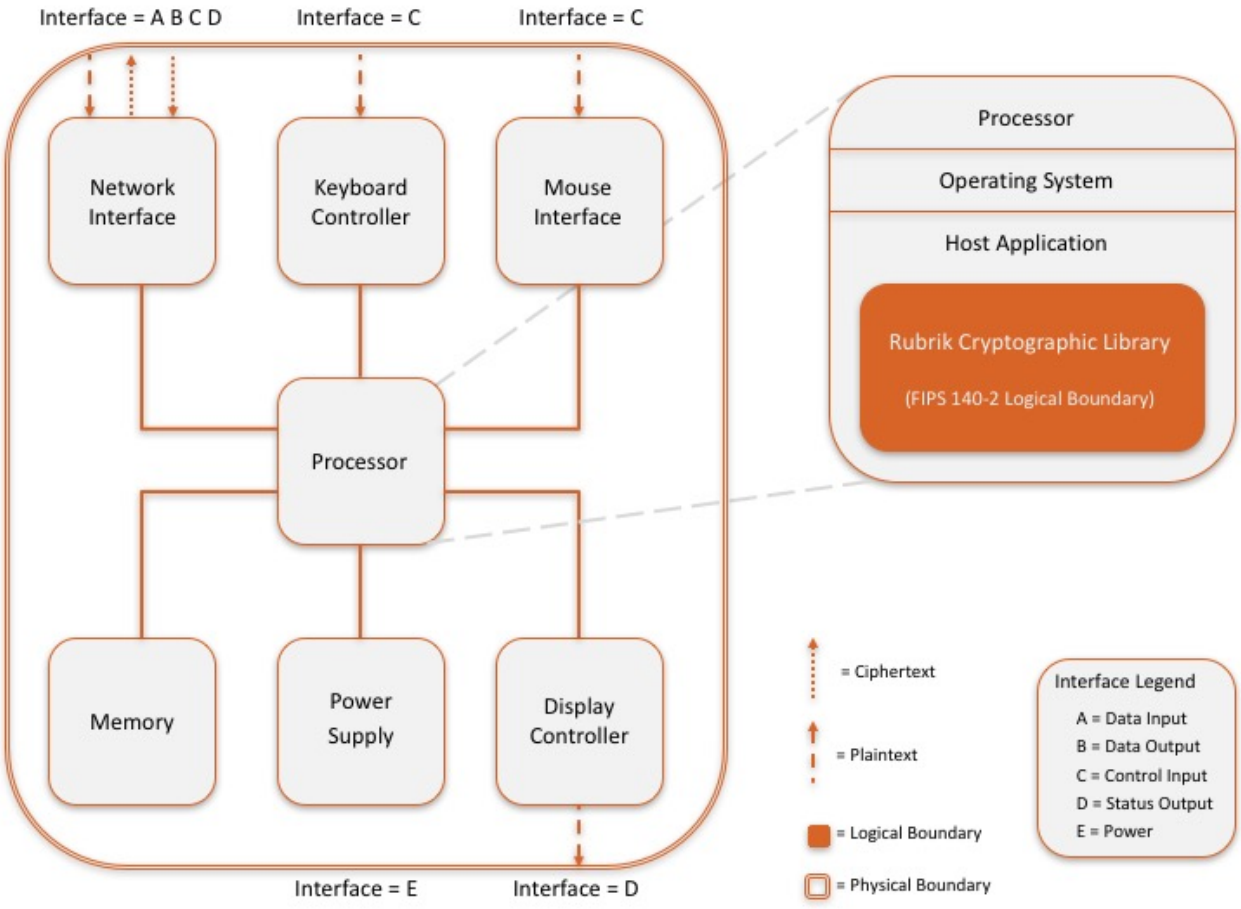

Figure 1 – Module Boundary and Interfaces Diagram

The interfaces (ports) for the physical boundary include the computer keyboard port, mouse port, network port, USB ports, display and power plug. When operational, the module does not transmit any information across these physical ports because it is a software cryptographic module. Therefore, the module's interfaces are purely logical and are provided through the Application Programming Interface (API) that a calling daemon can operate. The logical interfaces expose services that applications directly call, and the API provides functions that may be called by a referencing application (see Section 2.3 – Roles, Services, and Authentication for the list of available functions). The module distinguishes between logical interfaces by logically separating the information according to the defined API.

The API provided by the module is mapped onto the FIPS 140- 2 logical interfaces: data input, data output, control input, and status output. Each of the FIPS 140- 2 logical interfaces relates to the module's callable interface, as follows:

FIPS 140-2 Interface	Logical Interface	Module Physical Interface
Data Input	Input parameters of API function calls	Network Interface
Data Output	Output parameters of API function calls	Network Interface
Control Input	API function calls	Keyboard Interface, Mouse Interface
Status Output	For FIPS mode, function calls returning status information and return codes provided by API function calls.	Display Controller
Power	None	Power Supply

Table 4 – Logical Interface / Physical Interface Mapping

As shown in Figure 1 – Module Boundary and Interfaces Diagram and Table 5 – Module Services, Roles, and Descriptions, the output data path is provided by the data interfaces and is logically disconnected from processes performing key generation or zeroization. No key information will be output through the data output interface when the module zeroizes keys.

2.3 Roles, Services, and Authentication

The module supports a Crypto Officer and a User role. The module does not support a Maintenance role. The User and Crypto-Officer roles are implicitly assumed by the entity accessing services implemented by the Module.

2.3.1 Operator Services and Descriptions

The module supports services that are available to users in the various roles. All of the services are described in detail in the module's user documentation. The following table shows the services available to the various roles and the access to cryptographic keys and CSPs resulting from services:

Service	Roles	CSP / Algorithm	Permission
Module initialization	Crypto Officer	None	CO: execute
Symmetric encryption/decryption	User	AES Key, Triple-DES Key	User: read/write/execute
Digital signature	User	RSA Private Key, DSA Private Key	User: read/write/execute
Symmetric key generation	User	AES Key, Triple-DES Key	User: read/write/execute
Asymmetric key generation	User	RSA Private Key, DSA Private Key	User: read/write/execute

Service	Roles	CSP / Algorithm	Permission
Keyed Hash (HMAC)	User	HMAC Key HMAC SHA-1, HMAC SHA- 224, HMAC SHA-256, HMAC SHA-384, HMAC SHA-512	User: read/write/execute
Message digest (SHS)	User	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	User: read/write/execute
Random number generation	User	DRBG Seed and Seed Key	User: read/write/execute
Show status	Crypto Officer User	None	User and CO: execute
Self test	User	All CSPs	User: read/execute
Zeroize	Crypto Officer User	All CSPs	CO: read/write/execute

Table 5 – Module Services, Roles, and Descriptions

2.3.2 Operator Authentication

As required by FIPS 140-2, there are two roles (a Crypto Officer role and User role) in the module that operators may assume. As allowed by Level 1, the module does not support authentication to access services. As such, there are no applicable authentication policies. Access control policies are implicitly defined by the services available to the roles as specified in Table 5 – Module Services, Roles, and Descriptions.

2.4 Physical Security

This section of requirements does not apply to this module. The module is a software-only module and does not implement any physical security mechanisms.

2.5 Operational Environment

The module operates on a general purpose computer (GPC) running a general purpose operating system (GPOS). For FIPS purposes, the module is running on this operating system in single user mode and does not require any additional configuration to meet the FIPS requirements.

The module was tested on the following platforms:

- Red Hat Enterprise Linux 6.3 on a Dell Optiplex 755
- CentOS 6.3 on a Dell Optiplex 755
- CentOS 6.3 on a GigaVUE-TA1

- SUSE Linux Enterprise 11SP2 on a Dell Optiplex 755

The cryptographic module is also supported on the following operating environments for which operational testing and algorithm testing was not performed:

- Ubuntu 12, 14, 14.04, 15 and 16
- Red Hat Enterprise Linux 5, 7 and 7.2
- CentOS 5, 6, 7 and 7.2
- Debian 7 and 8
- openSUSE 13
- SUSE 12
- Oracle Linux 5, 6 and 7
- Fedora 22 and 23

Compliance is maintained for other versions of the respective operating system family where the binary is unchanged. No claim can be made as to the correct operation of the module or the security strengths of the generated keys when ported to an operational environment which is not listed on the validation certificate.

The GPC(s) used during testing met Federal Communications Commission (FCC) FCC Electromagnetic Interference (EMI) and Electromagnetic Compatibility (EMC) requirements for business use as defined by 47 Code of Federal Regulations, Part15, Subpart B. FIPS 140-2 validation compliance is maintained when the module is operated on other versions of the GPOS running in single user mode, assuming that the requirements outlined in NIST IG G.5 are met.

2.6 Cryptographic Key Management

The table below provides a complete list of Critical Security Parameters used within the module:

Keys and CSPs	Storage Locations	Storage Method	Input Method	Output Method	Zeroization	Access
AES Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
Triple-DES Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
RSA Public Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
RSA Private Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD

Keys and CSPs	Storage Locations	Storage Method	Input Method	Output Method	Zeroization	Access
DSA Public Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
DSA Private Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
HMAC Key	RAM	Plaintext	API call parameter	None	power cycle cleanse()	CO: RWD U: RWD
Integrity Key	Module Binary	Plaintext	None	None	None	CO: RWD U: RWD
EC DSA Private Key	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
EC DSA Public Key	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
EC DH Public Components	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
EC DH Private Components	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
HMAC DRBG Entropy	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
HMAC DRBG V Value (Seed Length)	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
HMAC DRBG Key	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD
HMAC DRBG init_seed	RAM	Plaintext	None	None	power cycle cleanse()	CO: RWD U: RWD

R = Read W = Write D = Delete

Table 6 – Module Keys/CSPs

The application that uses the module is responsible for appropriate destruction and zeroization of the key material. The module provides functions for key allocation and destruction which overwrite the memory that is occupied by the key information with zeros before it is deallocated.

2.6.1 Random Number Generation

The module uses SP800-90A DRBGs for creation of asymmetric and symmetric keys.

The module accepts input from entropy sources external to the cryptographic boundary for use as seed material for the module's Approved DRBGs. The calling application of the module shall use entropy sources that meet the security strength required for the random bit generation mechanism as shown in NIST Special Publication 800-90A Table 2 (Hash_DRBG, HMAC_DRBG) and Table 3 (CTR_DRBG).

The module performs continual tests on the random numbers it uses to ensure that the seed and seed key input to the Approved DRBGs do not have the same value. The module also performs continual tests on the output of the Approved DRBGs to ensure that consecutive random numbers do not repeat.

2.6.2 Key/Critical Security Parameter (CSP) Authorized Access and Use by Role and Service/Function

An authorized application as user (the User role) has access to all key data generated during the operation of the module.

2.6.3 Key/CSP Storage

Public and private keys are provided to the module by the calling process and are destroyed when released by the appropriate API function calls or during power cycle. The module does not perform persistent storage of keys.

2.6.4 Key/CSP Zeroization

The application is responsible for calling the appropriate destruction functions from the API. The destruction functions then overwrite the memory occupied by keys with zeros and deallocates the memory. This occurs during process termination / power cycle. Keys are immediately zeroized upon deallocation, which sufficiently protects the CSPs from compromise.

2.7 Self-Tests

FIPS 140-2 requires that the module perform self-tests to ensure the integrity of the module and the correctness of the cryptographic functionality at start up. In addition some functions require continuous verification of function, such as the random number generator. All of these tests are listed and described in this section. In the event of a self-test error, the module will log the error and will halt. The module must be initialized into memory to resume function.

The following sections discuss the module's self-tests in more detail.

2.7.1 Power-On Self-Tests

Power-on self-tests are executed automatically when the module is loaded into memory. The module verifies the integrity of the runtime executable using a HMAC-SHA1 digest computed at build time. If the fingerprints match, the power-up self-tests are then performed. If the power-up self-test is successful, a flag is set to place the module in FIPS mode.

TYPE	DETAIL
Software Integrity Check	• HMAC-SHA1 on all module components
Known Answer Tests[1]	• AES encrypt/decrypt • AES GCM • AES CCM • XTS-AES • AES CMAC • Triple-DES CMAC • ECDH • HMAC-SHA1 • HMAC-SHA224 • HMAC-SHA256 • HMAC-SHA384 • HMAC-SHA512 • RSA • SHA-1 • SHA-224 • SHA-256 • SHA-384 • SHA-512 • SP 800-90 DRBG (Hash_DRBG, HMAC_DRBG, CTR_DRBG) • Triple-DES encrypt/decrypt • ECC CDH
Pair-wise Consistency Tests	• DSA • RSA • ECDSA

Table 7 – Power-On Self-Tests

Input, output, and cryptographic functions cannot be performed while the Module is in a self-test or error state because the module is single-threaded and will not return to the calling application until the power-up self-tests are complete. If the power-up self-tests fail, subsequent calls to the module will also fail - thus no further cryptographic operations are possible.

2.7.2 Conditional Self-Tests

The module implements the following conditional self-tests upon key generation, or random number generation (respectively):

TYPE	DETAIL
Pair-wise Consistency Tests	• DSA • RSA • ECDSA

[1] Note that all SHA-X KATs are tested as part of the respective HMAC SHA-X KAT. SHA-1 is also tested independently.

TYPE	DETAIL
Continuous RNG Tests	• Performed on all Approved DRBGs, the non-approved X9.31 RNG, and the non-approved DUAL_EC_DRBG

Table 8 – Conditional Self-Tests

2.7.3 Cryptographic Function

The module verifies the integrity of the runtime executable using a HMAC-SHA1 digest which is computed at build time. If this computed HMAC-SHA1 digest matches the stored, known digest, then the power-up self-test (consisting of the algorithm-specific Pairwise Consistency and Known Answer tests) is performed. If any component of the power-up self-test fails, an internal global error flag is set to prevent subsequent invocation of any cryptographic function calls. Any such power-up self-test failure is a hard error that can only be recovered by reinstalling the module[2]. The power-up self-tests may be performed at any time by reloading the module.

No operator intervention is required during the running of the self-tests.

2.8 Mitigation of Other Attacks

The Module does not contain additional security mechanisms beyond the requirements for FIPS 140-2 Level 1 cryptographic modules.

[2] The initialization function could be re-invoked but such re-invocation does not provide a means from recovering from an integrity test or known answer test failure

3 Guidance and Secure Operation

3.1 Crypto Officer Guidance

3.1.1 Software Installation

The module is provided directly to solution developers and is not available for direct download to the general public. The module and its host application are to be installed on an operating system specified in Section 2.5 or one where portability is maintained.

3.1.2 Additional Rules of Operation

1. The writable memory areas of the module (data and stack segments) are accessible only by the application so that the operating system is in "single user" mode, i.e. only the application has access to that instance of the module.

2. The operating system is responsible for multitasking operations so that other processes cannot access the address space of the process containing the module.

3.2 User Guidance

3.2.1 General Guidance

The module is not distributed as a standalone library and is only used in conjunction with the solution.

The end user of the operating system is also responsible for zeroizing CSPs via wipe/secure delete procedures.

If the module power is lost and restored, the calling application can reset the IV to the last value used.