Citrix Systems, Inc.

Citrix FIPS Cryptographic Module

Non-Proprietary FIPS 140-2 Cryptographic Module Security Policy

Version: 1.8

Date: 2019-01-08

Acknowledgements

Citrix Systems, Inc. is both the vendor and consumer of this cryptographic module.

Jon Andersen, Ben Tucker, Przemek Dziedzic, Fernando Cervantes, Walter Olds, Sergio Borge, Chris Mayers, and Timothy Gaylor in no particular order are recognized as having either architected or engineered the software, performed testing, and managed laboratory submission of the module.

Change History

Version	Date	Updated By	Change
1.0	2017-03-22	Ben Tucker	Initial document release to validators.
1.1	2017-05-30	Ben Tucker	Updates per CMVP comments.
1.2	2017-06-07	Ben Tucker	Added OE19, non-accelerated OE.
1.3	2017-07-21	Ben Tucker	Added KTS, Software-Hybrid and hardware acceleration information.
1.4	2017-07-27	Ben Tucker	Added Processor Pictures as table 4.
1.5	2017-07-28	Ben Tucker	Removed OE19.
1.6	2017-08-04	Ben Tucker	Updated Physical Security to level 1 from N/A.
1.7	2018-07-26	Ben Tucker	Added Operational Environments 20 through 36
1.8	2019-01-08	Ben Tucker	Added SP 800-133 operation descriptions, textual clarifications.

Table of Contents

List of Tables

List of Figures

1 Introduction

This document defines the Security Policy for the Citrix FIPS Cryptographic Module, versions 1.0 and 1.0.1 (hereafter referred to as “the module”). It is a software toolkit which provides various cryptographic functions to support the Citrix product portfolio. This document describes how the module meets FIPS 140-2 Level 1 requirements, and how to operate it in the Approved mode of operation.

The tested configurations make use of Processor Algorithm Accelerators (PAAs): NEON extensions, AES/SHA Acceleration, AES-NI, or some combination thereof. Details are specified in the table below.

Table 1 – Cryptographic Module Configurations

	Operating Environment	Tested Platform
1	iOS 10 64bit on ARM v8-A with NEON extensions and AES/SHA Acceleration	Apple 12.9-inch iPad Pro (A1584)
2	Android 4.4 on ARM v7-A with NEON extensions	Google Nexus 5 (LG D820)
3	Android 5 on ARM v7-A with NEON extensions	Google Nexus 6 (Motorola Nexus 6 XT11003)
4	Android 6 on ARM v7-A with NEON extensions	Google Nexus 6 (Motorola Nexus 6 XT11003)
5	Windows 10 32bit on Intel Core i7 [4th Generation] with AES-NI	Lenovo 20CD00B2US
6	Android 6 on ARM v8-A with NEON extensions and AES/SHA Acceleration	Samsung Galaxy S6 (SM-G920T)
7	Android 7 on ARM v7-A with NEON extensions	Google Nexus 6 (Motorola Nexus 6 XT11003)
8	Android 7 on ARM v8-A with NEON extensions and AES/SHA Acceleration	Google Nexus 5X (LG H790)
9	Windows 10 64bit on Intel Core i7 [6th Generation] with AES-NI	Lenovo 20EV002JUS
10	Reserved (Operational Environment #10 is yet to be determined.)	
11	Linux 3.16 under XenServer 6 64bit on Intel Xeon 56xx series with AES-NI	Dell PowerEdge C6100
12	Linux 3.16 under ESXi 5 64bit on Intel Xeon 56xx series with AES-NI	HP ProLiant DL2000
13	Linux 3.16 under Hyper-V on Windows Server 2012 R2 64bit on Intel Xeon 56xx series with AES-NI	HP ProLiant DL2000
14	FreeBSD 8.4 32bit on Intel Xeon E5-26xx v2 series with AES-NI	Citrix NetScaler MPX-14000-FIPS
15	FreeBSD 8.4 64bit on Intel Xeon E5-26xx v2 series with AES-NI	Citrix NetScaler MPX-14000-FIPS

16	Mac OS X 10.12 64bit on Intel Core i7 with AES-NI	Apple Macbook Pro (A1398)
17	Linux 3.13 64bit on Intel Core i7 [6th Generation] with AES-NI	Lenovo 20EV002JUS
18	ViewSonic Thin OS on ARM v8-A with NEON extensions	ViewSonic VS16585
19	Reserved (Operational Environment #19 is yet to be determined.)	
20	Android 8 64bit on ARM v8-A with NEON extensions and AES/SHA Acceleration	Google Pixel 2
21	iOS 11 on ARM v8-A with NEON extensions and AES/SHA Acceleration	iPhone X
22	Windows 10 64bit on Intel Core i7 [8th Generation] with AES-NI	CyberPowerPC GLC2460
23	Windows 8.1 64bit on Intel Core i7 [8th Generation] with AES-NI	CyberPowerPC GLC2460
24	Linux 3.10 64bit on Intel Core i7 [8th Generation] with AES-NI	CyberPowerPC GLC2460
25	Linux 4.15 64bit on Intel Core i7 [8th Generation] with AES-NI	CyberPowerPC GLC2460
26	Linux 4.x under XenServer 7 64bit on Intel Xeon E5-26XX v4 series with AES-NI	HPE ProLiant BL460c Gen9
27	Linux 4.x under ESXi 5 64bit on Intel Xeon E5-26XX v4 series with AES-NI	HPE ProLiant BL460c Gen9
28	Linux 4.x under Hyper-V on Windows Server 2012 R2 64bit on Intel Xeon E5-26XX v4 series with AES-NI	HPE ProLiant BL460c Gen9
29	Mac OS X 10.13 64bit on Intel Core i7 [4th Generation] with AES-NI	Apple Mac Mini
30	Linux 4.15 64bit on Intel Xeon E5-26XX v3 series with AES-NI	HPE ProLiant BL460c Gen9
31	Linux 4.15 64bit on Intel Xeon E5-26XX v4 series with AES-NI	HPE ProLiant BL460c Gen9
32	Windows Server 2016 under HyperV on Windows Server 2016 64bit on Intel Xeon E5-26XX v3 series with AES-NI	HPE ProLiant BL460c Gen9
33	Windows Server 2016 under HyperV on Windows Server 2016 64bit on Intel Xeon E5-26XX v4 series with AES-NI	HPE ProLiant BL460c Gen9
34	Linux 4.x under XenServer 7 64bit on Intel Xeon E5-24XX v2 series with AES-NI	Dell PowerEdge R320

35	Android 9 on ARM v8-A with NEON extensions and AES/SHA Acceleration	Google Pixel 2
36	NoTouch Desktop on ARM v8-A with NEON extensions and AES/SHA Acceleration	N-computing RX-HDX for Citrix

The module is a software library providing a C-language API for use by other processes that require cryptographic functionality. The module is a software-hybrid module, intended for multi-chip standalone physical embodiments. The physical boundary is the general purpose computer on which the module is installed; the logical boundary is the module API. The module performs no communications other than with the calling application, with the sole exception of RNG seeding via an application callback API function (e.g., from /dev/random or otherwise as determined by the application through the callback API function).

The FIPS 140-2 security levels for the module are as follows:

Table 2 – Security Level of Security Requirements

Security Requirement	Security Level
Cryptographic Module Specification	1
Cryptographic Module Ports and Interfaces	1
Roles, Services, and Authentication	1
Finite State Model	1
Physical Security	1
Operational Environment	1
Cryptographic Key Management	1
EMI/EMC	1
Self-Tests	1
Design Assurance	1
Mitigation of Other Attacks	N/A
Overall Level	**1**

1.1 Physical Cryptographic Boundary

The physical form of the module is the general purpose computer on which the module runs. The module relies on its API for all input and output.

1.2 Logical Cryptographic Boundary and Input/Output

Table 3 describes the module’s logical I/O.

Table 3 – Ports and Interfaces

Signal Description	Logical Interface Type
API entry point and corresponding stack parameters	Control in

Signal Description	Logical Interface Type
API entry point data input stack parameters	Data in
API entry point return values and status stack parameters	Status out
API entry point data output stack parameters	Data out

As a software-hybrid module, control of the physical ports is outside module scope. However, when the module is performing self-tests, or is in an error state, all output on the logical data output interface is inhibited. The module is single-threaded and in error scenarios returns only an error value (no data output is returned).

Figure 1 depicts the module's operational environment.

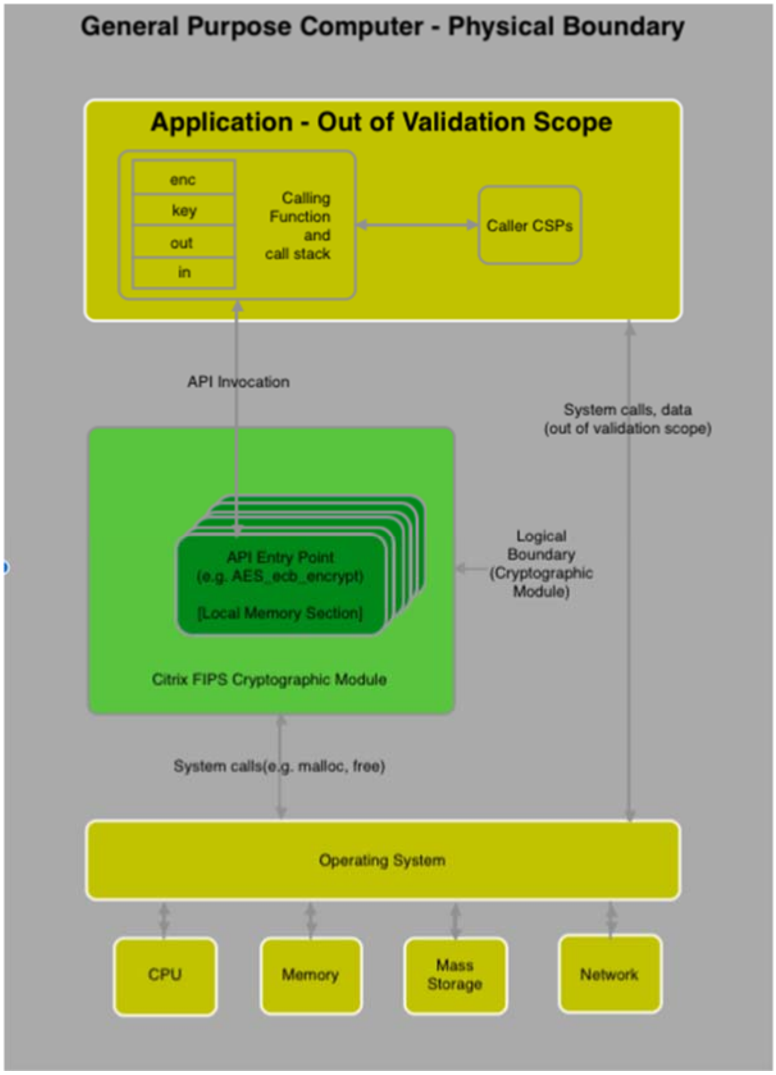

General Purpose Computer - Physical Boundary
Application - Out of Validation Scope
enc
key
out
in
Calling Function and call stack
Caller CSPs
API Invocation
System calls, data (out of validation scope)
API Entry Point (e.g. AES_ecb_encrypt)
[Local Memory Section]
Citrix FIPS Cryptographic Module
Logical Boundary (Cryptographic Module)
System calls(e.g. malloc, free)
Operating System
CPU
Memory
Mass Storage
Network

Figure 1 – Module Block Diagram

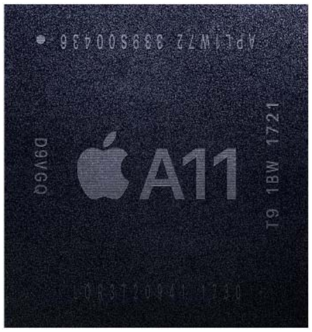
Table 4 – Processors

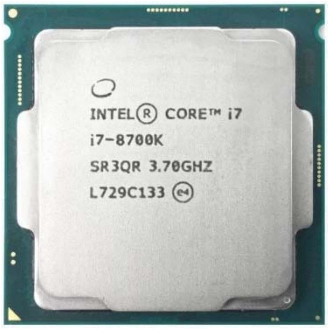
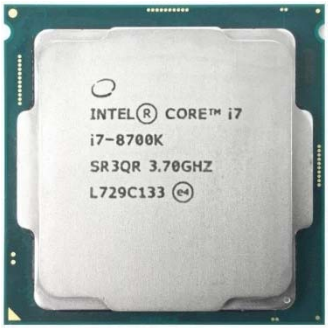
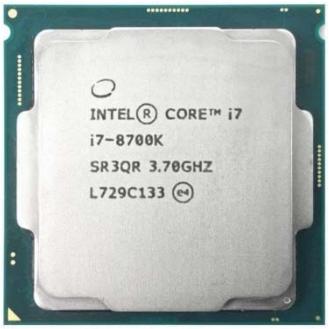
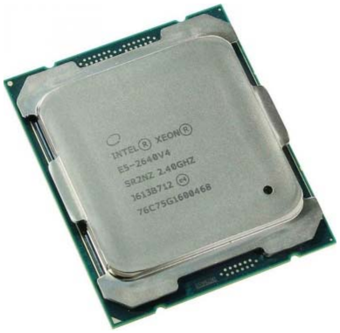
Environment	Processor
1	 anandtech.com
2	 alibaba.com
3	 phoronix.com
4	 phoronix.com

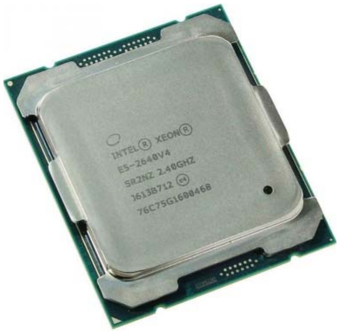
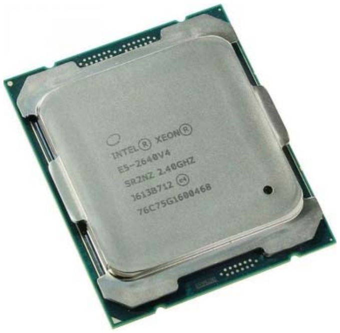
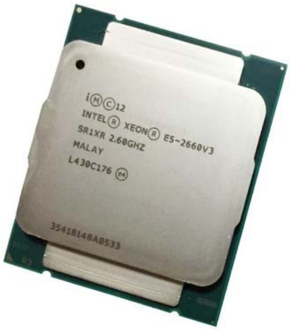
5	laptopmag.com
6	samsung.com
7	phoronix.com
8	alibaba.com

9	 pcworld.com
10	Reserved (Operational Environment #10 is yet to be determined.)
11	INTEL®©'09 E5645 INTEL® XEON® SLBWZ COSTA RICA 2.40GHZ/12M/5.86 32898275 dell.com
12	INTEL®©'09 E5645 INTEL® XEON® SLBWZ COSTA RICA 2.40GHZ/12M/5.86 32898275 dell.com
13	INTEL®©'09 E5645 INTEL® XEON® SLBWZ COSTA RICA 2.40GHZ/12M/5.86 32898275 dell.com

14	 alibaba.com
15	 alibaba.com
16	 laptopmag.com
17	 pcworld.com

18	 raspberrypi.org
19	N/A
20	 techprolonged.com
21	 wikimedia.org
22	 purepc.pl

23	 purepc.pl
24	 purepc.pl
25	 purepc.pl
26	 INTEL® XEON® E5-2640V4 SR2NZ 2.40GHZ J613B712 e4 76C75G1600468 alicdn.com

27	 INTEL® XEON® E5-2640V4 SR2NZ 2.40GHZ 3613B712 e4 76C75G1600468 alicdn.com
28	 INTEL® XEON® E5-2640V4 SR2NZ 2.40GHZ 3613B712 e4 76C75G1600468 alicdn.com
29	 cubeco.ir
30	 iM©12 INTEL® XEON® E5-2660V3 SR1XR 2.60GHZ MALAY L430C176 e4 3541814BA0533 alicdn.com

31	INTEL® XEON® E5-2640V4 SR2NZ 2.40GHZ 3613B712 76C75G1600468 alicdn.com
32	i(M)(C)12 INTEL® XEON® E5-2660V3 SR1XR 2.60GHZ MALAY L430C176 3541B148A0533 alicdn.com
33	INTEL® XEON® E5-2640V4 SR2NZ 2.40GHZ 3613B712 76C75G1600468 alicdn.com
34	i(M)(C)11 INTEL® XEON® E5-2420V2 SR1AJ 2.20GHZ COSTA RICA 3352B529 itinstock.com

35	 techprolonged.com
36	BROADCOM® BCM2837RIFBG TN1550 P11 559947-09 N3 W FCC ID: 2ABCB 20953 raspberrypi.org

1.3 Modes of Operation

The module supports both an Approved and a non-Approved mode of operation. The Approved mode is invoked by setting the appropriate flag (`FIPS_mode_set(1)`) and by following all procedural restrictions in this document, e.g., the algorithm restrictions in Section 2 and the operator-enforced security rules in Section 8.2.

The Approved mode flag can be checked with the command `FIPS_mode_get()`.

2 Cryptographic Functionality

The module implements the FIPS Approved and not-Approved-but-allowed cryptographic functions listed in the tables below. The implemented key establishment methodologies (KAS and various KDFs) operate on payload data and do not establish keys into the module itself.

Table 5 – Approved and CAVP Validated Cryptographic Functions

Algorithm	Description	Cert #
AES	[FIPS 197, SP 800-38A] Functions: Encryption, Decryption Modes: ECB, CBC, OFB, CFB1, CFB8, CFB128, CTR Key sizes: 128, 192, 256 bits	4397

Algorithm	Description	Cert #
AES-CMAC	[FIPS 197, SP 800-38B] Functions: MAC Generation, MAC Verification Key sizes: 128, 192, 256 bits	4397
AES-CCM	[FIPS 197, SP 800-38C] Functions: Authenticated Encryption, Authenticated Decryption Key sizes: 128, 192, 256 bits	4397
AES-GCM*	[FIPS 197, SP 800-38D] Functions: Authenticated Encryption, Authenticated Decryption Key sizes: 128, 192, 256 bits	4397
AES-XTS**	[FIPS 197, SP 800-38E] Functions: Encryption, Decryption Key sizes: 128, 256 bits	4397
AES-KW	[FIPS 197, SP 800-38F] Functions: Key Wrap, Key Unwrap: Forward and Inverse Modes: KW, KWP Key sizes: 128, 192, 256 bits Key establishment methodology provides between 128 and 256 bits of encryption strength.	4397
CVL: KAS Component	[All of SP 800-56A EXCEPT KDF] Parameter sets/Key sizes: dhOneFlow (FB, FC); Ephemeral Unified (EC, ED, EE)/P-256, P-384, P-512 [SP 800-56A Section 5.7.1.2 ECC CDH Primitive] Parameter sets/Key sizes: P-256, P-384, P-521, K-233, K-283, K-409, K-571, B-233, B-283, B-409, B-571 (ECC CDH with P-224 was also tested, but is not used.)	1106
CVL: Application Specific KDF†	[SP 800-135-rev1] Functions: TLS v1.0/1.1 KDF, TLS 1.2 KDF, ANSI X9.63-2001 KDF, SSH v2 KDF	1101, 1102 and 1103
CVL: RSASP	[FIPS 186-4, PKCS #1 v2.1] Function: Component Test (Signature Primitive) Key size: 2048	1105
CVL: RSADP	[SP 800-56B-rev1 §7.1.2] Function: Component Test (Decryption Primitive) Key size: 2048	1104

Algorithm	Description	Cert #
DRBG	[SP 800-90A] Functions: Hash DRBG, HMAC DRBG, CTR DRBG (Use df, No df) Security Strengths: 128, 192, and 256 DRBG instances are automatically seeded from the operating environment (e.g., by reading from /dev/random).	1417
DSA	[FIPS 186-4] Functions: PQG Generation, PQG Verification, Key Pair Generation, Signature Generation, Signature Verification Key sizes: 1024*, 2048, 3072 bits Hashes: SHA-1**, SHA-224, SHA-256, SHA-384, SHA-512 *DSA-1024 is restricted to signature verification and public key validation. **Signatures generated using SHA-1 are restricted to protocol use only.	1174
ECDSA	[FIPS 186-4] Functions: Key Pair Generation (ExtraRandomBits and TestingCandidates), Signature Generation, Signature Verification, Public Key Validation Curves and sizes: 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 Hashes: SHA-224, SHA-256, SHA-384, SHA-512 *These legacy curves are restricted to signature verification and public key validation.	1056
HMAC	[FIPS 198-1] Functions: MAC Generation, MAC Verification Hashes: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512	2923
KTS	[SP800-38F] Functions: Key Wrap, Key Unwrap Options: AES-KW, AES-KWP Key sizes: 128, 192, 256 bits Key establishment methodology provides between 128 and 256 bits of encryption strength.	AES #4397

Algorithm	Description	Cert #
RSA	[FIPS 186-4, ANSI X9.31-1998, and PKCS #1 v2.1 (PSS and PKCS1.5)] Functions: Key Pair Generation, Signature Generation, Signature Verification Key sizes: 1024*, 2048, 3072 bits Hashes: SHA-1*, SHA-224**, SHA-256, SHA-384, SHA-512 Signature schemes: X9.31, PKCS1.5, PSS *These legacy options are only supported for signature verification. **SHA-224 is only supported for PKCS1.5 and PSS.	2379
SHA	[FIPS 180-4] Functions: Digital Signature Generation, Digital Signature Verification, HMAC, standalone hashing Variants: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 (See other algorithm listings for specific uses of each variant.)	3626
Triple-DES	[SP 800-67, SP 800-20] Functions: Encryption, Decryption Modes: TECB, TCBC, TCFB-1, TCFB-8, TCFB-64, TOFB Key sizes: 3-key: 192	2371
Triple-DES-CMAC	[SP 800-67, SP 800-38B] Functions: MAC Generation, MAC Verification Key sizes: 3-key	2371

*When using GCM, care must be taken to never re-use IVs. In the event of unexpected interruption of a GCM session (e.g., power loss, system crash), keys and IVs must be redistributed.

**XTS must only be used for storage applications, e.g., full disk encryption. The XTS algorithm implementation includes a check to ensure Key_1 ≠ Key_2.

†Application Specific KDFs must only be used in the context of their respective applications, as per SP 800-135-rev1 (e.g., the SSH KDF must only be used in the context of SSH).

Table 6 – Approved Cryptographic Functions Tested with Vendor Affirmation

Algorithm	Description	Ref.
KAS: Key Agreement Using RSA	[SP 800-56B-rev1] Key sizes: 2048, 3072 bits Key Derivation Functions†: ANSI X9.42-2001 KDF**, ANS X9.63-2001 KDF (CVL #1101), SSH KDF (CVL #1102), TLS KDF (CVL #1103) Also using CVL Cert. #1104 (RSADP) Vendor Affirmation included examination of RSADP, RSAEP, RSASP Key Agreement Scheme provides between 112 and 256 bits of encryption strength.	IG §D.4
KAS: Key Agreement using DH, ECDH	[SP 800-56A-rev2] Parameter sets/Key sizes: FB, FC, EC, ED, EE Key Derivation Functions†: ANSI X9.42-2001 KDF**, ANS X9.63-2001 KDF (CVL #1101), SSH KDF (CVL #1102), TLS KDF (CVL #1103) Also using CVL Cert. #1106 (ECC CDH and KAS All-Except-KDF) Key Agreement Scheme provides between 112 and 256 bits of encryption strength.	IG §D.1-rev2
KDF, Password-Based*	[SP 800-132] Options: PBKDF with Option 1a, 1b, 2a, or 2b Functions: HMAC-based KDF using SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 Restrictions: Salt length ≥16 bytes, iteration count ≥1000 The strength of PBKDF-derived keys is limited by the strength of the input password. This implementation does not enforce a minimum password length. (i.e., one-character passwords are possible, but the resulting keys are trivially guessed.) It is the operator's responsibility to choose a password strength appropriate for their needs.	IG §D.6

Algorithm	Description	Ref.
CKG: Cryptographic Key Generation	[SP 800-133] Section 6.1 Asymmetric signature key generation using unmodified DRBG output • Performed in the crypto-module. Section 6.2 Asymmetric key establishment key generation using unmodified DRBG output • Performed in the crypto-module. Section 7.1 Direct symmetric key generation using unmodified DRBG output • Performed in the crypto-module. Note: The cryptomodule does not have a symmetric key generation function per se. Instead, the DRBG can be used by the operator to generate random data that can be imported into the module as symmetric key material. Section 7.3 Derivation of symmetric keys from a key agreement shared secret. • Performed in the crypto-module. Section 7.5 Derivation of symmetric keys from a password • Performed in the crypto-module. Section 7.6 Combining multiple keys and other data • Not supported.	IG §D.12

*Keys established with a Password-Based KDF (PBKDF) must only be used for storage application.

**The ANSI X9.42-2001 KDF uses SHS Cert. #3626.

†Application Specific KDFs must only be used in the context of their respective applications, as per SP 800-135-rev1 (e.g., the SSH KDF must only be used in the context of SSH).

Table 7 – Non-Approved but Allowed Cryptographic Functions

Algorithm	Description
MD5 within TLS	MD5 is a component of the TLSv1.0/v1.1 KDF. (SP800-135 rev1 §4.2.1)

Non-Approved Cryptographic Functions for use in non-FIPS mode only:

- RSA, DSA, ECDSA: non-legacy operations with disallowed key/hash sizes (e.g., RSA-1024 sig gen)
- ECC CDH: disallowed curve/key sizes (B/K/P 163, 192)
- MD5 for general use
- SHA-1 for non-protocol signature generation

2.1 Critical Security Parameters

All CSPs used by the module are described in this section. All usage of these CSPs by the module (including all CSP lifecycle states) is described in the services detailed in Section 4.

Table 8 – Critical Security Parameters (CSPs)

CSP	Size (bits)	Description / Usage
RSA SGK	2048/3072	RSA signature generation key
RSA KDK	2048/3072	RSA key decryption key
DSA SGK	2048/3072	DSA signature generation key
DH Private	2048/3072	DH private key agreement key
ECDSA SGK	224-571	ECDSA signature generation key
ECDH Private	224-571	ECDH private key agreement key
AES EDK	128/192/256	AES key for encrypt/decrypt as per SP800-38A
AES CMAC	128/192/256	AES CMAC key
AES CCM	128/192/256	AES CCM key
AES GCM	128/192/256	AES GCM key
AES KW	128/192/256	AES KW key (KW, KWP)

AES XTS	K1=128/256
	K2=128/256
		AES XTS keys

TDES EDK	192*	3-key Triple DES encrypt/decrypt key
TDES CMAC	192*	3-key Triple DES CMAC key
HMAC Key	160-512	HMAC key

Hash_DRBG State	V=440/880
	C=440/880
		V and C internal state values for Hash_DRBG

HMAC_DRBG State	V=160-512
	Key=160-512
		V and C internal state values for HMAC_DRBG

CTR_DRBG State	V=128
	Key=128-256
		V and Key internal state values for CTR_DRBG

Entropy Input	≥256	Entropy input pulled from the operating environment; used to seed a DRBG
KDF Input	≥128	Input materials to a Key Derivation Function
KDF Output	≥128	Key (or key block) output from a Key Derivation Function
Shared Secret	≥160	Shared secret established from a key agreement or key transport scheme.

* Triple-DES key size of 192 includes 24 parity bits.

2.2 Public Keys

Table 9 – Public Keys

Key	Size (bits)	Description / Usage
RSA SVK	1024/2048/3072	RSA signature verification key
RSA KEK	1024/2048/3072	RSA key encryption key
DSA SVK	1024/2048/3072	DSA signature verification key
DH Public	1024/2048/3072	DH public key agreement key
ECDSA SVK	192-571	ECDSA signature verification key
ECDH Public	224-571	ECDH public key agreement key

3 Roles, Authentication and Services

3.1 Assumption of Roles

The module supports two operator roles, User and Cryptographic Officer (CO). The roles have identical functionality, and are not authenticated. Multiple concurrent operators are not supported.

The table below lists the operator roles supported by the module. A new instance of the module is provided to each operator (i.e., calling application).

Table 10 – Roles Description

Role ID	Role Description	Authentication Type
CO	Operational usage of the module.	None
User	Operational usage of the module.	None

3.2 Services

All services implemented by the module are listed in the table below. Each service description also describes all usage of CSPs by the service.

Note that the non-Approved mode provides the same list of services, but with a wider selection of possible algorithms (i.e., the non-Approved algorithms listed below Table 6 may be used).

Table 11 –Services

Service	Roles	Description
Initialize	User, CO	Module initialization. Does not access CSPs.
Self-test	User, CO	Perform self tests (FIPS_selftest). Does not access CSPs.
Show status	User, CO	Functions that provide module status information: Version (as unsigned long, const char *, or the version number) FIPS Mode (Boolean) Does not access CSPs.
Zeroize	User, CO	Sub-services that destroy CSPs: DRBG Uninstantiate: zeroizes CSPs for a given DRBG Key Cleanse: zeroizes a given key All other services automatically overwrite CSPs stored in allocated memory. Stack cleanup is the responsibility of the calling application.

Service	Roles	Description
Random number generation	User, CO	Used for random number and symmetric key generation. Seed or reseed a DRBG instance Determine security strength of a DRBG instance Obtain random data Uses and updates Hash_DRBG CSPs, HMAC_DRBG CSPs, CTR_DRBG CSPs, Entropy Input.
Asymmetric key generation	User, CO	Used to generate DSA, ECDSA, RSA, DH, and ECDH keys: RSA SGK, RSA SVK; DSA SGK, DSA SVK; ECDSA SGK, ECDSA SVK; DH Private, DH Public; ECDH Private, ECDH Public There is one supported entropy strength for each mechanism and algorithm type, the maximum specified in SP800-90
Key Derivation	User, CO	Derivation of new keys using the implemented KDFs. Executes using KDF Input, KDF Output
Symmetric block cipher, SP800-38A	User, CO	Used to encrypt or decrypt data. Executes using AES EDK, TDES EDK (passed in by the calling process).
Symmetric block cipher, non-38A	User, CO	Used to process data with AES-CCM, AES-GCM, AES-CMAC, AES-XTS, or Triple-DES CMAC. Executes using the CSP corresponding to the given algorithm and block cipher mode (passed in by the calling process).
Key wrap/unwrap	User, CO	Symmetric key wrapping and unwrapping using AES Key Wrap. Uses the AES KW key.
Message digest	User, CO	Used to generate a SHA-1 or SHA-2 message digest. Does not access CSPs.
Keyed Hash	User, CO	Used to generate or verify data integrity with HMAC. Executes using HMAC Key (passed in by the calling process).
Key Encrypt/Decrypt	User, CO	Used to encrypt or decrypt a key value on behalf of the calling process Executes using RSA KDK, RSA KEK (passed in by the calling process).
Key agreement	User, CO	Used to perform key agreement primitives on behalf of the calling process Executes using DH Private, DH Public; ECDH Private, ECDH Public; RSA Private, RSA Public; RSA KDK, RSA KDK. (Passed in by the calling process.) Creates the Shared Secret.
Digital signature	User, CO	Used to generate or verify RSA, DSA or ECDSA digital signatures. Executes using RSA SGK, RSA SVK; DSA SGK, DSA SVK; ECDSA SGK, ECDSA SVK (passed in by the calling process).
Utility	User, CO	Miscellaneous helper functions. Does not access CSPs.

4 Self-tests

Each time the module is powered up it tests that the cryptographic algorithms still operate correctly and that sensitive data have not been damaged. Power up self–tests are available on demand by power cycling the module.

On power up or reset, the module performs the self-tests described in Table 12 below. All self-tests must be completed successfully prior to any other use of cryptography by the module. If one of the KATs fails, the module enters the error state.

Table 12 – Power Up Self-tests

Test Target	Type	Description
Software integrity	KAT	HMAC-SHA-1
HMAC	KAT	One KAT per SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 Per IG 9.3, this testing covers SHA POST requirements.
AES	KAT	Separate encrypt and decrypt, ECB mode, 128 bit key length
AES CCM	KAT	Separate encrypt and decrypt, 192 key length
AES GCM	KAT	Separate encrypt and decrypt, 256 key length
AES XTS	KAT	Separate encrypt and decrypt with AES-XTS-128 (for 256-bit XTS), and AES-XTS-256 (for 512-bit XTS)
AES KW	KAT	Separate encrypt and decrypt, 256 key length
AES CMAC	KAT	CMAC generate and verify, CBC mode, 128, 192, 256 key lengths
Triple-DES	KAT	Separate encrypt and decrypt, ECB mode, 3-Key of size 192
Triple-DES CMAC	KAT	CMAC generate and verify, CBC mode, 3-Key
RSA	KAT	Sign and verify using 2048 bit key, SHA-256, PKCS#1
DSA	PCT	Sign and verify using 2048 bit key, SHA-384
DRBG	KAT	CTR_DRBG: AES, 256 bit with and without derivation function HASH_DRBG: SHA-256 HMAC_DRBG: SHA-256
ECDSA	PCT	Keygen, sign, verify using P-224, K-233; with SHA-512.
ECC CDH	KAT	Shared secret calculation per SP 800-56A §5.7.1.2, IG 9.6
SSH KDF	KAT	SSH key derivation, SHA-256
X9.63 KDF	KAT	X9.63 key derivation, ECDH S/MIME
X9.42 KDF	KAT	X9.42 key derivation, SHA-256, DH S/MIME
RSA KAS	KAT	RSA Key Agreement
DH	KAT	DH Key Agreement

Table 13 – Conditional Self-tests

Test Target	Description
DRBG	DRBG Health Tests as per [SP800-90A §11.3]
DRBG	Continuous RNG test for stuck fault
Entropy source	Continuous RNG test on the entropy (DRBG seed) input
DSA	Pairwise consistency test on each generation of a key pair
ECDSA	Pairwise consistency test on each generation of a key pair
RSA	Pairwise consistency test on each generation of a key pair
DH	Pairwise consistency test on each generation of a key pair
ECDH	Pairwise consistency test on each generation of a key pair

5 Physical Security Policy

Physical security is not applicable to this Level 1 software-hybrid module.

6 Operational Environment

The module runs on a general purpose computer with hardware acceleration, including virtual environments (e.g., a virtual machine or hypervisor). (See Table 1 for a full list of OEs.) The operational environments provide industry-standard user access controls and separation of memory for different processes; this prevents other software components of the operating environment from accessing or tampering with the module's memory space.

The module uses HMAC-SHA1 as integrity protection against tamper of the module on the filesystem.

7 Mitigation of Other Attacks Policy

The module does not specifically mitigate attacks outside the scope of FIPS 140-2.

8 Security Rules and Guidance

8.1 Module-Enforced Security Rules

The module design corresponds to the following module security rules. This section documents the security rules enforced by the cryptographic module to implement the security requirements of this FIPS 140-2 Level 1 module.

1. The module provides two operator roles: User and Cryptographic Officer.
2. The module does not provide authentication.
3. The operator can command the module to perform the power up self-tests by re-instantiating the module or calling FIPS_selftest().
4. The module contains a Default Entry Point as per IG 9.10. Power up self-tests are run on module instantiation and do not require any operator action.

5. Data output is inhibited during self-tests and error states. Data output is logically disconnected from processes performing key generation and zeroization.
6. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
7. There are no restrictions on which keys or other CSPs are zeroized by the zeroization service.
8. The module does not support a maintenance interface or role.
9. The module does not support manual key entry.
10. The module does not have any external input/output devices used for entry/output of data.
11. The module only enters/outputs CSPs via the API, or through the seeding mechanism (callback from the OE provides entropy, e.g., /dev/random).
12. The module does not output intermediate key values over the physical boundary.
13. The module checks XTS keys pairs for duplicates as per IG A.9.

8.2 Operator-Enforced Security Rules

The following rules must be followed by the operator, as they cannot be enforced by the module itself.

1. The user or operator must not share critical security parameters between Approved mode and non-Approved mode algorithms or functions.
2. The user or operator, when implementing other standards, such as Common Criteria Mobile Device Fundamentals 3.0, may need to perform cryptographic operations with this module to create KEKs via DRBG or other needed protection mechanisms using specific algorithms, such as PBKDF. Please refer to those other standards for specifics.
3. The user or operator of the module must enforce all required locking mechanisms for code re-entrancy into the module, if multi-threading is desired.

9 Secure Distribution and Operation

The Citrix FIPS Cryptographic Module is intended for sole use by Citrix Systems, Inc. and its employees. As a result, the module is only made available to Citrix personnel through a Citrix-owned secure internal server. Only authorized employees have access to the module.

The FIPS 140-2 required power-on self-tests execute automatically without user intervention upon initialization of the module. Once self-tests have completed successfully, the module is operating in the Approved mode of operation.

10 References and Definitions

The following standards are referred to in this Security Policy.

Table 14 – References

Abbreviation	Full Specification Name
FIPS 140-2	Security Requirements for Cryptographic Modules, May 25, 2001
SP800-131A	Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths, January 2011
SP800-132	Recommendation for Password-Based Key Derivation, Part 1: Storage Applications
SP800-135-rev1	Recommendation for Existing Application-Specific Key Derivation Functions
SP800-38A	Recommendation for Block Cipher Modes of Operation
SP800-38F	Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping
SP800-56A and SP800-56A-rev2	Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography
SP800-56B-rev1	Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography
FIPS 186-4	Digital Signature Standard (DSS)