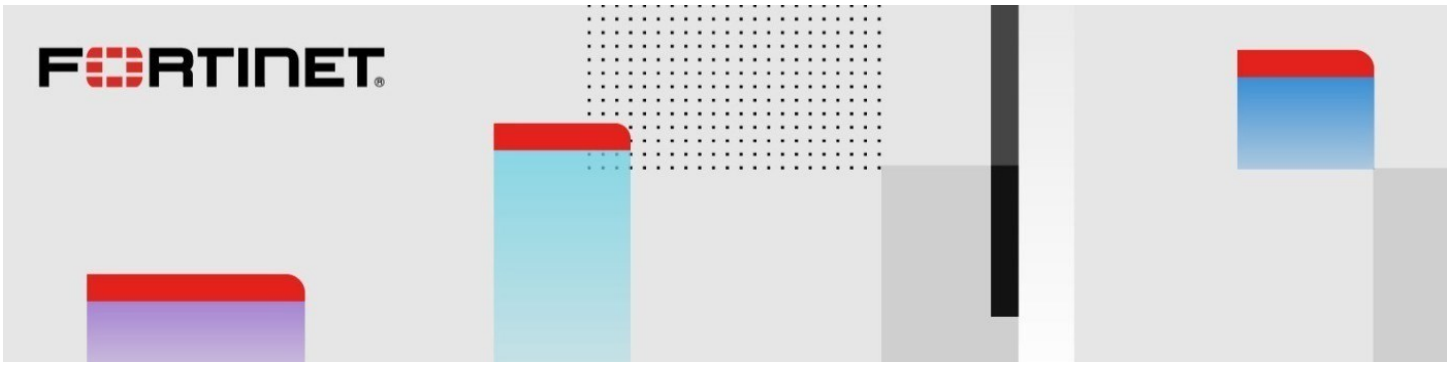


FIPS 140-3 Non-Proprietary Security Policy

FortiClient Crypto Library

Document Version:	4.1
Publication Date:	Thursday, October 3, 2024
Software Version:	7.0.2

Fortinet Technologies Inc.

FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO GUIDE
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/support-and-training/training.html

NSE INSTITUTE
https://training.fortinet.com

FORTIGUARD CENTER
https://fortiguard.com/

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

FortiClient Crypto Library FIPS 140-3 Security Policy

04-708-802275-20241003

TABLE OF CONTENTS

1.0 General

1.1. Overview

This document is a FIPS 140-3 Security Policy for Fortinet's FortiClient Crypto Library, version 7.0.2. This policy describes how the FortiClient Crypto Library (hereafter referred to as the 'module') meets the FIPS 140-3 security requirements and how to operate the modules in a FIPS compliant manner. This policy was created as part of the FIPS 140-3 Level 1 validation of the module.

The Federal Information Processing Standards Publication 140-3 - *Security Requirements for Cryptographic Equipment* (FIPS 140-3) details the United States Federal Government requirements for cryptographic equipment. Detailed information about the FIPS 140-3 standard and validation program is available on the NIST (National Institute of Standards and Technology) website at https://csrc.nist.gov/projects/cryptographic-module-validation-program.

1.2. Security Levels

The module meets the overall requirements for a FIPS 140-3 Level 1 validation

.

Table 1: Security Levels

ISO/IEC 24759 Section 6	FIPS 140-3 Section Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	N/A
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	N/A
	Overall	1

1.3. Additional Information

This policy deals specifically with operation and implementation of the modules in the technical terms of the FIPS 140-3 standard and the associated validation program. Other Fortinet product manuals, guides and technical notes can be found at the Fortinet technical documentation website at https://docs.fortinet.com.

Additional information on the entire Fortinet product line can be obtained from the following sources:

- Find general product information in the product section of the Fortinet corporate website at https://www.fortinet.com/products.
- Find on-line product support for registered products in the technical support section of the Fortinet corporate website at https://support.fortinet.com/.
- Find contact information for technical or sales related questions in the contacts section of the Fortinet corporate website at https://www.fortinet.com/contact.
- Find security information and bulletins in the FortiGuard Center of the Fortinet corporate website at https://wwww.fortiguard.com.

2.0 Cryptographic Module Specification

2.1. Module Description

2.1.1. Description

The FortiClient Crypto Library (hereafter referred to as the module) is a software cryptographic module that provides cryptographic services for FortiClient, Fortinet's endpoint security application. FortiClient and the module are designed to execute on a general purpose computer (GPC) hardware platform. The module has no physical characteristics and relies on the physical characteristics of the GPC on which it runs.

The module is a component of the FortiClient software and requires the following:

- A commercially available, general purpose, x86 compatible computer
- Windows 10
- Araneus Alea II entropy token
- Microsoft EMS server

The purpose of the module is providing cryptographic services for FortiClient.

2.1.2. Purpose or Use

This module provides cryptographic services for FortiClient, Fortinet's endpoint security application for Microsoft Windows.

2.1.3. Module Type

This module is a software cryptographic module.

2.1.4. Module Embodiment

The module is classified as a multi-chip standalone cryptographic module.

2.1.5. Module Characteristics

Please refer to Section 2 of the Security Policy.

2.1.6. Cryptographic Boundary

The module's cryptographic boundary encompasses the following FortiClient software binaries:

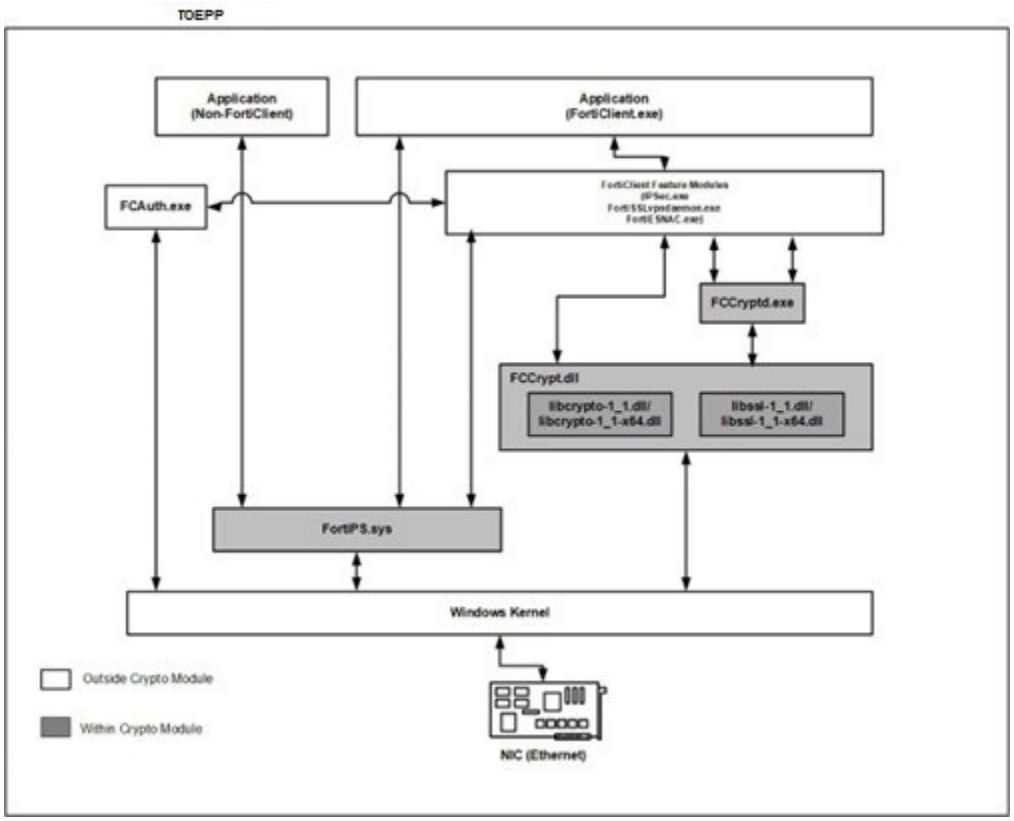
- FCCryptd.exe
- FCCrypt.dll, which serves as a wrapper for:
    - libcrypto-1_1.dll (32bit version) or libcrypto-1_1.dll-x64.dll (64bit version)
    - libssl-1_1.dll (32bit version) or libssl-1_1-x64.dll (64bit version)
- FortIPS.sys

2.1.7. Tested Operational Environment's Physical Perimeter

The TOEPP refers to the physical perimeter of the chassis of the general-purpose computer (GPC) on which the module is running.

2.1.8. Diagram, Schematic, or Photograph

Figure 1: Physical Perimeter of the TOEPP



The entire box in Figure 1 represents the TOEPP, while the gray boxes within the TOEPP represent the cryptographic boundary. For details on the cryptographic boundary and the Tested Operational Environment's Physical Perimeter, please refer to sections 2.1.6 and 2.1.7, respectively.

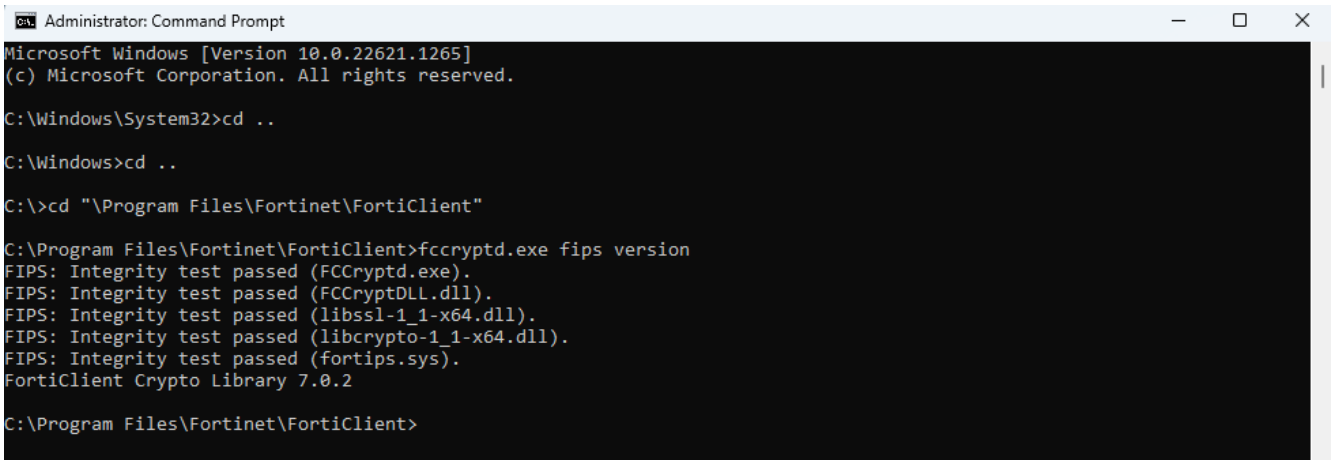
2.2. Version Information

The validated FortiClient Crypto Library version is 7.0.2. To verify the installed version, open a Windows Command Prompt window as an administrator and run the following command:

- `fccryptd.exe fips version`

The command runs the software integrity test and displays the module version as shown in the following image.

Figure 2: Module Version Output



2.3. Operating Environments

2.3.1. Hardware OEs

Not Applicable. FortiClient is a level 1 Software Module.

2.3.2. Software, Firmware, Hybrid OEs

The FortiClient Crypto Library module is a software module and is considered as a modifiable OE. The FortiClient module was tested on the following platform(s).

Table 2: Operating Environments – Software/Firmware/Hybrid

Operating System (Guest OS if Hypervisor)	Hardware Platform	Processor(s)	PAA/PAI
Microsoft Windows 10 (64-bit)	Dell XPS 8700	Intel Core i7-4770	N/A

2.3.3. Executable Code List

Table 3: Executable Code Sets – Software/Firmware/Hybrid

Package/File Names	Software/Firmware Version	Integrity Test Implemented
Fccryptd.exe	Software version: 7.0.2	RSA-3072
Fccryptd.dll	Software version: 7.0.2	RSA-3072
Fortips.sys	Software version: 7.0.2	RSA-3072

2.3.4. Vendor Affirmed Operating Environments

Fortinet vendor affirms the following operating environments.

Note: 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 OE which is not listed on the validation certificate.

Table 4: Vendor Affirmed Operating Environments

Operating System	Hardware Platform
Microsoft Windows 10	GPC with x86 or x86-64 bit processor
Microsoft Windows 11	GPC with x86 or x86-64 bit processor

2.4. Excluded Components

None.

2.5. Modes of Operation

2.5.1. Modes List and Description

The module does not support multiple modes of operation. The validated configuration of the module meets the FIPS 140-3 requirements by default once the startup procedures described in section 11.1 have been performed - i.e. the module has no approved mode that can be enabled or disabled by the user or local administrator and is always in the approved mode of operation.

The module assumes Microsoft EMS server usage for FortiClient deployment package creation, providing centralized management and deployment services, and is included in the FortiClient software package. The FortiClient deployment package is created on the EMS server with "FIPS" selected as an option and then deployed to the client PC.

Table 5: Modes of Operation

Name	Description	FIPS	Status Indicator
FIPS Deployed Package	The module is deployed as a FIPS package on the client PC via EMS	Approved	Run the command: fccryptd.exe fips version

2.5.2. Mode Change Instructions

The module only supports one mode of operation.

2.5.3. Degraded Mode

The module does not support a degraded mode of operation.

2.6. Algorithms

2.6.1. Approved Algorithms

Table 6: Approved Algorithms for Fccrypd.exe

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Strength(s)	Use / Function
A3419	AES-CBC FIPS PUB 197	CBC	Key Length: 128, 192, 256	Encryption/Decryption
A3419	AES-ECB FIPS PUB 197	ECB	Key Length: 128	Encryption/ Decryption
A3419	Counter DRBG SP 800-90Arev1	Counter-Based	Prediction Resistance: No Derivation Function Enabled: Yes Mode: AES-CBC (256-bit)	Random Bit Generation
A3419	ECDSA KeyGen (FIPS186-4) FIPS 186-4	ECDSA KeyGen	Curve: P-256, P-384, P-521	Key Pair Generation
A3419	ECDSA KeyVer (FIPS186-4) FIPS 186-4	ECDSA KeyVer	Curve: K-233, P-256, P-384, P-521	Public Key Validation
A3419	ECDSA SigGen (FIPS186-4) FIPS 186-4	ECDSA SigGen	Curve: P-256, P-384, P-521 Hash Algorithms: SHA2-256, SHA2- 384, SHA2-512	Digital Signature Generation
A3419	ECDSA SigVer (FIPS186-4) FIPS 186-4	ECDSA SigVer	Curve: P-256, P-384, P-521 Hash Algorithms: SHA2-256, SHA2- 384, SHA2-512	Digital Signature Verification
A3419	HMAC-SHA-1 FIPS PUB 198-1	SHA-1	MAC Length: 160 Key Length: 112-1024	Message Authentication
A3419	HMAC-SHA2-256 FIPS PUB 198-1	SHA2-256	MAC Length: 256 Key Length: 112-1024	Message Authentication
A3419	HMAC-SHA2-384	SHA2-384	MAC Length: 384	Message Authentication

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Strength(s)	Use / Function
	FIPS PUB 198-1		Key Length: 112-1024	
A3419	HMAC-SHA2-512 FIPS PUB 198-1	SHA2-512	MAC Length: 512 Key Length: 112-1024	Message Authentication
A3419	HMAC-SHA3-512 FIPS PUB 198-1	SHA3-512[1]	MAC Length: 512 Key Length: 112-1024	Message Authentication
A3419	KAS-ECC-SSC SP 800-56Arev3	Scheme: ephemeralUnified	P-256, P-384, P-521	Shared Secret Computation
A3419	KAS-FFC-SSC Sp800-56Arev3	Scheme: dhEphem	ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192	Shared Secret Computation
A3419	PBKDF SP 800-132 (Option 1b)	PBKDF[2]	HMAC-SHA-1 Password Length: 8-105 Key Length: 160 Iteration Count: 1000[3]	Password-Based Key Derivation
A3419	RSA KeyGen (FIPS186-4) FIPS 186-4	RSA KeyGen	Modulo Sizes: 2048, 3072, 4096	Key Pair Generation
A3419	RSA SigGen (FIPS186-4) FIPS 186-4	RSA SigGen	Modulo Sizes: 2048, 3072, 4096	Digital Signature Generation
A3419	RSA SigVer (FIPS186-4) FIPS 186-4	RSA SigVer	Modulo Sizes: 1024 (Legacy)[4], 2048, 3072, 4096	Digital Signature Verification
A3419	SHA-1 FIPS 180-4	SHA-1	Message Length: 0-65528 Increment 8	Message Digest
A3419	SHA2-256 FIPS 180-4	SHA2-256	Message Length: 0-65528 Increment 8	Message Digest
A3419	SHA2-384 FIPS 180-4	SHA2-384	Message Length: 0-65528 Increment 8	Message Digest
A3419	SHA2-512 FIPS 180-4	SHA2-512	Message Length: 0-65528 Increment 8	Message Digest
A3419	SHA3-512 FIPS 180-4	SHA3-512[1]	Message Length: 0-65528 Increment 8	Message Digest
A3419	Safe Primes Key Generation SP 800-56Arev3, Appendix D	MODP	ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192	Prime Generation

[1] Note: SHA3-256, SHA3-384, HMAC SHA3-256 and HMAC SHA3-384 are CAVP tested, but not used by any service within the module.
[2] Note: Keys derived from passwords, as described in SP 800-132, may only be used in storage applications.
[3] Note: The recommended iteration count of 1000 is used, as per SP 800-132.
[4] Legacy usage only. These legacy algorithms can only be used on data that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M

Table 7: Approved Algorithms for FortIPS.sys

CAVP Cert	Algorithm and Standard	Mode / Method	Description / Key Size(s) / Strength(s)	Use / Function
A4688	AES-CBC FIPS PUB 197	CBC	Key Length: 128, 192, 256	Encryption/Decryption
A4688	HMAC-SHA2-256 FIPS PUB 198-1	SHA2-256	MAC Length: 256 Key Length: 112-1024	Message Authentication
A4688	HMAC-SHA2-384 FIPS PUB 198-1	SHA2-384	MAC Length: 384 Key Length: 112-1024	Message Authentication
A4688	HMAC-SHA2-512 FIPS PUB 198-1	SHA2-512	MAC Length: 512 Key Length: 112-1024	Message Authentication
A4688	SHA2-256 FIPS 180-4	SHA2-256	Message Length: 0-65528 Increment 8	Message Digest
A4688	SHA2-384 FIPS 180-4	SHA2-384	Message Length: 0-65528 Increment 8	Message Digest
A4688	SHA2-512 FIPS 180-4	SHA2-512	Message Length: 0-65528 Increment 8	Message Digest

2.6.2. Vendor Affirmed Algorithms

Table 8: Vendor Affirmed Algorithms

Algorithm	Algorithm Properties	OE	Reference
CKG (Asymmetric)	Key Type: Asymmetric	Microsoft Windows 10 running on a Dell XPS 8700 with an Intel Core i7-4770(Haswell)	Section 5.1 and 5.2 of SP 800-133rev2 IG D.H
CKG (Symmetric)	Key Type: Symmetric	Microsoft Windows 10 running on a Dell XPS 8700 with an Intel Core i7-4770(Haswell)	Section 4, 6.1, 6.2.3 of SP 800-133rev2 IG D.H

2.6.3. Non-Approved, Allowed Algorithms

None.

2.6.4. Non-Approved, Allowed Algorithms with No Security Claimed

None.

2.6.5. Non-Approved, Not Allowed Algorithms

None.

2.7. Security Function Implementations

Table 9: Security Function Implementations

Name	Type	Description	SF Properties	Algorithms	Algorithm Properties
AES for Symmetric Encryption/Decryption (FCT Library)	BC-UnAuth	Encryption/Decryption	Publication: SP 800-38A	AES-CBC (FortiClient 7.0.2 Crypto Library for Windows) AES-ECB (FortiClient 7.0.2 Crypto Library for Windows)	Key Sizes: AES-CBC: 128, 192, 256 AES-ECB: 128
AES for Symmetric Encryption/Decryption (IPSec Driver)	BC-UnAuth	Encryption/Decryption	Publication: SP 800-38A	AES-CBC (FortiClient 7.0.2 IPSec Crypto Library for Windows)	Key Sizes: 128, 192, 256
DRBG	DRBG	Deterministic Random Bit Generation	Publication: SP 800-90Arev1	Counter DRBG (FortiClient 7.0.2 Crypto Library for Windows)	256-bit AES-CBC Prediction Resistance: No Derivation Function: Yes
ECDSA for Key Generation	AsymKeyPair-KeyGen	Key Generation of the ECDSA Private Key and ECDSA Public Key	Publication: FIPS 186-4	ECDSA KeyGen (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) CKG (Asymmetric) (FortiClient 7.0.2 Crypto Library for Windows)	Curve: P-256, P-384, P-521
ECDSA for Key Verification	AsymKeyPair-KeyVer	ECDSA Public Key Verification	Publication: FIPS 186-4	ECDSA KeyVer (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows)	Curve: K-233, P-256, P-384, P-521
ECDSA for Signature Generation	DigSig-SigGen	ECDSA Signature Generation for the ECDSA Private Key	Publication: FIPS 186-4	ECDSA SigGen (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2	Curve: P-256, P-384, P-521

Name	Type	Description	SF Properties	Algorithms	Algorithm Properties
				Crypto Library for Windows)	
ECDSA for Signature Verification	DigSig-SigVer	ECDSA Signature Verification for the ECDSA Public Key	Publication: FIPS 186-4	ECDSA SigVer (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows)	Curve: P-256, P-384, P-521
HMAC for Keyed Hash Operations (FCT Library)	MAC	Message Authentication	Publication: FIPS 198-1	HMAC-SHA-1 (FortiClient 7.0.2 Crypto Library for Windows) HMAC-SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) HMAC-SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) HMAC-SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows) HMAC-SHA3-512 (FortiClient 7.0.2 Crypto Library for Windows) SHA-1 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows) SHA3-512	MAC Lengths: 160, 256, 384, 512

Name	Type	Description	SF Properties	Algorithms	Algorithm Properties
				(FortiClient 7.0.2 Crypto Library for Windows)	
HMAC for Keyed Hash Operations (IPSec Driver)	MAC	Message Authentication	Publication: FIPS 198-1	HMAC-SHA2-256 (FortiClient 7.0.2 IPSec Crypto Library for Windows) HMAC-SHA2-384 (FortiClient 7.0.2 IPSec Crypto Library for Windows) HMAC-SHA2-512 (FortiClient 7.0.2 IPSec Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 IPSec Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 IPSec Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 IPSec Crypto Library for Windows)	MAC Lengths: 256, 384, 512
ECC-SSC	KAS-SSC	ECDH Shared Secret Computation	Publication: SP 800-56Arev3	KAS-ECC-SSC Sp800-56Arev3 (FortiClient 7.0.2 Crypto Library for Windows)	Curves: P-256, P-384, P-521
FFC-SSC	KAS-SSC	DH Shared Secret Computation	Publication: SP 800-56Arev3	KAS-FFC-SSC Sp800-56Arev3 (FortiClient 7.0.2 Crypto Library for Windows)	Safe Primes: MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192
PBKDF	PBKDF	Password-Based Key Derivation	Publication: SP 800-132	PBKDF (FortiClient 7.0.2 Crypto Library for Windows) HMAC-SHA-1 (FortiClient 7.0.2 Crypto Library for Windows) SHA-1 (FortiClient 7.0.2 Crypto Library for Windows)	HMAC-SHA-1 Password Length: 8-105
RSA for Key Generation	AsymKeyPair- KeyGen	Key Pair Generation of the RSA Private Key and RSA Public Key	Publication: FIPS 186-4	RSA KeyGen (FIPS186-4) (FortiClient 7.0.2 Crypto Library for	Modulo Sizes: 2048, 3072, 4096

Name	Type	Description	SF Properties	Algorithms	Algorithm Properties
				Windows) CKG (Asymmetric) (FortiClient 7.0.2 Crypto Library for Windows)	
RSA for Signature Generation	DigSig-SigGen	RSA Signature Generation for the RSA Private Key	Publication: FIPS 186-4	RSA SigGen (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows)	Modulo Sizes: 2048, 3072, 4096
RSA for Signature Verification	DigSig-SigVer	RSA Signature Verification for the RSA Public Key	Publication: FIPS 186-4	RSA SigVer (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) SHA-1 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows)	Modulo Sizes: 1024 [Legacy], 2048, 3072, 4096
SHA for Message Digest (FCT Library)	SHA	Message Digest	Publication: FIPS 180-4, FIPS 202	SHA-1 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 Crypto Library for Windows)	Message Length: 0-65528 Increment 8

Name	Type	Description	SF Properties	Algorithms	Algorithm Properties
				SHA3-512 (FortiClient 7.0.2 Crypto Library for Windows)	
SHA for Message Digest (IPSec Driver)	SHA	Message Digest	Publication: FIPS 180-4, FIPS 202	SHA2-256 (FortiClient 7.0.2 IPSec Crypto Library for Windows) SHA2-384 (FortiClient 7.0.2 IPSec Crypto Library for Windows) SHA2-512 (FortiClient 7.0.2 IPSec Crypto Library for Windows)	Message Length: 0-65528 Increment 8
RSA and SHA for Integrity Test	DigSig-SigGen, SHA	RSA and SHA for the pre-operational integrity tests.	Publication: FIPS 198-1, FIPS 186-4	RSA SigVer (FIPS186-4) (FortiClient 7.0.2 Crypto Library for Windows) SHA2-256 (FortiClient 7.0.2 IPSec Crypto Library for Windows)	Modulo Size: 3072
Safe Prime Key Generation	AsymKeyPair- SafePri	Key Generation for all module services that utilize KAS-FFC-SSC	Publication: SP 800-56Arev3	Safe Primes Key Generation (FortiClient 7.0.2 Crypto Library for Windows)	Safe Primes: MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192

2.8. Algorithm Specific Information

Symmetric Keys:

The module generates symmetric keys as per section 6.2.3 of SP 800-133rev2 using PBKDF2 (Password Based Key Derivation Function).

PBKDF:

PBKDF2 uses HMAC-SHA-1 and a 128-bit salt to derive keys, which are used solely for storage purposes, in accordance with SP 800-132 and IG D.N.

The probability of guessing a password at random (S) is equal to the Character Pool Size (C) raised to the power of the number of characters in the password (N). The Character Pool Size (C) is 95, as the password can contain lowercase (a- z) [26], uppercase (A-Z) [26], digits (0-9) [10], and special characters (!"$,) [33]. The minimum password length is 8 characters and is enforced. However, there is no constraint on the number of times a character is used from the character pool. Therefore, the probability of guessing a password (S) is between 95^8 and 95^105 – i.e. using an 8 character password the probability of guessing the password is 1 in 6,634,204,312,890,625.

Asymmetric Keys:

ECDSA and RSA key generation are performed using the approved Counter DRBG provided in accordance with Section 4 of SP 800-133rev2. For ECDSA, the elliptic curve is selected with appropriate base order to meet SP 800-57 Part 1 requirements.

2.9. RNG and Entropy

2.9.1. Entropy Information

The module uses an entropy token (Araneus Alea II) to seed the DRBG during the modules' boot process and to periodically reseed the DRBG. The entropy token is not within the TOEPP (per IG 9.5.A) of the module; therefore, no assurance can be made regarding the correct operation of the entropy token, nor is there a guarantee of the stated entropy.

Entropy Strength

The entropy loaded into the approved AES-256 bit DRBG is 256 bits

Reseed Period

The DRBG is seeded from the entropy token during the boot process and then reseeded periodically. The default reseed period is once every 24 hours (1440 minutes) and is configurable (1 to 1440 minutes). The entropy token must be installed to complete the boot process and to reseed the DRBG.

Table 10: Entropy Sources

Entropy Sources	Minimum Number of Bits of Entropy	Details
Araneus Alea II USB token	256	The Entropy Input String is 256 bits and expected to provide Full Entropy. The module requests 10240 bits of data from the token and passes it through SHA2-256 (A3419) before seeding the DRBG with the 256-bit output. This 256-bit input is expected to provide 256-bits of entropy from the external token. No assurance of the minimum strength of generated SSPs (e.g., keys) IG 9.3.A Scenario 1c

2.9.2. RNG Information

The module uses the approved Counter DRBG as per section 10.2.1 of SP 800-90Arev1, which is seeded using the entropy source described in section 2.9.1 of the Security Policy to generate random numbers.

2.10. Key Generation

The module complies with Section 4, 5, 6.1, and 6.2.3 of SP 800-133rev2, where the module uses its Approved DRBG to generate random values and seeds used for symmetric and asymmetric key generation. The resulting symmetric key or generated seed is an unmodified output from the DRBG.

2.11. Key Establishment

2.11.1. Key Agreement Information

For KAS-ECC-SSC, elliptic-curve groups approved for use by the key-establishment schemes specified in SP 800-56Arev3 Appendix D are used.

For KAS-FFC-SSC, an approved safe prime group listed in SP 800-56A rev3 Appendix D is used.

2.11.2. Key Transport Information

None.

2.12. Industry Protocols

None.

2.13. Design and Rules

Refer to section 11.1 ‘Startup Procedures’ for Installation process.

2.14. Initialisation

Refer to section 11.1 ‘Startup Procedures’ for Installation process.

2.15. Additional Information

None.

3.0 Cryptographic Module Interfaces

3.1. Ports and Interfaces

FortiClient is a software module and therefore does not have physical interfaces. FortiClient's logical interfaces and the types of data passed over the interfaces are described below.

Table 11: Ports and Interfaces

Physical Port	Logical Interface	Data that Passes Over the Port/Interface
N/A	Data Input	API input parameters for data- plain text and cipher text, digital signatures, public keys
N/A	Data Output	API output parameters for data- encrypted or decrypted, digital signatures, hashes
N/A	Control Input	API input commands from FortiClient application, entropy input
N/A	Status Output	API return values to the FortiClient application

Note: Data I/O occurs between the FortiClient application and the crypto module. All the data output via data output interface is inhibited when the module is performing pre-operational tests, zeroization, or enters an error state. The module does not support a control output interface

3.2. Trusted Channel Specification

Not Applicable (Level 1 Module)

3.3. Control Interface Not Inhibited

Not Applicable. Entering the error mode shuts down the module (and the FortiClient application) completely. See 10.4 (Error States) for more information.

3.4. Additional Information

None.

4.0 Roles, Services, and Authentication

4.1. Authentication Methods

The FortiClient module does not claim any authentication methods as it is a FIPS 140-3 level 1 software module.

4.2. Roles

The module provides the following roles:

- Crypto Officer (CO)

A User with Windows administrative privileges assumes the "Crypto Officer Role". Multiple accounts can be logged in to the Windows PC, but only one account (and therefore only one role) can be active at a time through Windows user switching.

The module does not provide a Maintenance role.

Table 12: Roles

Name	Type	Operator Type	Authentication Methods
Crypto-Officer	Role	CO	N/A

4.3. Approved Services

The following tables detail the types of approved services available to each role in approved mode of operation, the types of access for each role and the Keys or CSPs they affect.

Generate	G
Read Access	R
Write Access	W
Execute Access	E
Zeroize	Z

Table 13: Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Function Implementations	Roles	Roles SSP Access
Asymmetric Key Generation	RSA and ECDSA Keys are Generated	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Asymmetric Key Generated	ECDSA for Key Generation ECDSA for Key Verification RSA for key Generation	CO	ECDSA Public Key: G, R ECDSA Private Key: G, R RSA Public Key: G, R RSA Private Key: G, R
Backup / Restore	Back and Restore module configuration	Successful completion of service. IG 2.4.C, Scenario 2	FortiClient GUI	Module Configuration Backed Up or Restored	N/A	CO	N/A
Derive Key Via PBKDF2	The module derives key using user password	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Key is Derived	PBKDF SHA for Message Digest (FCT Library) HMAC for Keyed Hash Operations (FCT Library)	CO	HMAC Keys: R, G PBKDF2 Derived Keys: E User Password: W, E
Execute Manual Self-Tests	Manual Self-Tests are executed	Successful completion of service. IG 2.4.C, Scenario 2	Windows CLI Command	Self-Test results displayed in Command Prompt Window	RSA and SHA for Integrity Test	CO	N/A
Generate Message Digest	Message Digests are generated using SHA algorithms	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Message Digest Generated	SHA for Message Digest (FCT Library) SHA for Message Digest (IPSec Driver)	CO	N/A
Generate Keyed Hash	Keyed hash is generated to provide message authentication	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Keyed Hash Generated	HMAC for Keyed Hash Operations (FCT Library) HMAC for Keyed Hash Operations (IPSec Driver)	CO	HMAC Keys: E, G
Install	FortiClient module installed on the GPC	Successful completion of service. IG 2.4.C, Scenario 2	Windows GUI	FortiClient installed on GPC	N/A	CO	N/A
Initialize	FortiClient module installed on the GPC	Successful completion of service. IG 2.4.C, Scenario 2	Windows GUI	Module Initialized as part of the FortiClient startup	N/A	CO	Software Integrity Key: E
Key Agreement	KAS-FFC-SSC and KAS-ECC-SSC are generated	Successful completion of service. IG 2.4.C,	API Call Parameters	Key Agreement Performed	ECC-SSC ECDSA for Key Generation FFC-SSC	CO	ECDH Shared Secret Keys: G, E ECDH Private

Name	Description	Indicator	Inputs	Outputs	Security Function Implementations	Roles	Roles SSP Access
	using FFC DH and ECC respectively	Scenario 2			Safe Prime Key Generation		Keys: G, E ECDH Public Keys: G, E DH Private Keys: G, E DH Public Keys: G, E DH Shared Secret Keys: G, E
Perform Digital Signature	Digital Signature generated using RSA and ECDSA asymmetric keys	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Digital Signature Performed	ECDSA for Signature Generation ECDSA for Signature Verification RSA for Signature Generation RSA for Signature Verification	CO	ECDSA Public Key: G, E ECDSA Private Key: G, E RSA Public Key: G, E RSA Private Key: G, E
Random Bit Generation	A counter-based random bit is generated using the Araneus Alea II entropy token.	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Random bits generated	DRBG	CO	Entropy Input String: W, E DRBG Seed: G, E DRBG Output: G, R DRBG 'V' Values: G, E DRBG 'Key' Value: G, E
Show Module Status	Module Status Displayed	Successful completion of service. IG 2.4.C, Scenario 2	FortiClient GUI	Show Approved mode (FortiClient Console)	N/A	CO	N/A
Show Version Information	Crypto Module version shown	Successful completion of service. IG 2.4.C, Scenario 2	Windows CLI command	Show FortiClient crypto module version	N/A	CO	N/A
Symmetric Key Encryption/ Decryption	AES Keys are used to encrypt plaintext and decrypt ciphertext	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Perform Symmetric key Encryption/ Decryption	AES for Symmetric Encryption/Decryption (FCT Library) AES for Symmetric Encryption/Decryption (IPSec Driver)	CO	AES Keys: W, E
Symmetric Key Generation	AES Keys are Generated	Successful completion of service. IG 2.4.C, Scenario 2	API Call Parameters	Symmetric Key Generated	AES for Symmetric Encryption/Decryption (FCT Library) AES for Symmetric Encryption/Decryption (IPSec Driver) DRBG CKG (Symmetric)	CO	AES Keys: G, E
Zeroize Keys	All Sensitive Security	Successful completion	GUI/CLI	Stored SSPs zeroized and	N/A	CO	All Keys: Z

Name	Description	Indicator	Inputs	Outputs	Security Function Implementations	Roles	Roles SSP Access
	Parameters are zeroized	of service. IG 2.4.C, Scenario 2		FortiClient restored to factory settings.			

4.4. Non-Approved Services

None.

4.5. External Software/Firmware Loaded

None. The module does not use or require external software or firmware components.

4.6. Bypass Actions and Status

None. The module does not implement a bypass mode.

4.7. Cryptographic Output Actions and Status

None. The module does not support self-initiated cryptographic output.

4.8. Additional Information

None.

5.0 Software/Firmware Security

5.1. Integrity Techniques

The module uses an RSA PKCS1.5 3072 with SHA2-256 (Cert. #3419) signatures to verify the integrity of its binaries. The integrity check runs automatically when FortiClient starts, but it can also be triggered manually (on demand) from the Windows command prompt using the following command :

```
fccryptd.exe fips kat integrity
```

If the tests passes, the results are output to the command prompt window.

If the tests fail, FortiClient will log the failure and shuts down. See Section 10.4 (Error States) for more information.

5.2. Initiate on Demand

Refer to section 10.5 (Operator Initiation Self-Test) for more information.

5.3. Open Source Parameters

Not Applicable, the module is delivered as a compiled executable.

5.4. Non-Reconfigurable Memory End of Life Procedures

Not Applicable.

5.5. Additional Information

None.

6.0 Operational Environment

6.1. Operational Environment Type and Requirements

The FortiClient module operates in a modifiable operational environment and runs on a commercially available GPC. For specific information regarding the operational environment, please refer to section 2.3.1. There are no user-configurable settings for the module. For more details on the installation process, please refer to section 11.1.

6.2. Configuration Settings and Restrictions

None.

6.3. Additional Information

Since the operating environment (Windows) lets users install new software, the module falls under the category of a modifiable OE.

7.0 Physical Security

Not applicable. FortiClient is a level 1 software module.

8.0 Non-Invasive Security

Not Applicable.

9.0 Sensitive Security Parameters Management

9.1. Storage Areas

Table 14: Storage Areas

Name	Description	Persistence Type
Plaintext in SDRAM	System Memory	Dynamic

9.2. SSP Input/Output Methods

Table 15: SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
API Input	Application	Crypto Module Software	Plaintext	Manual	Electronic	N/A
API Output	Crypto Module Software	Application	Plaintext	Manual	Electronic	N/A
Entropy Input	Entropy Source (Araneus Alea II)	Crypto Module Software	Plaintext	Manual	Electronic	N/A

The module does not support manual SSP entry or intermediate key generation output.

9.3. SSP Zeroization Methods

As per SP IG 9.3 Resolution 2, Fortinet follows SSP procedural zeroization which is achieved through the following steps:

Table 16: SSP Zeroisation Methods

Method	Description	Rationale	Operator Initiation Capability
Procedural Zeroisation	1. Uninstalling the Module 2. Overwriting the mass storage 3. Power Cycling the Host Computer	In accordance with IG 9.7.B Resolution 2, successful completion of the procedural zeroisation itself serves as the implicit indicator that zeroisation is complete	Yes

9.4. SSPs

Table 17: SSPs (Part 1)

Name	Description	Size	Strength	Type	Generated By	Established By
AES Keys	Symmetric Key used to Encrypt and Decrypt Data	128, 192, 256 bits	128, 192, 256 bits	Symmetric Key Cryptography	Internal as per SP 800-133rev2 Section 6.1	N/A
DH Private Keys	Key Agreement and Key Establishment	2048 – 8192 bits	112 – 200 bits	Asymmetric Public Key	Internal as per SP 800-56Arev3	N/A
DH Public Keys	Key Agreement and Key Establishment	2048 – 8192 bits	112 – 200 bits	Asymmetric Public Key	Internal as per SP 800-56Arev3	N/A
DH Shared Secret Keys	Shared Secret Computation	2048 – 8192 bits	112 – 200 bits	Key Agreement	Internal as per SP 800-56Arev3	N/A
Entropy Input String	Input String from the Entropy Pool	256 bits	256 bits	Entropy	External	N/A
DRBG Seed	256 bit seed used by the DRBG	256 bits	256 bits	Seeding Material for the DRBG	Internal as per SP 800-90Arev1	N/A
DRBG Output	Random Numbers used in cryptographic algorithms	256 bits	256 bits	DRBG	Internal as per SP 800-90Arev1	N/A
DRBG 'V' Values	Internal State Value for the DRBG	256 bits	256 bits	DRBG	Internal as per SP 800-90Arev1	N/A
DRBG 'Key' Value	Internal State Value for the DRBG	256 bits	256 bits	DRBG	Internal as per SP 800-90Arev1	N/A
ECDH Private Keys	Key Agreement and Key Establishment	P-256, P-384, P-521 curves	128 – 256 bits	Asymmetric Private Key	Internal as per SP 800-56Arev3	N/A
ECDH Public Keys	Key Agreement and Key Establishment	P-256, P-384, P-521 curves	128 – 256 bits	Asymmetric Public Key	Internal as per SP 800-56Arev3	N/A
ECDH Shared Secret Keys	Shared Secret Computation	P-256, P-384, P-521 curves	128 – 256 bits	Key Agreement	Internal as per SP 800-56Arev3	N/A
ECDSA Public Key	Digital Signature Verification	P-256, P-384, P-521 curves	128 – 256 bits	Public Key Cryptography	Internal as per FIPS 186-4	N/A
ECDSA Private Key	Digital Signature Generation	P-256, P-384, P-521 curves	128 – 256 bits	Private Key Cryptography	Internal as per FIPS 186-4	N/A
HMAC Keys	Keyed Hash	160, 256, 384, 512 bits	>= 112 bits	Message Authentication	External	N/A
PBKDF2 Derived Keys	Key used to Derive Symmetric Keys	160 bits	160 bits	Key Derivation	Derived via SP 800-133rev2	N/A
RSA Public Key	Asymmetric Key used to Encrypt and Decrypt Data	2048, 3072 bits	112 – 128 bits	Public Key Cryptography	Internal as per FIPS 186-4	N/A
RSA Private Key	Asymmetric Key used to Encrypt and Decrypt Data	2048, 3072 bits	112 – 128 bits	Private Key Cryptography	Internal as per FIPS 186-4	N/A
Software Integrity Key	Key used to ensure the integrity and authenticity of configuration data	3072 bits	128 bits	Authentication	N/A	Pre-Loaded

Name	Description	Size	Strength	Type	Generated By	Established By
User Password	Input to PBKDF2 for Key Derivation	8 – 105 characters	8 – 105 characters	Key Derivation	N/A	N/A

Table 18: SSPs (Part 2)

Used By	Inputs/Outputs	Storage	Temporary Storage Duration	Zeroization	Category	Related SSPs
AES for Symmetric Encryption/Decryption (FCT Library) AES for Symmetric Encryption/Decryption (IPSec Driver)	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
FFC-SSC Safe Prime Key Generation	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	DH Public Keys: Paired With
FFC-SSC Safe Prime Key Generation	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	PSP	DH Private Keys: Paired With
FFC-SSC Safe Prime Key Generation	API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	DH Public Keys: Derived From DH Private Keys: Derived From
DRBG	Entropy Input	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
DRBG	N/A	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	DRBG Entropy Input: Derived From
N/A	Entropy Input	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
DRBG	N/A	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
DRBG	N/A	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
ECC-SSC ECDSA for Key Generation	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	ECDH Public Keys: Paired With
ECC-SSC ECDSA for Key Verification	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	PSP	ECDH Private Keys: Paired With

Used By	Inputs/Outputs	Storage	Temporary Storage Duration	Zeroization	Category	Related SSPs
ECC-SSC ECDSA for Key Generation	API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	ECDH Public Keys: Derived From ECDH Private Keys: Derived From
ECDSA for Signature Verification	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	PSP	ECDSA Private Key: Paired With
ECDSA for Signature Generation ECDSA for Key Generation	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	ECDSA Public Key: Paired With
HMAC for Keyed Hash Operations (FCT Library) HMAC for Keyed Hash Operations (IPSec Driver) SHA for Message Digest (IPSec Driver) SHA for Message Digest (FCT Library)	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	N/A
PBKDF HMAC for Keyed Hash Operations (FCT Library) SHA for Message Digest (FCT Library)	API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	CSP	User Password: Derived From
RSA for Signature Verification	API Input API Output	Plaintext in SDRAM	API Call Lifetime	Procedural Zeroization (See Section 9.3)	PSP	RSA Private Key: Paired With
RSA for Signature Generation RSA for Key Generation	API Input API Output	Plaintext in SDRAM	API Call Lifetime	API Call Lifetime	CSP	RSA Public Key: Paired With
RSA and SHA for Integrity Test	N/A	Plaintext in SDRAM	API Call Lifetime	API Call Lifetime	Non-SSP	N/A
PBKDF HMAC for Keyed Hash Operations (FCT Library) SHA for Message Digest (FCT Library)	API Input API Output	Plaintext in SDRAM	API Call Lifetime	API Call Lifetime	CSP	PBKDF2 Derived Keys: Derived By

10.0 Self-Tests

10.1. Pre-Operational Self-Tests

The cryptographic module's pre-operational self-tests provide the operator with assurance that no faults have been introduced that could hinder correct operation, as outlined in ISO/IEC 19790:2012, section 7.10.1. Upon successfully passing the self-test, the module logs a pass indicator message in cryptdlog.txt. Refer to section 10.6 for more information.

The module performs the startup configuration integrity test by performing the following steps:

- Upon first startup, fccryptd.exe calculates an HMAC using SHA2-256 for the configuration settings stored in the registry.
- The calculated HMAC value is then stored in the registry.
- When authorized configuration changes are made, fccryptd.exe recalculates and stores the new HMAC value.
- On subsequent startups, fccryptd.exe verifies the integrity by recalculating the HMAC and comparing it with the stored HMAC value.
- If the stored and recalculated HMAC values don't match, the configuration integrity test fails.

The configuration integrity test can be run on demand using **`fccrypt fips kat configuration`** at the command line.

The following is an example of a successful self-test:

```
FIPS: HMAC self-test passed
```

In the event that any of the self-tests fail, FortiClient logs an error indicator message for the specific test(s) in cryptdlog.txt. Refer to section 10.4 for examples and more information.

Table 19: Pre-Operational Self-Tests

Algorithm	Implementation	Test Properties	Test Method	Type	Indicator	Details
HMAC-SHA2-256	Fccryptd.exe (Cert. #A3419)	Configuration Integrity Test	KAT	Critical Functions Test	Pass Indicator Error Indicator	Verify
HMAC-SHA2-256	FortIPS.sys (Cert. #A4688)	Configuration Integrity Test	KAT	Critical Functions Test	Pass Indicator Error Indicator	Verify
SHA2-256 and RSA SigVer (FIPS 186-4)	Fccryptd.exe (Cert. #A3419)	RSA PKCS1.5 3072 and SHA2-256	KAT	Software Integrity Test	Pass Indicator Error Indicator	Sign and Verify

10.2. Conditional Self-Tests

Cryptographic Algorithm Self Tests (CASTs): The module performs self-tests on approved algorithms, as required by FIPS 140-3 Implementation Guidance (IG) section 10.3.A. If any self-test fails, FortiClient logs the specific test(s) in cryptdlog.txt (refer to Section 10.6 for more information).

The module also performs Critical Functions Test (CFT), Continuous Pair-wise Consistency Test (CPCT), as outlined in FIPS 140-3 IG section 10.3.A.

Table 20: Conditional Self-Tests

Algorithm	Implementation	Test Properties	Test Method	Type	Indicator	Details	Condition
AES-CBC	Fccryptd.exe (Cert. #A3419)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Encrypt	Boot Up
AES-CBC	Fccryptd.exe (Cert. #A3419)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Decrypt	Boot Up
AES-CBC	FortIPS.sys (Cert. #A4688)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Encrypt	Boot Up
AES-CBC	FortIPS.sys (Cert. #A4688)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Decrypt	Boot Up
AES-ECB	Fccryptd.exe (Cert. #A3419)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Encrypt	Boot Up
AES-ECB	Fccryptd.exe (Cert. #A3419)	Key Size: 128	KAT	CAST	Pass Indicator Error Indicator	Decrypt	Boot Up
Counter DRBG	Fccryptd.exe (Cert. #A3419)	AES-CBC (256 bit)	Health Tests	CAST	Pass Indicator Error Indicator	Instantiate, Generate, and Reseed	Boot Up
Counter DRBG	Fccryptd.exe (Cert. #A3419)	AES-CBC (256 bit)	KAT	CAST	Pass Indicator Error Indicator	Prediction Resistance: No Derivation Function: Yes	Boot Up
DH Key Assurance Check	Fccryptd.exe (Cert. #A3419)	KAS-FFC-SSC	CFT	CFT	Pass Indicator Error Indicator	FFC Full validation (Section 5.6.2.3.1 of SP 800-56Arev3)	Upon Generation of a DH Public Key
ECDH Key Assurance Check	Fccryptd.exe (Cert. #A3419)	KAS-ECC-SSC	CFT	CFT	Pass Indicator Error Indicator	ECC Full validation (Section 5.6.2.3.3 SP 800-56Arev3)	Upon Generation of a ECDH Public Key

Algorithm	Implementation	Test Properties	Test Method	Type	Indicator	Details	Condition
ECDSA KeyGen (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	P-256	PCT	PCT	Pass Indicator Error Indicator	Key Pair Generation	Prior to the First Operational Use.
ECDSA SigGen (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	P-256	KAT	CAST	Pass Indicator Error Indicator	Sign	Boot Up
ECDSA SigVer (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	P-256	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-256	Fccryptd.exe (Cert. #A3419)	HMAC-SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-384	Fccryptd.exe (Cert. #A3419)	HMAC-SHA2-384	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-512	Fccryptd.exe (Cert. #A3419)	HMAC-SHA2-512	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA3-256	Fccryptd.exe (Cert. #A3419)	HMAC-SHA3-256	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA3-384	Fccryptd.exe (Cert. #A3419)	HMAC-SHA3-384	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA3-512	Fccryptd.exe (Cert. #A3419)	HMAC-SHA3-512	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-256	Fortips.sys (Cert. #A4688)	HMAC-SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-384	Fortips.sys (Cert. #A4688)	HMAC-SHA2-384	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
HMAC-SHA2-512	Fortips.sys (Cert. #A4688)	HMAC-SHA2-512	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
KAS-ECC-SSC[5]	Fccryptd.exe (Cert. #A3419)	P-256	KAT	CAST	Pass Indicator Error Indicator	Verify computation of shared secret Z in	Boot Up

5

Algorithm	Implementation	Test Properties	Test Method	Type	Indicator	Details	Condition
						Ephemeral Unified scheme	
KAS-FFC-SSC	Fccryptd.exe (Cert. #A3419)	MODP-2048	KAT	CAST	Pass Indicator Error Indicator	Verify computation of shared secret Z in dhEphem scheme	Boot Up
PBKDF2	Fccryptd.exe (Cert. #A3419)	HMAC-SHA-1	KAT	CAST	Pass Indicator Error Indicator	Key Derivation	Boot Up
RSA KeyGen (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	2048 bit	PCT	PCT	Pass Indicator Error Indicator	Key Pair Generation	Prior to the First Operational Use.
RSA SigGen (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	2048 bit	KAT	CAST	Pass Indicator Error Indicator	Sign	Boot Up
RSA SigVer (FIPS186-4)	Fccryptd.exe (Cert. #A3419)	2048 bit	KAT	CAST	Pass Indicator Error Indicator	Verify	Boot Up
SHA2-256	Fccryptd.exe (Cert. #A3419)	SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
SHA2-384	Fccryptd.exe (Cert. #A3419)	SHA2-384	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
SHA2-512	Fccryptd.exe (Cert. #A3419)	SHA2-512	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
SHA3-256[6]	Fccryptd.exe (Cert. #A3419)	SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
SHA3-384[6]	Fccryptd.exe (Cert. #A3419)	SHA2-384	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
SHA3-512[6]	Fccryptd.exe (Cert. #A3419)	SHA2-512	KAT	CAST	Pass Indicator Error Indicator	Hashing	Boot Up
TLS KDF	N/A	SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Implemented Exclusively for Self-Tests	Boot Up

[6] Note: SHA3-256, SHA3-386, HMAC SHA3-256 and HMAC SHA3-384 are CAVP tested, but not used by any service within the module.

Algorithm	Implementation	Test Properties	Test Method	Type	Indicator	Details	Condition
IKEv2	N/A	SHA2-256	KAT	CAST	Pass Indicator Error Indicator	Implemented Exclusively for Self-Tests	Boot Up

10.3. Periodic Self-Test Information

Table 21: Periodic Information

Period	Periodic Method
On-Demand	**Pre-Operational Self-Tests:** • Programmatically **Conditional Cryptographic Self-Tests:** • Programmatically • Manually **Pair-Wise Consistency Test:** • Programmatically

10.4. Error States

Table 22: Error States

State Name	Description	Conditions	Recovery Method	Indicator

Critical Error State	FortiClient Logs the Specific Test or Tests that failed in cryptdlog.txt	Boot Up, Manual	Power-Cycle	`Check Logs`
				`2022-12-13 01:08:03.763 [ 4824, 2300] - CryptdFipsEnterErrorMode()-> msg =`
				`FIPS: Running FIPS self-test ...`
				`FIPS Error: AES-CBC self-test failed.`
				`FIPS Error: Running FIPS self-test --> failed.`

10.5. Operator Initiation Self-Test

The startup self-tests can also be initiated on demand from the Windows command prompt using the following commands:

- **`FCCryptd.exe fips kat all`** (to initiate all self-tests)
- **`FCCryptd.exe fips kat <test>`** (to initiate a specific self-test)
- **`FCCryptd.exe fips <test>`** (to list available tests)
- **`FCCryptd.exe fips kat pbkdf`**

10.6. Additional Information

The results of the self-tests are logged to a file. For a default installation of FortiClient on the C: drive, you can view the Cryptdlog.txt file in: C:\Program Files\Fortinet\FortiClient\logs\fips.

When the self-tests are run, each implementation of an algorithm is tested. For example, when the AES self-test is run, all AES implementations are tested.

11.0 Life-Cycle Assurance

11.1. Startup Procedures

The FortiClient FIPS 140-3 installer packages are created and deployed using Microsoft's Enterprise Mobility and Security (EMS) platform. Operating the module without following the guidance in Section 11 will result in non-compliant behavior and is outside the scope of this Security Policy.

The basic steps in the deployment process are as follows:

1. Download FortiClient from the Fortinet support site. 32bit and 64bit versions are available:
   - FortiClientSetup_7.0.8.7048.zip (Vendor Affirmed)
   - FortiClientSetup_7.0.8.7048_x64.zip
2. Upload FortiClient to your EMS platform
3. Create FortiClient installation packages with the desired features and select the 'FIPS 140-3' option'.
4. Install the entropy token on the PC.
5. Deploy FortiClient using your EMS platform or install FortiClient manually on the PC.

It is important to note that the version number on the zip files includes the build number. Thus, to verify the installed version, open a Windows Command Prompt window as an administrator and run the following command:

- **`fccryptd.exe fips version`**

The command runs the software integrity test and displays the module version as shown in the following image.

Figure 3: Approved Mode Indicator

Launch the FortiClient application as you would launch any other Windows application. Failure to perform the steps outlined in this section will result in the module operating in a non-compliant state.

11.2. Administrator Guidance

FortiClient administrator guidance is publicly available from the Fortinet Technical Documentation site. The key administrator guidance documents are listed below:

- FortiClient Administration Guide
- EMS Administration Guide
- EMS QuickStart Guide

11.3. Non-Administrator Guidance

None. Non-Administrator guidance is included in the FortiClient Administration Guide.

11.4. Maintenance Requirements

None.

11.5. End of Life

Not Applicable.

11.6. Additional Information

None.

12.0 Mitigation of Other Attacks

Not Applicable.