GlobalProtect App (Android/iOS/Linux/macOS/Windows)

Software Version: 6.0.10

Hardware Version:
Intel Core i3-1215U
Intel Core i7-1250U
Apple M Series M1
Apple A Series A14
Qualcomm Snapdragon 888

FIPS 140-3 Non-Proprietary Security Policy

Palo Alto Networks, Inc.
www.paloaltonetworks.com

Revision Date: October 3, 2024
Document Version: 1.20

Table of Contents

1. General

The table below provides the Security Levels of the various sections of FIPS 140-3 in relation to the Palo Alto Networks GlobalProtect App (hereinafter referred to as the Module).

ISO/IEC 24759 Section 6. [Number Below]	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	1
8	Non-Invasive Security	N/A
9	Sensitive Security Parameter Management	1
10	Self-Tests	1
11	Life-Cycle Assurance	3
12	Mitigation of Other Attacks	N/A
Overall		1

Table 1 - Security Levels

2. Cryptographic Module Specification

The GlobalProtect App is a software-hybrid cryptographic module that runs on commercially available operating systems and mobile devices to provide security for users. Its cryptographic boundary is the entire software of the package, which is noted in Section 6 of this Security Policy. The GlobalProtect App secures traffic using TLS or IPsec, and allows users to connect to corporate networks to access their company's resources from anywhere in the world.

The module uses GlobalProtect App version 6.0.10 and meets an overall Security Level of 1. The GlobalProtect App provides only an Approved mode of operation, and is configured during initialization to operate only in an Approved mode of operation when in the operational state. Details regarding how to enter the Approved mode of operation is noted in the Life-Cycle Assurance section under Secure Operation. The Life-Cycle Assurance section also provides details regarding proper download/installation as well as steps to zeroize the module. The module is classified as a multi-chip standalone software-hybrid module.

FIPS 140-3 conformance testing was performed at Security Level 1 with the following configurations noted in the table below.

#	Operating System	Hardware Platform	Processor	PAA/Acceleration
1	Linux Ubuntu 20.04	HP Pavilion	Intel Core i3-1215U	AES-NI
2	Windows 11	HP Envy	Intel Core i7-1250U	AES-NI
3	macOS Big Sur 11	MacBook Air	Apple M Series M1	NEON
4	iOS 16	iPhone 12 Mini	Apple A Series A14	NEON
5	Android 12	Samsung Galaxy S21 Ultra	Qualcomm Snapdragon 888	AES-NI

Table 2 - Tested Operational Environments

#	Operating System	Hardware Platform
1	Windows 11	ARM Devices
2	Windows 10	Intel and ARM Devices
3	macOS Big Sur, Monterey	Intel Devices
4	macOS Big Sur, Ventura	ARM Devices
5	RedHat 8.1	GPC
6	CentOS 8.3	GPC
7	Google Android 13	Pixel 4 and Pixel 6
8	Apple iOS	Apple iPhone

Table 2A - Vendor Affirmed Operational Environments

The module utilizes the following Approved algorithms that have the following CAVP certificates:

CAVP Cert	Algorithm and Standard	Mode/Method	Description/Key Size(s) / Key Strength(s)	Use / Function
A1362	Counter DRBG [SP 800-90Arev1]	CTR DRBG	AES 256 bits without Derivation Function and with Prediction Resistance Enabled	Vetted conditioner for ESV Cert. #E14
A2873	Conditioning Component AES-CBC-MAC [SP 800-90B]	AES-CBC-MAC	128 bits	Intel Conditioner for Entropy Source
A2999	AES-CBC [SP 800-38A]	CBC	128, 192 and 256 bits	Encryption Decryption
A2999	AES-CTR [SP 800-38A]	CTR	128, 192 and 256 bits *Note: 128, 192, and 256 bits were tested, but not available for use*	Encryption Decryption
A2999	AES-ECB [SP 800-38A]	ECB	128, 192 and 256 bits	Encryption Decryption
A2999	AES-GCM [SP 800-38D]	GCM	128 and 256 bits *Note: 192 bits tested, but not available for use*	Encryption Decryption
A2999	Counter DRBG [SP 800-90Arev1]	CTR DRBG	AES 256 bits with Derivation Function Enabled	Random Bit Generator
A2999	ECDSA KeyGen (FIPS 186-4)	ECDSA KeyGen	P-256, P-384, P-521	Key Generation

A2999	ECDSA KeyVer (FIPS 186-4)	ECDSA KeyVer	P-256, P-384, P-521	Public Key Validation
A2999	ECDSA SigGen (FIPS 186-4)	ECDSA SigGen	P-256, P-384, P-521 with SHA2-256, SHA2-384, and SHA2-512	Signature Generation
A2999	ECDSA SigVer (FIPS 186-4)	ECDSA SigVer	P-256, P-384, P-521 with SHA2-256, SHA2-384, and SHA2-512	Signature Verification
A2999	HMAC-SHA-1 [FIPS 198-1]	HMAC	HMAC-SHA-1 with λ=160	Authentication for protocols
A2999	HMAC-SHA2-256 [FIPS 198-1]	HMAC	HMAC-SHA2-256 with λ=256	Authentication for protocols
A2999	HMAC-SHA2-384 [FIPS 198-1]	HMAC	HMAC-SHA2-384 with λ=384	Authentication for protocols
A2999	HMAC-SHA2-512 [FIPS 198-1]	HMAC	HMAC-SHA2-512 with λ=512 *Note: Tested, but not available for use*	Authentication for protocols
A2999	KAS-ECC-SSC SP 800-56Ar3	SP 800-56Arev3. KAS-ECC per IG D.F Scenario 2 path (2)	EphemeralUnified scheme using P-256/P-384/P-521 providing 128/192/256 bits of strength	Key Exchange
A2999	KDF TLS [SP 800-135rev1] (CVL)	TLS 1.2 KDF	TLS v1.2 Hash Algorithm: SHA2-256, SHA2-384	TLS
A2999	RSA SigGen (FIPS 186-4)	RSA SigGen (FIPS 186-4)	PKCS #1 v1.5: 2048, 3072, and 4096-bit with hashes SHA2-256/384/512	Signature Generation
A2999	RSA SigVer (FIPS 186-4)	RSA SigVer (FIPS 186-4)	PKCS #1 v1.5: 2048, 3072, 4096-bit (per IG C.F) with hashes SHA2-256, SHA2-384, SHA2-512(Signature Verification)	Signature Verification
A2999	SHA-1 [FIPS 180-4]	SHA	SHA-1	Non-Digital Signature Applications (e.g. component of HMAC)
A2999	SHA2-256 [FIPS 180-4]	SHA2	SHA2-256	Digital Signature Generation/Verification Non-Digital Signature Applications (e.g. component of HMAC)
A2999	SHA2-384 [FIPS 180-4]	SHA2	SHA2-384	Digital Signature Generation/Verification Non-Digital Signature Applications (e.g. component of HMAC)
A2999	SHA2-512 [FIPS 180-4]	SHA2	SHA2-512	Digital Signature Generation/Verification Non-Digital Signature Applications (e.g. component of HMAC)
A3429	SHA2-256 [FIPS 180-4]	SHA2	SHA2-256	Vetted conditioner for ESV Cert. #E15
AES Cert. A2999	KTS [SP 800-38F]	SP 800-38A, FIPS 198-1, and SP 800-38F. KTS (key	AES-CBC plus HMAC	Key Wrapping

and HMAC Cert. A2999		wrapping and unwrapping) per IG D.G.	128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength	
AES-GCM Cert. A2999	KTS [SP 800-38F]	SP 800-38D and SP 800-38F. KTS (key wrapping and unwrapping) per IG D.G.	AES-GCM 128 and 256-bit keys providing 128 or 256 bits of encryption strength	Key Wrapping
ESV Cert. #E14	ESV [SP 800-90B]	ESV	Apple corecrypto physical entropy source	Entropy
ESV Cert. #E15	ESV [SP 800-90B]	ESV	Apple corecrypto non-physical entropy source	Entropy
KAS-ECC-SSC Cert. #A2999, KDF TLS Cert. #A2999	KAS [SP 800-56Arev3]	SP 800-56Arev3. KAS-ECC per IG D.F Scenario 2 path (2).	P-256, P-384, and P-521 curves providing 128, 192, or 256 bits of encryption strength	Key Exchange with protocol KDF
N/A	ENT (P) (SP 800-90B)	ENT	ENT (P)	Entropy
Vendor Affirmed	CKG (SP 800-133rev2)	Section 5.2	Cryptographic Key Generation; SP 800-133 and IG D.H.	Key Generation Note: The seeds used for asymmetric key pair generation are produced using the unmodified/direct output of the DRBG

Table 3 - Approved Algorithms

Notes:

- There are some algorithm modes that were tested but not implemented by the module. Only the algorithms, modes, and key sizes that are implemented by the module are shown in this table.
- The module's AES-GCM implementation conforms to IG C.H scenario #1 following RFC 5288 for TLS. The module is compatible with TLSv1.2 and provides support for the acceptable GCM cipher suites from SP 800-52 Rev1, Section 3.3.1. The operations of one of the two parties involved in the TLS key establishment scheme were performed entirely within the cryptographic boundary of the module being validated. The counter portion of the IV is set by the module within its cryptographic boundary. When the IV exhausts the maximum number of possible values for a given session key, the first party, client or server, to encounter this condition will trigger a handshake to establish a new encryption key. In case the module's power is lost and then restored, a new key for use with the AES GCM encryption/decryption shall be established.
- No parts of the TLS protocol, other than the KDF, have been tested by the CAVP and CMVP.
- In accordance with FIPS 140-3 IG D.H, the cryptographic module performs Cryptographic Key Generation as per the requirements from section 6 in SP800-133rev2. The resulting generated seed used in the asymmetric key generation is the unmodified output from SP800-90A DRBG.
- As the module can only operate in the Approved mode of operation, the following tables are not present in this Security Policy:
  - Non-Approved Algorithms Allowed in Approved Mode of Operation
  - Non-Approved Algorithms Allowed in Approved Mode of Operation with No Security Claimed
  - Non-Approved Algorithms Not Allowed in Approved Mode of Operation

Cryptographic Boundary

Figure 1 below depicts the cryptographic boundary and physical perimeter (light blue color area). The cryptographic boundary includes all of the software components and the specified hardware components (CPU's). The physical perimeter is the Tested Operational Environment's Physical Perimeter (TOEPP) on which the module runs.

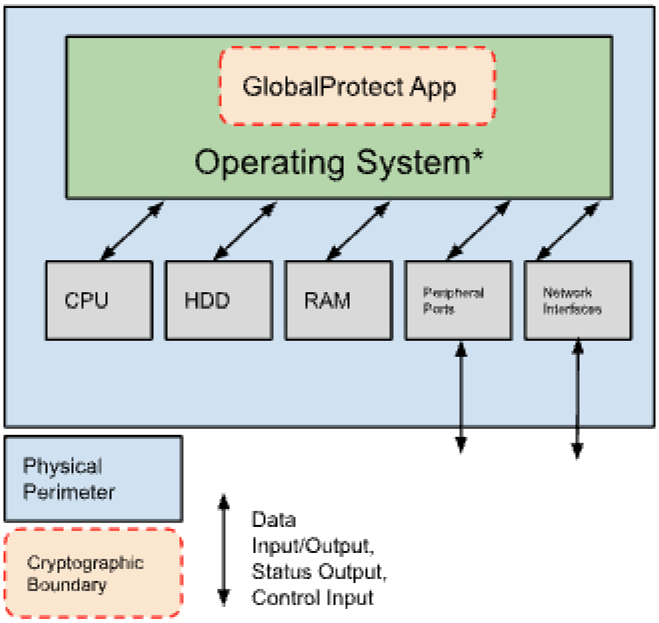

Figure 1: Cryptographic Boundary

* See details below regarding Operating Systems/Environments tested

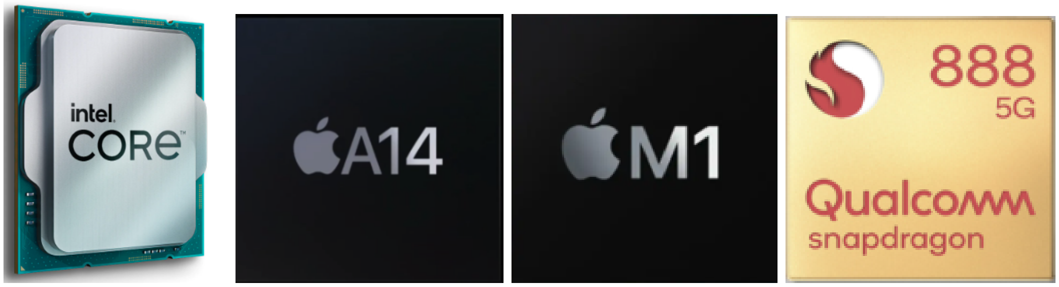

Figure 1B: Hardware Components

3. Cryptographic Module Interfaces

The module provides a number of physical and logical interfaces to the device, and the physical interfaces provided by the module are mapped to the following FIPS 140-3 defined logical interfaces: data input, data output, control input, control output (N/A), status output, and power. The logical interfaces and their mapping are described in the following table.

Physical Port	Logical Interface	Data that passes over port/interface
Physical ports of the tested platform	Status Output	GUI status window and log files generated and output via GUI/CLI
Physical ports of the tested platform	Data Input	Portal information, keys from OS certificate store or during TLS/IPsec negotiation
Physical ports of the tested platform	Data Output	Keys for establishing secure sessions such as TLS
Physical ports of the tested platform	Control Input	GUI/CLI, string value from pangps.xml, com.paloaltonetworks.gp.pangps.plist, MDM, or Windows Registry (See Secure Operation section below)

Table 4 - Ports and Interfaces

Note: Physical ports include items such as LAN/USB/Monitor/Keyboard.

4. Roles, Services, and Authentication

The module supports one role, which is the Crypto-Officer. The module does not provide a maintenance role or bypass capability, and services for both roles are noted below. There is no authentication supported by the module, and self-initiated cryptographic output capability is not supported.

Role	Service	Input	Output
Crypto-Officer	Show Status	Request system status	Module displays status information
	Show Version	Query module for version information	Module displays version information
	Self-Test	Command module to run Self-Tests	Module provides output of Self-Test results via logs

	Security Configuration Management	Configuring module with setup data details to support VPN establishment	Logs provide configuration changes
	VPN Tunnel	Initialize VPN connection	System log provide VPN status
	Zeroize	Command module to zeroize	All SSPs zeroized (module uninstalled)

Table 5 - Roles, Service Commands, Input and Output

Service	Description	Approved Security Functions		Keys and/or SSPs	Roles	Access rights to Keys and/or SSPs	Indicator
Show Status	Provides information regarding the status of the system (fetched via GUI/CLI)	N/A		N/A	Crypto-Officer	N/A	System logs or status window
Show Version	Provides information regarding version	N/A		N/A	Crypto-Officer	N/A	Module provides version information
Self-Test	Performs on-demand Self-Tests (executed via reboot of platform)	RSA SigVer (FIPS 186-4)		Software Integrity Verification Key	Crypto-Officer	E	System logs
Security Configuration Management	Configures the module with necessary setup details to support VPN establishment (updated via GUI/CLI)	N/A		CA Certificates	Crypto-Officer	R/W/E	System logs
				RSA Public Keys			
				RSA Private Keys			
				ECDSA Public Keys			
				ECDSA Private Keys			
VPN Tunnel	Creates an SSL/IPsec VPN tunnel (executed by operator interaction with GUI/CLI)	KAS	KDF TLS	TLS Pre-Master Secret	Crypto-Officer	G/E/Z	System logs
			KDF TLS	TLS Master Secret		G/E/Z	
			CKG, ECDSA KeyGen (FIPS 186-4), ECDSA KeyVer (FIPS 186-4), KAS-ECC-SSC	TLS ECDHE Public Components		G/E/Z	
				TLS ECDHE Private Components		G/E/Z	
		KTS	HMAC-SHA2-256 HMAC-SHA2-384	TLS HMAC Keys		G/E/Z	
			AES-CBC	TLS Encryption Keys			
		KTS	AES-GCM				
		RSA SigVer (FIPS 186-4)		CA Certificates		W/E	

		ECDSA SigVer (FIPS 186-4)				
		RSA SigVer (FIPS 186-4)	RSA Public Keys		W/E	
		RSA SigGen (FIPS 186-4)	RSA Private Keys		E	
		ECDSA SigVer (FIPS 186-4)	ECDSA Public Keys		W/E	
		ECDSA SigGen (FIPS 186-4)	ECDSA Private Keys		E	
		AES-CBC	IPSec Session Keys		W/E	
		AES-GCM				
		HMAC-SHA-1	IPSec Authentication Keys		W/E	
		Counter DRBG, ENT (P), ESV	Entropy Input String, DRBG Seed		G/E/Z	
		Counter DRBG	DRBG Key		G/E/Z	
			DRBG V		G/E/Z	
Zeroize	Removes all SSPs from the module (Performed via uninstall of the module)	N/A	All Keys and SSPs	Crypto-Officer	Z	Removal of module and confirmation via OS status window

Table 6 - Approved Services

G = Generate: The module generates or derives the SSP.

R = Read: The SSP is read from the module (e.g. the SSP is output).

W = Write: The SSP is updated, imported, or written to the module.

E = Execute: The module uses the SSP in performing a cryptographic operation.

Z = Zeroise: The module zeroises the SSP.

Note: There is no table for non-Approved services as the module only supports Approved services.

5. Software/Firmware Security

The module performs the Software Integrity test by verifying the digital signature of the module using RSA 2048 with SHA2-256 (Cert. #A2999) or RSA 3072 with SHA2-384 (Cert. #A2999) during the Pre-Operational Self-Test. RSA 2048 with SHA2-256 is used for Windows/macOS/iOS/Android (OE #2, 3, 4, 5 in Table 2) and RSA 3072 with SHA2-384 is used for Linux (OE #1 in Table 2). The Software Integrity Verification Key is used for this integrity test.

The integrity test can be performed by restarting the GlobalProtect app service, which is noted in the Life-Cycle Assurance section for each platform. The test can also be performed by restarting the platform for which the module runs on. Either of the actions (restarting the GlobalProtect app service or restarting the host platform) can be used to perform the integrity test on demand.

For information regarding the file type, see details in Operational Environment. The module comes packaged and ready for installation once it has been downloaded from the Palo Alto Networks support site.

6. Operational Environment

The module has a modifiable operational environment, and was tested on the following environments operating on a general-purpose computing platform. For details regarding platforms tested on, see Table 2. To properly run the module on the operating environments, see Life-Cycle Assurance for details on configuring the systems.

Platform	Package Name
Linux	PanGPLinux-6.0.10.tgz
Windows	GlobalProtect64-6.0.10.msi
macOS	GlobalProtect-6.0.10.pkg
iOS	6.0.10 on App Store
Android	6.0.10 on Google Play

Table 7 - GlobalProtect Package Names

To install, download the following from the Palo Alto Networks Support site (https://support.paloaltonetworks.com/) or on the mobile platform (e.g. Apple App Store or Google Play).

Operator porting rules:

The CMVP allows user porting of a validated software module to an operational environment which was not included as part of the validation testing. An operator may install and run the GlobalProtect App on any general purpose computer (GPC) or platform using the specified operating system on the validation certificate or other compatible operating system and affirm the modules continued FIPS 140-3 validation compliance.

The CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when ported and executed in an operational environment not listed on the validation certificate.

7. Physical Security

The module is a multi-chip standalone software-hybrid module that meets Level 1 physical security requirements. Physical security is provided by the production grade components on the GPC that the module runs on. The production grade components also come with standard passivation applied to them.

8. Non-Invasive Security

No approved Non-Invasive attack mitigation test metrics are defined at this time.

9. Sensitive Security Parameter Management

The following table details all the sensitive security parameters utilized by the module.

Key/SSP Name/Type	Strength	Security Function and Cert. Number	Generation	Import/Export[1]	Establishment	Storage[2,3]	Zeroization[4]	Use & Related Keys
CA Certificates	112 - 256 bits	RSA SigVer (FIPS 186-4), ECDSA SigVer (FIPS 186-4) Cert. #A2999	N/A	Import/Exported in plaintext	N/A	Protected in OS key store	Zeroization service	ECDSA/RSA Public key used to extend trust to a root CA, intermediate CA, and left/end entity certificates
RSA Public Keys	112 - 150 bits	RSA SigVer (FIPS 186-4) Cert. #A2999	N/A	Import: Yes from OS key store or Plaintext during TLS handshake Export: Plaintext during TLS handshake	N/A	Protected in OS key store	Zeroization service	RSA public keys managed as certificates for the verification of signatures, establishment of TLS, and peer authentication. (RSA 2048/3072/4096 bits)
RSA Private Keys	112 - 150 bits	RSA SigGen (FIPS 186-4) Cert. #A2999	N/A	Imported: Encrypted via TLS Exported: No	N/A	Protected in OS key store	Zeroization service	RSA Private key used for authentication, and signature generation (RSA 2048, 3072, or 4096 bits).
ECDSA Public Keys	128 - 256 bits	ECDSA SigVer (FIPS 186-4) Cert. #A2999	N/A	Import: Yes from OS key store or Plaintext during TLS handshake Export: Plaintext during TLS handshake	N/A	Protected in OS key store	Zeroization service	ECDSA public keys managed as certificates for the verification of signatures, establishment of TLS, and peer authentication. (P-256/384/521)
ECDSA Private Keys	128 - 256 bits	ECDSA SigGen (FIPS 186-4) Cert. #A2999	N/A	Imported: Encrypted via TLS Exported: No	N/A	Protected in OS key store	Zeroization service	ECDSA Private key used for authentication, and signature generation (P-256, P-384 or P-521.
TLS ECDHE Private Components	128 - 256 bits	ECDSA KeyGen (FIPS 186-4), ECDSA KeyVer (FIPS 186-4), KAS-ECC-SSC Cert. #A2999	CKG	N/A	N/A	RAM – Plaintext	Zeroized at session termination	ECDHE private component used in key agreement (P-256, P-384, P-521)
TLS ECDHE Public Components	128 - 256 bits	KAS-ECC-SSC Cert. #A2999	N/A	Import: No Export: Only exits the module to the peer for TLS protocol implementation	N/A	RAM – Plaintext	Zeroized at session termination	ECDHE public component used in key agreement (P-256/384/521)
TLS Pre-Master Secret	N/A	KDF TLS Cert. #A2999	N/A	N/A	KAS	RAM – Plaintext	Zeroized at session termination	Value used during TLS handshake for session negotiation
TLS Master Secret	N/A	KDF TLS Cert. #A2999	N/A	N/A	Derived using SP 800-135 KDF	RAM – Plaintext	Zeroized at session termination	Secret value used to derive the TLS session key
TLS Encryption Keys	128 or 256 bits	AES-CBC, AES-GCM Cert. #A2999	N/A	N/A	Derived using SP 800-135 KDF	RAM – Plaintext	Zeroized at session termination	AES keys used in TLS connections (AES 128/256 bits GCM or CBC)

[1] All SSPs are electronically imported/exported. At first load, the CA Certificates, RSA/ECDSA Public Keys are manually imported.

[2] For items noted as protected by OS key store, this refers to the platform on which the module is installed providing a space to host these keys/SSPs and utilizing control mechanisms to ensure only proper individuals can access these items (e.g. operator must authenticate to GPC to access them). The GPC also provides security as the key store is protected in the physical perimeter of the GPC. These keys are considered to be outside the module's cryptographic boundary.

[3] The module does not provide persistent keys/SSPs storage.

[4] Zeroization service is an explicit service unless it is handling temporary values, which are zeroized implicitly upon session termination or when the platform is rebooted. When the zeroization service is invoked, it overwrites files with a random pattern.

TLS HMAC Keys	256 - 384 bits	HMAC-SHA2-256, HMAC-SHA2-384 Cert. #A2999	N/A	NA	Derived using SP 800-135 KDF	RAM – Plaintext	Zeroized at session termination	HMAC keys used in TLS connections (SHA2-256, SHA2-384)
IPSec Session Keys	128 bits minimum	AES-CBC, AES-GCM Cert. #A2999	N/A	Imported Encrypted via AES-GCM/CBC	KTS	RAM – Plaintext	Zeroized at session termination	Used to encrypt sessions AES CBC (128 bits) or AES GCM (128 or 256 bits)
IPSec Authentication Key	160 bits	HMAC-SHA-1 Cert. #A2999	N/A	Imported Encrypted via AES-GCM/CBC	KTS	RAM – Plaintext	Zeroized at session termination	Used as part of authentication for IPsec data (HMAC-SHA-1)
Entropy Input String	112 bits minimum	CKG (vendor affirmed), ENT (P), ESV, Counter DRBG Cert. #A2999	Entropy as per SP 800-90B	N/A	N/A	RAM - plaintext	Power cycle	DRBG entropy input string coming from the entropy source used in the generation of random values
DRBG Seed	384 bits	Counter DRBG Cert. #A2999	Entropy as per SP 800-90B	N/A	N/A	RAM - plaintext	Power cycle	DRBG seed coming from the entropy input string used in the generation of random values
DRBG Key	256 bits	Counter DRBG Cert. #A2999	Constructed as per SP-800-90 Ar1	N/A	N/A	RAM - plaintext	Power cycle	Internal DRBG State
DRBG V	128 bits	Counter DRBG Cert. #A2999	Constructed as per SP 800-90Ar1	N/A	N/A	RAM - plaintext	Power cycle	Internal DRBG State
Software Integrity Verification Key	2048 or 3072 bits	RSA SigVer (FIPS 186-4) Cert. #A2999	N/A	N/A	Pre-computed at compile time	RAM - plaintext	Zeroization service	Used to verify the integrity of the module *(Note: This is not considered an SSP)*

Table 8 - SSPs

Entropy Sources	Minimum Number of Bits of Entropy	Details
Apple Non-Physical Entropy Source	384 bits	The module uses entropy provided by Apple's entropy source, which is covered by ESV cert. #E15. This entropy source provides full entropy per output. The DRBG is seeded with 384 bits of entropy from this source. (Apple A Series processor)
Apple Physical Entropy Source	384 bits	The module uses entropy provided by Apple's entropy source, which is covered by ESV cert. #E14. This entropy source provides full entropy per output. The DRBG is seeded with 384 bits of entropy from this source. (Apple M Series processor)
Intel RDSEED	384 bits	Entropy provided by Intel CPU with RDSEED as the noise source to provide at least 384 bits of entropy to seed the DRBG. This entropy source provides full entropy per output. The DRBG is seeded with 384 bits of entropy from this source. (Windows and Linux platforms)
N/A	112 bits	For Android platforms, the module performs an entropy load that meets FIPS 140-3 IG 9.3.A Scenario 2(b). The DRBG is seeded with 384 bits of data which is assumed to contain at least 112 bits of entropy. No assurance of the minimum strength of generated SSPs (e.g., keys)

Table 9 - Non-Deterministic Random Number Generation Specification

10. Self-Tests

The cryptographic module performs the following tests below. The operator can command the module to perform the pre-operational and cryptographic algorithm self-tests (CASTs) by reloading the module or power cycling the underlying platform; these tests do not require any additional operator action. In the event that a Self-Test fails, the module will enter an error state until the issue is resolved, and provide a status output message with the failure.

Pre-Operational Self-Tests

Algorithm	Self-Test Details
Software Integrity Test	Digital signature verification (PKCS #1 v1.5) using RSA 2048 bits with SHA2-256 or RSA 3072 bits with SHA2-384 (Linux) Note: The RSA and SHA2-256/SHA2-384 CASTs are performed prior to the Software Integrity Test.

Table 10 - Pre-Operational Self-Tests

Conditional Self-Tests

Algorithm	Self-Test Details
AES ECB Encrypt	KAT using AES ECB 128 bits
AES ECB Decrypt	KAT using AES ECB 128 bit
AES GCM Encrypt	KAT using AES GCM 256 bits
AES GCM Decrypt	KAT using AES GCM 256 bits
Counter DRBG	KAT: AES-256 Counter DRBG Note: DRBG Health Tests as specified in SP800-90A Section 11.3 are performed (i.e. instantiate/generate/reseed)

ECDSA Sign	KAT using P-256 and SHA2-256
ECDSA Verify	KAT using P-256 and SHA2-256
HMAC-SHA-1	KAT using HMAC-SHA-1
HMAC-SHA2-224	KAT using HMAC-SHA2-224 Note: Only used for self-test.
HMAC-SHA2-256	KAT using HMAC-SHA2-256
HMAC-SHA2-384	KAT using HMAC-SHA2-384
HMAC-SHA2-512	KAT using HMAC-SHA2-512
RSA Sign	KAT using RSA 2048 bits and SHA2-256 (PKCS #1 v1.5 and PKCS PSS)
RSA Verify	KAT using RSA 2048 bits and SHA2-256 (PKCS #1 v1.5 and PKCS PSS)
SHA-1	KAT using SHA-1
SHA2-256	KAT using SHA2-256
SHA2-384	KAT using SHA2-384
SHA2-512	KAT using SHA2-512
SP 800-56Ar3 KAS-ECC-SSC	KAT for KAS-ECC-SSC using P-256 (Shared Secret Computation) primitive Z value
SP 800-135r1 KDF TLS	KAT for TLSv1.2 KDF
SP 800-90B Health Tests	SP 800-90B Health Tests on the Entropy Source

Table 11 – Conditional Cryptographic Algorithm Self-Tests

Algorithm	Self-Test Details
ECC	ECC Pair-wise Consistency Test (PCT) for ECDSA and KAS-ECC key pairs

Table 12 – Conditional Pair-Wise Consistency Tests

Algorithm	Self-Test Details
SP 800-56Arev3 KAS-ECC-SSC	SP 800-56Arev3 Assurance Tests based on Sections 5.5.2, 5.6.2, and 5.6.3

Table 13 – Conditional Critical Functions Test

Error Handling

In the event that the module encounters an error, the following provide the status indicators:

Error	Status Indicator
Conditional Cryptographic Algorithm Self-Test Failure	System prints log with error message
Integrity Test Failure	System prints log with error message
Conditional Test Failure	System prints log with error message

Table 14 - Error Indicators

11. Life-Cycle Assurance

The GlobalProtect App is designed to handle the various stages of a module's Life-Cycle. The sections below highlight the details for each stage.

Secure Delivery Procedures

The security of the module is maintained during the transfer of these products from production sites to the customer through the following mechanisms:

- The customer visits the Palo Alto Networks support site and downloads the proper version of the module

- Palo Alto Networks provides a SHA2-256 checksum on the support site to validate the proper version, and to ensure that the downloaded software matches the one provided by Palo Alto Networks

Secure Operation

The steps below are required in order to initialize the module into an Approved state (compliant state). Failure to follow the directions below will result in the module operating in a non-compliant state, which is considered out of scope of this validation.

Linux - Ubuntu

To prep this environment for GlobalProtect initialization, perform the following steps:

- Visit https://ubuntu.com/advantage and receive a token
- On the endpoint issue the following commands:
    - sudo apt update
    - sudo apt install ubuntu-advantage-tools
    - sudo ua attach <token>
        - Note: This token is from the advantage site noted above
    - sudo ua enable fips-updates
- Reboot the endpoint

Once complete, initialize the GP App into the Approved state using the following procedure:

- Download the desired bundle (e.g. PanGPLinux-6.0.10.tgz)
- Navigate to the folder it is hosted and untar the bundle
    - tar -xvf PanGPLinux-6.0.10.tgz
- Once complete, run the following to install the UI version (Note: for 20.04 you must use the focal file)
    - sudo apt-get install ./GlobalProtect_UI_focal_deb-*.deb
- Once complete, the UI will pop up and ask for a portal address
- To enable the Approved state ("FIPS-CC mode"), perform the following:
    - Edit pangps.xml that is located in /opt/paloaltonetworks/globalprotect with the following string:
        - <enable-fips-cc-mode>yes</enable-fips-cc-mode>
    - Reboot the system for the changes to take effect
- Once complete, the "About" section will note the version and "FIPS-CC Mode Enabled" as status output

Windows 11

To prep this environment for GlobalProtect initialization, perform the following steps:

- Launch Command Prompt
- Enter regedit to open the Windows Registry
- In the Windows Registry, go to:

    - HKEY_LOCAL_MACHINES\System\CurrentControlSet\Control\Lsa\FipsAlgorithmPolicy\
- Right-click the Enabled registry value and then select Modify…
- Set the Value Data to 1
- Click OK, and then restart the endpoint

Once the above has been complete, perform the following to initialize the GP App into the Approved state:

- Launch the Command Prompt
- Enter regedit to open the Windows Registry
- In the Windows Registry, go to: HKEY_LOCAL_MACHINE\SOFTWARE\Palo Alto Networks\GlobalProtect\Settings\
- Click Edit and then select New > String Value
- When prompted, set the Name of the new registry value to enable-fips-cc-mode
- Right-click the new registry and then select Modify…
- To initialize the Approved state ("FIPS-CC mode"), set the Value Data to yes
- Click OK
- Restart the GlobalProtect App service
    - Launch the Command Prompt
    - Enter services.msc to open the Windows Services manager
    - From the Services list, select PanGPS
    - Restart the service (Stop and then Start)

The module will display the following message in the About section following the service restart: "FIPS-CC Mode Enabled".

macOS

For the GlobalProtect App running on macOS, complete the steps below:

- Launch a plist editor, such as Xcode.
- In the plist editor, open the following plist file:
  /Library/Preferences/com.paloaltonetworks.GlobalProtect.settings.plist
- Locate the GlobalProtect App Settings dictionary: /Palo Alto Networks/GlobalProtect/Settings
    - Note: If the Settings dictionary does not exist, create it. You can add each key to the Settings dictionary as a string
- Initialize the Approved state ("FIPS-CC mode") for the GlobalProtect App by adding the following key-value pair in the Settings dictionary:
    - <key>enable-fips-cc-mode</key>
    - <string>yes</string>
- Restart the GlobalProtect App service by one of the following methods:
    - Reboot your endpoint
    - Restart application through Activity Monitor
        - Launch Finder
        - From the Finder sidebar, select Applications
        - Open the Utilities folder
        - Open Activity Monitor
        - Stop the PanGPS service (GlobalProtect)

    - Restart the GlobalProtect App application and GlobalProtect App service (PanGPS)
        - Launch Terminal
        - Execute the following commands:

```
username>$ launchctl unload -S Aqua /Library/LaunchAgents/com.paloaltonetworks.gp.pangpa.plist
username>$ launchctl unload -S Aqua /Library/LaunchAgents/com.paloaltonetworks.gp.pangps.plist
username>$ launchctl load -S Aqua /Library/LaunchAgents/com.paloaltonetworks.gp.pangps.plist
username>$ launchctl load -S Aqua /Library/LaunchAgents/com.paloaltonetworks.gp.pangpa.plist
```

iOS

For the GlobalProtect App running on iOS, complete the steps below:

- Access the App Store on the Apple device
- Search for GlobalProtect and download the application
- Once the app has been downloaded, navigate to the MDM to initialize the Approved state ("FIPS-CC mode") on the endpoint
- On the MDM service such as Workspace One, enter the following custom key:
    - Key: enable-fips-cc-mode
    - Value: yes
- Push the configuration to the iOS device, and then restart the application

Android

To initialize the GP App into its Approved state ("FIPS-CC mode"), follow the procedure below:

- Access the Google Play store
- Search for GlobalProtect and download the application
- Once the app has been downloaded, navigate to the MDM to initialize the Approved state (FIPS-CC mode) on the endpoint
- On the MDM service such as Workspace One, enter the following custom key:
    - Key: enable-fips-cc-mode
    - Value: yes
- Push the configuration to the Android device, and then restart the application

End of Life / Sanitization

End of life dates for software modules are announced publicly via Palo Alto Networks' services website. Crypto-Officers shall follow the procedure below for the secure destruction of their module:

Note: This process will cause the module to no longer function after it has wiped all configurations and keys.

Linux

1. Launch the Terminal
2. Issue the following command:
    a. sudo apt-get remove globalprotect

Windows

1. Select Start > Control Panel > Programs > Programs and Features

2. Select GlobalProtect from the list, and then click Uninstall
3. When prompted to continue the uninstall, click Yes.

macOS

1. Issue the following as an administrator on the macOS device:
    a. sudo /Applications/GlobalProtect.app/Contents/Resources/uninstall_gp.sh

iOS

1. Tap and hold the GlobalProtect App icon until the icon jiggles
2. Tap the X on the top-left corner of the icon
3. When prompted, select Delete GlobalProtect
4. Tap Done or press/swipe for the home button to return to the home screen

Android

1. Launch the Settings app
2. Tap Apps & Notifications
3. Tap GlobalProtect
4. Tap Uninstall

Administrator/User Guidance

Palo Alto Networks provides documentation for all products, which can be accessed here: https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/globalprotect/6-0/globalprotect-app-user-guide/globalprotect-app-user-guide.pdf

12. Mitigation of Other Attacks

This module is not designed to mitigate against any other attacks outside of the FIPS 140-3 scope.