Palo Alto Networks

GlobalProtect App

FIPS 140-3 Non-Proprietary Security Policy

Table of Contents

List of Tables

List of Figures

1 General

1.1 Overview

This document may freely be reproduced and distributed in its entirety.

The GlobalProtect App network security client for endpoints, from Palo Alto Networks®, enables organizations to protect the mobile workforce by extending the Next-Generation Security Platform to all users, regardless of location. It secures traffic by applying the platform's capabilities to understand application use, associate the traffic with users and devices, and enforce security policies with next-generation technologies.

1.2 Security Levels

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	1
12	Mitigation of other attacks	N/A
	Overall Level	1

Table 1: Security Levels

2 Cryptographic Module Specification

2.1 Description

Purpose and Use:

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 utilizes GlobalProtect App version 6.0.10. Please refer to the Tested Operational Environment table below for specifics. 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.

Module Type: Software-hybrid

Module Embodiment: Multi-Chip Standalone

Module Characteristics:

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).

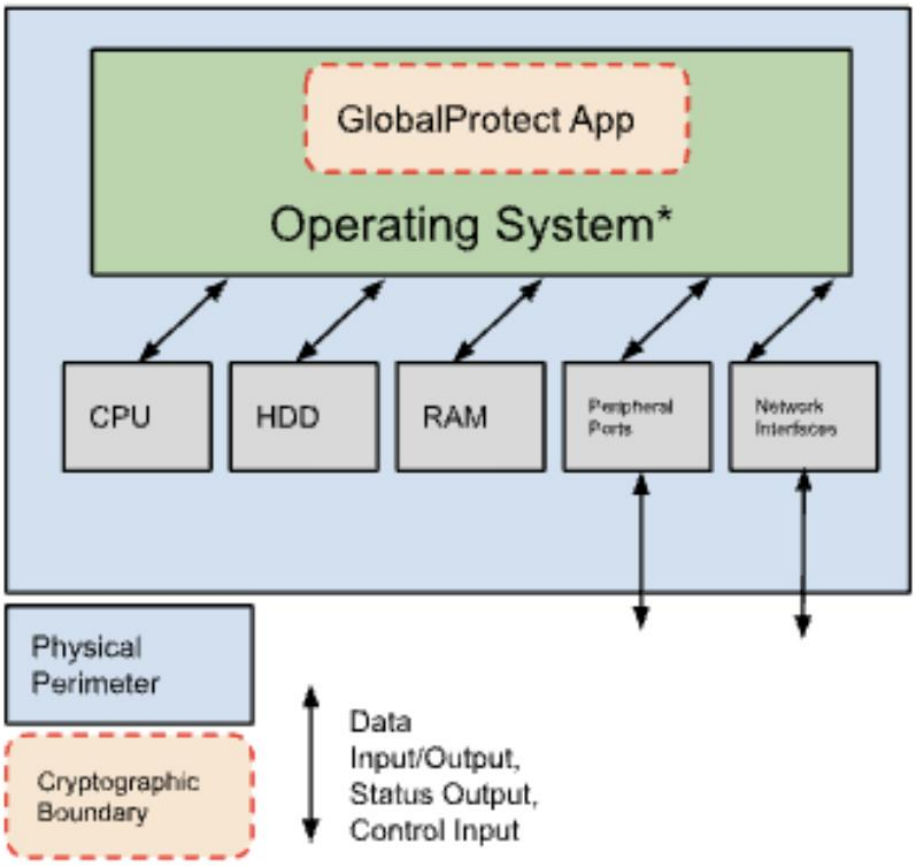

Figure 1: Cryptographic Boundary

Tested Operational Environment's Physical Perimeter (TOEPP):

The Tested Operational Environment's Physical Perimeter is the physical perimeter of the GPC on which the module runs.

2.2 Tested and Vendor Affirmed Module Version and Identification

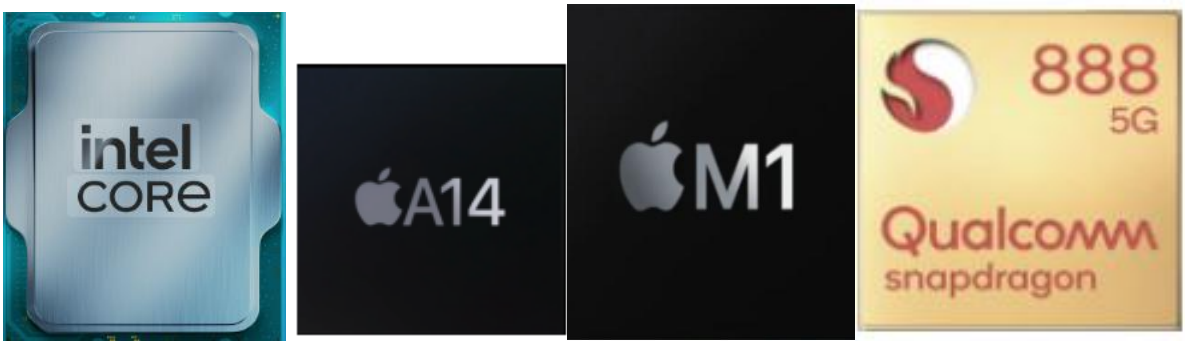

Figure 2. Hardware Components

Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):

Package or File Name	Software/ Firmware Version	Features	Integrity Test
6.0.10 from Apple App Store	6.0.10	N/A	Yes
6.0.10 from Google Play Store	6.0.10	N/A	Yes
GlobalProtect-6.0.10.pkg	6.0.10	N/A	Yes
GlobalProtect64-6.0.10.msi	6.0.10	N/A	Yes
PanGPLinux-6.0.10.tgz	6.0.10	N/A	Yes

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

Tested Module Identification – Hybrid Disjoint Hardware:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
Apple A Series A14	Apple A Series A14	N/A	N/A	N/A
Apple M Series M1	Apple M Series M1	N/A	N/A	N/A
Intel Core i3-1215U	Intel Core i3-1215U	N/A	N/A	N/A
Intel Core i7-1250U	Intel Core i7-1250U	N/A	N/A	N/A
Qualcomm Snapdragon 888	Qualcomm Snapdragon 888	N/A	N/A	N/A

Table 3: Tested Module Identification – Hybrid Disjoint Hardware

Tested Operational Environments - Software, Firmware, Hybrid:
The module is a modifiable operational environment as per FIPS 140-3 Level 1 specifications. The hypervisor environment provides an isolated operating environment and is the single operator of the virtual machine.

The tested operating environments isolate virtual systems into separate isolated process spaces. Each process space is logically separated from all other processes by the operating environments software and hardware. The module functions entirely within the process space of the isolated system as managed by the single operational environment. This implicitly meets the FIPS 140-3 requirement that only one (1) entity at a time can use the cryptographic module.

Operating System	Hardware Platform	Processors	PAA/PAI	Hypervisor or Host OS	Version(s)
Android 12	Samsung Galaxy S21 Ultra	Qualcomm Snapdragon 888	Yes	N/A	6.0.10
iOS 16	iPhone 12 Mini	Apple A Series A14	Yes	N/A	6.0.10
Linux Ubuntu 20.04	HP Pavilion	Intel Core i3-1215U	Yes	N/A	6.0.10
macOS Big Sur 11	MacBook Air	Apple M Series M1	Yes	N/A	6.0.10
Windows 11	HP Envy	Intel Core i7-1250U	Yes	N/A	6.0.10

Table 4: Tested Operational Environments - Software, Firmware, Hybrid

The following processor algorithm accelerators are used for the above environments:

Android 15: AES-NI
iOS 26: NEON
Linux Ubuntu 20.04: AES-NI
macOS Big Sur 11: NEON
Windows 11: AES-NI

Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:

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

Table 5: Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

2.3 Excluded Components

N/A – The module doesn’t support excluded components.

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Type	Status Indicator
Approved Mode	The module has one approved mode of operation and is always in approved mode after initialization	Approved	Global indicator ("FIPS-CC")

Table 6: Modes List and Description

The module only operates in an approved mode of operation and is in the approved mode when installed, initialized and configured per section 11.1 of the Security Policy.

Mode Change Instructions and Status :

See Life-Cycle Assurance section.

2.5 Algorithms

Approved Algorithms:

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A2999	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-GCM	A2999	Direction - Decrypt, Encrypt IV Generation - Internal IV Generation Mode - 8.2.1 Key Length - 128, 192, 256 Tag Length - 104, 112, 120, 128, 64, 96 IV Length - IV Length: 96 Payload Length - Payload Length: 8-65536 Increment 8 AAD Length - AAD Length: 0-65536 Increment 8	SP 800-38D
Counter DRBG	A2999	Prediction Resistance - No, Yes Supports Reseed - Yes Mode - AES-128, AES-192, AES-256 Derivation Function Enabled - No, Yes Additional Input - Additional Input: 256, Additional Input: 320, Additional Input: 384 Entropy Input - Entropy Input: 256, Entropy Input: 320, Entropy Input: 384 Nonce - Nonce: 256, Nonce: 320, Nonce: 384 Personalization String Length - Personalization String Length: 256, Personalization String Length:	SP 800-90A Rev. 1

Algorithm	CAVP Cert	Properties	Reference
		320, Personalization String Length: 384 Returned Bits - 512	
ECDSA KeyGen (FIPS186-4)	A2999	Curve - P-256, P-384, P-521 Secret Generation Mode - Testing Candidates	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A2999	Curve - P-256, P-384, P-521	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A2999	Curve - P-256, P-384, P-521 Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A2999	Curve - P-256, P-384, P-521 Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512	FIPS 186-4
HMAC-SHA-1	A2999	MAC - MAC: 160 Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-256	A2999	MAC - MAC: 256 Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
HMAC-SHA2-384	A2999	MAC - MAC: 384 Key Length - Key Length: 256-2048 Increment 8	FIPS 198-1
KAS-ECC-SSC Sp800-56Ar3	A2999	Domain Parameter Generation Methods - P-256, P-384, P-521 Scheme - ephemeralUnified - KAS Role - initiator, responder	SP 800-56A Rev. 3
KDF TLS (CVL)	A2999	TLS Version - v1.2 Hash Algorithm - SHA2-256, SHA2-384	SP 800-135 Rev. 1
RSA SigGen (FIPS186-4)	A2999	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA2-256	FIPS 186-4
RSA SigVer (FIPS186-4)	A2999	Signature Type - ANSI X9.31, PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096 Hash Pair - Hash Algorithm - SHA-1 Public Exponent Mode - Fixed Fixed Public Exponent - 010001	FIPS 186-4
SHA-1	A2999	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-256	A2999	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-384	A2999	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4
SHA2-512	A2999	Message Length - Message Length: 0-65536 Increment 8	FIPS 180-4

Table 7: Approved Algorithms

Vendor-Affirmed Algorithms:

Name	Properties	Implementation	Reference
CKG	Key Type:Asymmetric	N/A	Cryptographic Key Generation; SP 800-133rev2 and IG D.H (asymmetric seeds) from Section 4 Example 1

Table 8: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

N/A for this module.

Non-Approved, Not Allowed Algorithms:

N/A for this module.

2.6 Security Function Implementations

Name	Type	Description	Properties	Algorithms
KAS-ECC (TLSv1.2)	KAS-Full	Full KAS-ECC Key Agreement used for TLSv1.2 service	IG:IG D.F Scenario 2 Path 2, split Key Confirmation:No Key Derivation:IG 2.4.B SP 800-135rev1 CVL Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	KAS-ECC-SSC Sp800-56Ar3: (A2999) KDF TLS: (A2999) HMAC-SHA2-256: (A2999) HMAC-SHA2-384: (A2999) SHA2-256: (A2999) SHA2-384: (A2999)
KAS-ECC-KeyGen (TLSv1.2)	CKG KAS-KeyGen	KAS ECC keygen used in	Curves:P-256, P-384, and P-521	Counter DRBG: (A2999)

Name	Type	Description	Properties	Algorithms
		TLSv1.2 service	Encryption Strength:128, 192, or 256 bits	
KTS (TLSv1.2 with AES and HMAC)	KTS-Wrap	KTS via TLSv1.2 service by using AES and HMAC	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	AES-CBC: (A2999) HMAC-SHA2-256: (A2999) HMAC-SHA2-384: (A2999) SHA2-256: (A2999) SHA2-384: (A2999) KDF TLS: (A2999)
KTS (TLSv1.2 with AES-GCM)	KTS-Wrap	KTS via TLSv1.2 service by using AES-GCM	Standard:SP 800-38F IG D.G:Approved Key Wrapping Caveat:Key establishment methodology providing between 128 and 256 bits of security strength	AES-GCM: (A2999) AES-CBC: (A2999)
Session Authentication (IPsec)	MAC	IPSec session authentication		HMAC-SHA-1: (A2999)
Session Authentication (TLSv1.2)	MAC	TLSv1.2 session authentication		HMAC-SHA2-256: (A2999) HMAC-SHA2-384: (A2999) SHA2-256: (A2999) SHA2-384: (A2999)
Session Encryption/Decryption (IPsec)	BC-Auth BC-UnAuth	IPsec session protection		AES-CBC: (A2999) AES-GCM: (A2999)
Session Encryption/Decryption (TLSv1.2)	BC-Auth BC-UnAuth	TLSv1.2 session protection		AES-CBC: (A2999) AES-GCM: (A2999)

Name	Type	Description	Properties	Algorithms
TLS ECDSA KeyGen	AsymKeyPair-KeyGen CKG	ECDSA KeyGen for TLSv1.2		ECDSA KeyGen (FIPS186-4): (A2999) Counter DRBG: (A2999)
TLS ECDSA KeyVer	AsymKeyPair-KeyVer CKG	ECDSA KeyVer for TLSv1.2		ECDSA KeyVer (FIPS186-4): (A2999) Counter DRBG: (A2999)
TLS ECDSA SigGen	DigSig-SigGen	ECDSA SigGen for TLSv1.2		ECDSA SigGen (FIPS186-4): (A2999) Counter DRBG: (A2999) SHA2-256: (A2999) SHA2-384: (A2999) SHA2-512: (A2999)
TLS ECDSA SigVer	DigSig-SigVer	ECDSA SigVer for TLSv1.2		ECDSA SigVer (FIPS186-4): (A2999) SHA2-256: (A2999) SHA2-384: (A2999) SHA2-512: (A2999) SHA-1: (A2999)
TLS RSA SigGen	DigSig-SigGen	RSA SigGen for TLSv1.2		RSA SigGen (FIPS186-4): (A2999) SHA2-256: (A2999) SHA2-384: (A2999) SHA2-512: (A2999)

Name	Type	Description	Properties	Algorithms
TLS RSA SigVer	DigSig-SigVer	RSA SigVer for TLSv1.2		RSA SigVer (FIPS186-4): (A2999) SHA2-256: (A2999) SHA2-384: (A2999) SHA2-512: (A2999) SHA-1: (A2999)
TLSv1.2 Keying Materials Development	KAS-135KDF	TLSv1.2 session keying materials, used to derive TLSv1.2 session keys		KDF TLS: (A2999) HMAC-SHA2-256: (A2999) HMAC-SHA2-384: (A2999) SHA2-256: (A2999) SHA2-384: (A2999)

Table 9: Security Function Implementations

2.7 Algorithm Specific Information

2.7.1 IG C.H Conformance

GCM is used in the context of TLS:
For TLS, The GCM implementation meets Scenario 1 of IG C.H: it is used in a manner compliant with SP 800-52rev2 and in accordance with Section 4 of RFC 5288 for TLS key establishment, and ensures when the nonce_explicit part of the IV exhausts all possible values for a given session key, that a new TLS handshake is initiated per sections 7.4.1.1 and 7.4.1.2 of RFC 5246. During operational testing, the module was tested against an independent version of TLS and found to behave correctly

- From this RFC, the GCM cipher suites in use are TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, and TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384.

In all of the above cases, the nonce explicit is always generated deterministically. AES GCM keys are zeroized when the module is power-cycled. For each new TLS session, a new AES GCM key is established.

2.7.2 IG C.F Conformance

The module utilizes Approved modulus sizes 2048, 3072, and 4096 bits for RSA signatures. This functionality has been CAVP tested as noted above. The minimum number of Miller Rabin tests for each modulus size is implemented according to Table C.2 of FIPS 186-4. For modulus size 4096, the module implements the largest number of Miller-Rabin tests shown in Table C.2. RSA SigVer is CAVP tested for all three supported modulus sizes as noted above. The module does not perform FIPS 186-2 SigVer. All supported modulus sizes are CAVP testable and tested as noted above. The module does not implement RSA key transport in the approved mode.

2.8 RBG and Entropy

Cert Number	Vendor Name
E14	Apple
E15	Apple
E303	Palo Alto Networks, Inc.

Table 10: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Apple corecrypto non-physical entropy source	Non-Physical	iOS 16 on Apple A Series (ARMv8.5-A) A14 Bionic	256 bits	256 bits	A3429 (SHA2-256
Apple corecrypto physical entropy source	Physical	Apple M Series M1	256 bits	256 bits	A1362 (CTR_DRBG AES-256)
Palo Alto Networks DRNG RDSEED Entropy Source - ALDER LAKE-CPU-10 Die	Physical	Intel® CoreTM CPU based on the ALDER LAKE-CPU-10 Die with FCBGA1744 Package Intel® CoreTM CPU based on the ALDER LAKE-CPU-10 Die with FCBGA1781 Package	128 bits	128 bits	A2873 (AES-CBC-MAC)

Table 11: Entropy Sources

For iOS, Linux, macOS, and Windows platforms, the entropy sources provide full entropy output. The DRBG is seeded with 384 bits of output from the entropy source, so it is fully seeded.

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).

2.9 Key Generation

The module implements CKG where seeds for asymmetric keys are produced using the unmodified/direct output of the DRBG.

2.10 Key Establishment

2.10.1 Key Agreement
The module provides the following key/SSP establishment services in the approved mode of operation:

- KAS-ECC Shared Secret Computation
  - The module provides SP800-56Arev3 compliant key establishment according to FIPS 140-3 IG D.F scenario 2 path (1) with KAS-ECC shared secret computation. The shared secret computation provides between 128 and 256 bits of encryption strength.

2.10.2 Key Transport

The module implements the following approved key transport methods as specified in FIPS 140-3 IG D.G, the underlying algorithms of which have been CAVP tested and validated:

-SP 800-38F Key Transport using AES CBC for encryption and HMAC for authentication. (Approved method #2 from IG D.G)
- SP 800-38F Key Transport using AES GCM for encryption and authentication. (Approved method #2 from IG D.G)

2.11 Industry Protocols

The module supports the following industry protocols:

- TLS v1.2
- IPSEC

No parts of the TLS protocol other than the KDF has been tested by the CAVP/CMVP. The module does not implement an IKEv2 KDF, but rather imports the session and authentication keys as described in section 9.4 of this document.

3 Cryptographic Module Interfaces

The modules are multi-chip standalone modules with ports and interfaces as shown below. The modules do not implement a control output interface.

3.1 Ports and Interfaces

Physical Port	Logical Interface(s)	Data That Passes
N/A	Status Output	GUI status window and log files generated and output via GUI
N/A	Data Input	Portal information, keys from OS certificate store or during TLS/IPsec negotiation
N/A	Data Output	Keys for establishing secure sessions such as TLS
N/A	Control Input	GUI, string value from pangps.xml, com.paloaltonetworks.gp.pangps.plist, MDM, or Windows Registry (See Secure Operation section below)

Table 12: Ports and Interfaces

4 Roles, Services, and Authentication

4.1 Authentication Methods

Method Name	Description	Security Mechanism	Strength Each Attempt	Strength per Minute
N/A	N/A	N/A	N/A	N/A

Table 13: Authentication Methods

The module is a level 1 software module, and as such does not support authentication to meet any FIPS 140-3 requirements.

4.2 Roles

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

Table 14: Roles

4.3 Approved Services

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Security Configuration Management	Configures the module with necessary setup details to support	Configuration/ System Logs	Input configuration for various cryptographic functions	Module uses the configuration for cryptographic purposes	None	Crypto Officer - AES GCM IV: R,W,E - CA Certificates: R,W,E

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	VPN establishment (updated via GUI)					- ECDSA Private Keys: R,W,E - ECDSA Public Keys: R,W,E - RSA Private Keys: R,W,E - RSA Public Keys: R,W,E
Self-Tests	Performs on-demand Self-Tests (executed via reboot of platform)	System Logs	Self-test command or rebooting the module	Status of the self-tests	TLS RSA SigVer	Crypto Officer - Software Integrity Verification Key: E
Show Status	Provides information regarding the status of the system (fetched via GUI/)	Configuration/ System Logs	Initiate show status command	Module provides status output of module	None	Crypto Officer
Show Version	Shows the version of the module	Version displayed via System Logs / UI	Input command for version	Module displays version information	None	Crypto Officer
VPN Tunnel	Creates an SSL/IPsec VPN tunnel (executed by operator interactio	System log provide VPN status	Initialize VPN connection	Status of the VPN Tunnel	KAS-ECC (TLSv1.2) KAS-ECC-KeyGen (TLSv1.2) KTS (TLSv1.2 with AES and HMAC) KTS (TLSv1.2	Crypto Officer - AES GCM IV: R,W,E - CA Certificates: W,E - DRBG

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	n with GUI)				with AES-GCM) Session Authentication (IPsec) Session Authentication (TLSv1.2) Session Encryption/Decryption (TLSv1.2) TLS ECDSA KeyGen TLS ECDSA KeyVer TLS ECDSA SigGen TLS ECDSA SigVer TLS RSA SigGen TLS RSA SigVer TLSv1.2 Keying Materials Development Session Encryption/Decryption (IPsec)	Key: G,E,Z - DRBG Seed: G,E,Z - DRBG V: G,E,Z - ECDSA Private Keys: W,E - ECDSA Public Keys: W,E - Entropy Input String: G,E,Z - IPSec Authentication: W,E - IPSec Session Keys: W,E - RSA Private Keys: E - RSA Public Keys: W,E - TLS ECDHE Private Components: G,E,Z - TLS ECDHE Public Components: G,R,W,E,Z - TLS Encryption Keys: G,E,Z - TLS HMAC Keys: G,E,Z

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- TLS Master Secret: G,E,Z - TLS Pre-Master Secret: G,E,Z
Zeroize	Zeroizes all SSPs from the module (Performed via uninstall of the module)	Zeroization indicator	Initiating zeroization command	Status of the zeroization process	None	Crypto Officer - AES GCM IV: Z - CA Certificates: Z - DRBG Key: Z - DRBG Seed: Z - DRBG V: Z - ECDSA Private Keys: Z - ECDSA Public Keys: Z - Entropy Input String: Z - IPSec Authentication: Z - IPSec Session Keys: Z - RSA Private Keys: Z - RSA Public Keys: Z - Software Integrity Verification Key: Z - TLS ECDHE

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Private Components: Z - TLS ECDHE Public Components: Z - TLS Encryption Keys: Z - TLS HMAC Keys: Z - TLS Master Secret: Z - TLS Pre-Master Secret: Z

Table 15: Approved Services

4.4 Non-Approved Services

N/A for this module.

5 Software/Firmware Security

5.1 Integrity Techniques

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 the Windows/macOS/iOS/Android OEs and RSA 3072 with SHA2-384 is used for the Linux OE. The Software Integrity Verification Key is used for this integrity test. The module's executable code is in the form of the compiled software image loaded onto the module.

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.

5.2 Initiate on Demand

The pre-operational self-tests can be initiated by power cycling the module. When this is performed, the module automatically runs the cryptographic algorithm self-tests in addition to the pre-operational software integrity test.

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Modifiable

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.

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.

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 Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
HDD	Non-Volatile Memory	Static
RAM	Volatile Memory	Dynamic

Table 16: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Module Public Key Output	HDD	External (Outside of the Module's Boundary)	Plaintext	Automated	Electronic	
Password/Secret Input via TLSv1.2 encrypted by AES and HMAC	External (Outside of the Module's Boundary)	HDD	Encrypted	Automated	Electronic	KTS (TLSv1.2 with AES and HMAC)
Password/Secret Input via TLSv1.2 encrypted by AES-GCM	External (Outside of the Module's Boundary)	HDD	Encrypted	Automated	Electronic	KTS (TLSv1.2 with AES-GCM)
Peer Public Key Input	External (outside of module's boundary)	HDD	Plaintext	Automated	Electronic	

Table 17: SSP Input-Output Methods

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
Power Cycle / Session Termination	Operator powers the module off or session terminates	Powering off the module or terminating the session will erase all SSPs stored in the RAM of the module and make them non-retrievable.	Power module off
Zeroization Command	CO issues zeroization service	The zeroization command will erase all SSPs stored in the RAM	Uninstalling the module

Zeroization Method	Description	Rationale	Operator Initiation
		or in the Flash of the module and make them non-retrievable.	

Table 18: SSP Zeroization Methods

9.4 SSPs

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
AES GCM IV	IV used for AES GCM functions as per IG C.H Scenario #1	96 bits - 96 bits	Initialization Vector - PSP	Counter DRBG (A2999)		TLSv1.2 Keying Materials Development
CA Certificates	ECDSA/RSA Public key used to extend trust to a root CA, intermediate CA, and left/end entity certificates	2048 - 4096 bits - 112 - 256 bits	Public Key - PSP			ECDSA SigVer (FIPS186-4) (A2999) RSA SigVer (FIPS186-4) (A2999)
DRBG Key	Internal DRBG State	256 bits - 256 bits	DRBG Key - CSP	Counter DRBG (A2999)		Counter DRBG (A2999)
DRBG Seed	DRBG seed coming from the entropy input string used in the generation of random values	384 bits - 384 bits	DRBG Seed - CSP	Entropy source		Counter DRBG (A2999)
DRBG V	Internal DRBG State	128 bits - 128 bits	DRBG Internal State V value - CSP	Counter DRBG (A2999)		Counter DRBG (A2999)
ECDSA Private Keys	ECDSA Private key for generation	128 - 256 bits - 128 -	Private Key - CSP			TLS ECDSA SigGen

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	of signatures and authentication (P-256, P-384, or P-521)	256 bits				
ECDSA Public Keys	ECDSA public keys managed as certificates for the verification of signatures, establishment of TLS, and peer authentication. (P-256/384/521)	128 - 256 bits - 128 - 256 bits	Public Key - PSP			TLS ECDSA SigVer
Entropy Input String	DRBG entropy input string coming from the entropy source used in the generation of random values	112 bits minimum - 112 bits	DRBG - CSP	Entropy source		Counter DRBG (A2999)
IPSec Authentication	(HMAC-SHA-1, 160 bits) Used as part of authentication for IPsec data	160 bits - 160 bits	Session Key - CSP			Session Authentication (IPsec)
IPSec Session Keys	Used to encrypt sessions utilizing IPSec.	128 or 256 bits - 128 or	Session Key - CSP			Session Encryption/Decryption (IPsec)

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
	(AES 128-CBC, 128/256-GCM)	256 bits				
RSA Private Keys	RSA Private keys for generation of signatures, authentication or key establishment. (RSA 2048, 3072, or 4096-bit)	2048 - 4096 bits - 112-150 bits	Private Key - CSP	RSA SigGen (FIPS186-4) (A2999)		TLS RSA SigGen
RSA Public Keys	RSA public keys managed as certificates for the verification of signatures, establishment of TLS, operator authentication and peer authentication. (RSA 2048, 3072, or 4096-bit)	2048 - 4096 bits - 112-150 bits	Public Key - PSP	RSA SigVer (FIPS186-4) (A2999)		TLS RSA SigVer
Software Integrity Verification Key	Used to verify the integrity of the module (Note: This is not considered an SSP)	2048 or 3072 bits - 112-128 bits	Public Key - Neither	Pre-loaded		RSA SigVer (FIPS186-4) (A2999)
TLS ECDHE Private	ECDHE private component used in key	128 - 256 bits - 128 -	Private Key - CSP	KAS-ECC-KeyGen		TLSv1.2 Keying Materials Development

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
Components	agreement (P-256, P-384, P-521)	256 bits		(TLSv1.2)		
TLS ECDHE Public Components	ECDHE public component used in key agreement (P-256/384/521)	128 - 256 bits - 128 - 256 bits	Public Key - PSP	KAS-ECC-KeyGen (TLSv1.2)	KAS-ECC (TLSv1.2)	TLSv1.2 Keying Materials Development
TLS Encryption Keys	AES (128 or 256 bit) keys used in TLS connections (GCM; CBC)	128 - 256 bits - 128 - 256 bits	Session Key - CSP	KDF TLS (A2999)	KAS-ECC (TLSv1.2)	Session Encryption/Decryption (TLSv1.2)
TLS HMAC Keys	HMAC keys used in TLS connections (HMAC-SHA2-256/384) (256, 384 bits)	256 - 384 bits - - 256 - 384 bits -	Session Key - CSP - CSP	KDF TLS (A2999)	KAS-ECC (TLSv1.2)	Session Authentication (TLSv1.2)
TLS Master Secret	Secret value used to derive the TLS session key	384 bits - 384 bits	Master Secret - CSP		KAS-ECC (TLSv1.2)	TLSv1.2 Keying Materials Development
TLS Pre-Master Secret	Secret value used to derive the TLS Master Secret along with client and server random nonces	256 bits, 384 bits, 521 bits - 256 bits, 384 bits, 521 bits	Shared Secret - CSP		KAS-ECC (TLSv1.2)	TLSv1.2 Keying Materials Development

Table 19: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
AES GCM IV		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
CA Certificates	Peer Public Key Input Password/Secret Input via TLSv1.2 encrypted by AES and HMAC Password/Secret Input via TLSv1.2 encrypted by AES-GCM	HDD:Encrypted		Zeroization Command	
DRBG Key		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
DRBG Seed		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
DRBG V		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
ECDSA Private Keys	Password/Secret Input via TLSv1.2 encrypted by AES and HMAC Password/Secret Input via TLSv1.2 encrypted by AES-GCM	HDD:Encrypted		Zeroization Command	
ECDSA Public Keys	Password/Secret Input via TLSv1.2 encrypted by AES and HMAC Password/Secret Input via TLSv1.2 encrypted by AES-GCM	HDD:Encrypted		Zeroization Command	
Entropy Input String		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
IPSec Authentication	Password/Secret Input via TLSv1.2 encrypted by AES-GCM	RAM:Plaintext	Duration of use	Power Cycle / Session Termination	

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
IPSec Session Keys	Password/Secret Input via TLSv1.2 encrypted by AES-GCM	RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
RSA Private Keys	Password/Secret Input via TLSv1.2 encrypted by AES and HMAC Password/Secret Input via TLSv1.2 encrypted by AES-GCM	HDD:Encrypted		Zeroization Command	
RSA Public Keys	Peer Public Key Input Module Public Key Output Password/Secret Input via TLSv1.2 encrypted by AES and HMAC Password/Secret Input via TLSv1.2 encrypted by AES-GCM	HDD:Encrypted		Zeroization Command	
Software Integrity Verification Key		RAM:Plaintext	Duration of use	Zeroization Command	
TLS ECDHE Private Components		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
TLS ECDHE Public Components	Peer Public Key Input Module Public Key Output	RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
TLS Encryption Keys		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
TLS HMAC Keys		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
TLS Master Secret		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	
TLS Pre-Master Secret		RAM:Plaintext	Duration of use	Power Cycle / Session Termination	

Table 20: SSP Table 2

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.

10.1 Pre-Operational Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
RSA	RSA 2048 bits with SHA2-256 or RSA 3072 bits with SHA2-384 (Linux)	KAT	SW/FW Integrity	Self-Test successful	Digital signature verification

Table 21: Pre-Operational Self-Tests

Verified with RSA 2048 with SHA2-256 (Cert. #A2999) or RSA 3072 with SHA2-384 (Cert. #A2999)

Note: The RSA and SHA-256/384 KATs are performed prior to the Software integrity test.

10.2 Conditional Self-Tests

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES GCM Decrypt	256 Bits	KAT	CAST	Self-test output message	Decrypt	After each power-on or reloading module
AES GCM Encrypt	256 Bits	KAT	CAST	Self-test output message	Encrypt	After each power-on or reloading module
AES-ECB Decrypt	128 Bits	KAT	CAST	Self-test output message	Decrypt	After each power-on or reloading module
Counter DRBG	N/A	KAT	CAST	Self-test output message	SP 800-90Arev1 Instantiate/Generate/Reseed Known Answer Tests	After each power-on or

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
						reloading module
ECDSA KeyGen	256 Bit Minimum	PCT	PCT	System log messages	ECDSA/KAS-ECC pairwise consistency test	On session
ECDSA Sign	256 Bits	KAT	CAST	Self-test output message	Sign	After each power-on or reloading module
ECDSA Verify	256 Bits	KAT	CAST	Self-test output message	Verify	After each power-on or reloading module
HMAC-SHA-1	160 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or reloading module
HMAC-SHA2-256	256 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or reloading module
HMAC-SHA2-384	384 Bits	KAT	CAST	Self-test output message	Keyed Hash	After each power-on or reloading module
RSA Sign	2048 Bits	KAT	CAST	Self-test output message	Sign	After each power-on or reloading module
RSA Verify	2048 Bits	KAT	CAST	Self-test output message	Verify	After each power-on or reloading module
SHA-1	160 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or reloading module

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
SHA2-256	256 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or reloading module
SHA2-384	384 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or reloading module
SHA2-512	512 Bits	KAT	CAST	Self-test output message	Hash	After each power-on or reloading module
SP 800-135rev1 TLS 1.2 with SHA2-256 KDF	N/A	KAT	CAST	Self-test output message	TLSv1.2 with SHA2-256	After each power-on or reloading module
SP 800-56A Rev 3 Assurance Tests	N/A	Critical Functions	Critical Function	System log messages	Assurance tests for SP 800-56A Rev3	On session
SP 800-56Ar3 KAS-ECC-SSC	256 Bits	KAT	CAST	Self-test output message	KAS Computation	After each power-on or reloading module
SP 800-90B RCT/APT Health Tests on Entropy Source	N/A	Fault-Detection	CAST	Self-test output message	Health tests done on entropy source	After each power-on or reloading module

Table 22: Conditional Self-Tests

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
RSA	KAT	SW/FW Integrity	On Demand	Manually or Scheduled

Table 23: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES GCM Decrypt	KAT	CAST	On Demand	Manually
AES GCM Encrypt	KAT	CAST	On Demand	Manually
AES-ECB Decrypt	KAT	CAST	On Demand	Manually
Counter DRBG	KAT	CAST	On Demand	Manually
ECDSA KeyGen	PCT	PCT	On session	On session
ECDSA Sign	KAT	CAST	On Demand	Manually
ECDSA Verify	KAT	CAST	On Demand	Manually
HMAC-SHA-1	KAT	CAST	On Demand	Manually
HMAC-SHA2-256	KAT	CAST	On Demand	Manually
HMAC-SHA2-384	KAT	CAST	On Demand	Manually
RSA Sign	KAT	CAST	On Demand	Manually
RSA Verify	KAT	CAST	On Demand	Manually
SHA-1	KAT	CAST	On Demand	Manually
SHA2-256	KAT	CAST	On Demand	Manually
SHA2-384	KAT	CAST	On Demand	Manually
SHA2-512	KAT	CAST	On Demand	Manually
SP 800-135rev1 TLS 1.2 with SHA2-256 KDF	KAT	CAST	On Demand	Manually
SP 800-56A Rev 3 Assurance Tests	Critical Functions	Critical Function	On session	On session
SP 800-56Ar3 KAS-ECC-SSC	KAT	CAST	On Demand	Manually
SP 800-90B RCT/APT Health Tests on Entropy Source	Fault-Detection	CAST	On Demand	Manually

Table 24: Conditional Periodic Information

10.4 Error States

Name	Description	Conditions	Recovery Method	Indicator
Conditional Cryptographic Algorithm Self-Test Failure	KAT failure	KAT failure	N/A	System prints log with error message
Conditional Test Failure	Pairwise conditional test failure	Pairwise test failure	N/A	System prints log with error message
Integrity Test Failure	Signature verification fails on software integrity test	Signature verification failure	N/A	System prints log with error message

Table 25: Error States

In case of Integrity test failure, system prints log with error message “Integrity Test Failure”. In case of Conditional Self-Test failure, the system prints log with error message “Conditional Test Failure”.

10.5 Operator Initiation of Self-Tests

Perform a power cycle or reload module.

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.

11.1 Installation, Initialization, and Startup Procedures

The vendor provided life-cycle assurance documentation describes configuration management, design, finite state model, development, testing, delivery & operation, end of life procedures, and guidance. 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 prepare 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 prepare 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

Note: Disabling FIPS-CC mode causes a complete factory reset, which is described in the Zeroization section below. Failure to follow these instructions would result in the module operating in a non-compliant state.

11.2 Administrator Guidance

The Administrator Guidance can be obtained from Palo Alto Network's public site:

https://docs.paloaltonetworks.com/globalprotect

11.3 End of Life

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

12 Mitigation of Other Attacks

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