# WildFire 10.1 WF-500

FIPS 140-3 Non-Proprietary Security Policy

Version: 0.7

Revision Date: August 28, 2024

**Palo Alto Networks, Inc.**
www.paloaltonetworks.com

# Table of Contents

# 1. General

The Wildfire 10.1 WF-500 from Palo Alto Networks Inc., hereafter referred to as "WildFire" or the "cryptographic module" is a multi-chip standalone hardware cryptographic module designed to fulfill FIPS 140-3 level 2 requirements. The WildFire 10.1 WF-500 module identifies unknown malware, zero-day exploits, and Advanced Persistent Threats (APTs) through dynamic analysis, and automatically disseminates protection in near real-time to help security teams meet the challenge of advanced cyber-attacks.

Unknown files are analyzed by WildFire (WF) in a scalable sandbox environment where new threats are identified, and protections are automatically developed and delivered in the form of an update. The result is a unique, closed loop approach to controlling cyber threats that begins with positive security controls to reduce the attack surface, inspection of all traffic, ports, and protocols to block all known threats, and rapid detection of unknown threats by observing their actual behavior.

The cryptographic module meets the overall requirements applicable to Level 2 security of FIPS 140-3.

*Table 1 – Security Levels*

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

# 2. Cryptographic Module Specification

The Palo Alto Networks, Inc. WF-500 is a multi-chip standalone module. The module is shown in Figure 1. The module boundary is the outer chassis enclosure. The cryptographic boundary includes all the logical components of the modules and the physical perimeter is the outer perimeter of the enclosure of the WF-500. Figure 2 through Figure 5 provide images of the module with the FIPS kit's opacity shields in place. See the Physical Security section for details regarding the module's physical security mechanisms.

*Table 2 - Cryptographic Module Tested Configuration*

| Model | Hardware [Part Number and Version] | Firmware Version | Distinguishing Features |
|---|---|---|---|
| WF-500 | 910-000097<br>Physical Kit: 920-000145 | 10.1.5 | See 'Cryptographic Module Interfaces' Section |

## Approved Mode of Operation

The module supports only one mode, which is the Approved mode of operation (FIPS-CC mode). The following section details the procedure necessary to place the module into the Approved mode of operation.

The following procedure will initialize the module into the Approved mode of operation:

- Install module and interface connections in addition to the physical kit.
- The tamper-evident seals and opacity shields must be installed as per Appendix A for the module to operate in the Approved mode of operation.
- Apply power to the device.
- Establish a serial connection to the console port and command the module to enter into maintenance mode.
  - During initial boot up, break the boot sequence via the console port connection (by pressing the maint button when instructed to do so) to access the main menu.
- Select "Continue."
- Select the "Set FIPS-CC" option, and press enter.
- Select "Enable FIPS-CC Mode," and press enter.
- When prompted, select "Reboot" and the module will re-initialize and continue into the Approved mode.
- The module will reboot.
- In the Approved mode, the console port is available only as a status output port.
- Once the module has finished booting, the Crypto Officer can authenticate using the default credentials that come with the module
  - Once authenticated, the module will automatically require the operator to change their password; and the default credential is overwritten

The module will automatically indicate the Approved mode of operation in the following manner:

- Status output interface will indicate "**** FIPS-CC MODE ENABLED ****" via the CLI session.
- Status output interface will indicate "FIPS-CC mode enabled successfully" via the console port.

Should one or more power-up self-tests fail, the module will not enter the Approved mode of operation. Feedback will consist of:

- The module will output "FIPS-CC failure."
- The module will reboot and enter a state in which the reason for the reboot can be determined by following the on-screen instructions.

Note: Disabling “FIPS-CC” mode causes a complete factory reset, which is described in the Zeroization section below.

The module does not support a degraded mode of operation.

### Non-Compliant State

Failure to follow the directions in the Approved Mode of Operation above or rules noted in Section 11 will result in the module operating in a non-compliant state, which is considered out of scope of this validation.

### Zeroization

To initiate the zeroization service, perform the following steps:

- Access the module’s CLI via SSH, and command the module to enter maintenance mode; the module will reboot
  - Note: Establish a serial connection to the console port
- After reboot, select “Continue.”
- Select “Factory Reset.”
- The module will perform a zeroization, and provide the following message once complete:
  - “Factory Reset Status: Success”

### Approved and Allowed Algorithms

The cryptographic module has the following CAVP certificates:

*Table 3 - Approved Algorithms*[1]

| CAVP Cert | Algorithm and Standard | Mode/Method | Description / Key Size(s) / Key Strength(s) | Use / Function |
|---|---|---|---|---|
| A2137 | AES-CBC [SP 800-38A] | CBC | 128, 192 and 256 bits | Encryption, Decryption |
| A2137 | AES-CFB128 [SP 800-38A] | CFB128 | 128 bits | Encryption, Decryption |
| A2137 | AES-CTR [SP 800-38A] | CTR | 128, 192 and 256 bits | Encryption, Decryption |
| A2137 | AES-GCM [SP 800-38D] | GCM* | 128 and 256 bits | Encryption, Decryption |
| A2137 | Counter DRBG [SP 800-90Arev1] | CTR DRBG | AES 256 bits with Derivation Function Enabled | Random Bit Generator |
| A2137 | ECDSA KeyGen (FIPS 186-4) | ECDSA KeyGen | P-256, P-384, P-521 | Key Generation |
| A2137 | ECDSA KeyVer (FIPS 186-4) | ECDSA KeyVer | P-256, P-384, P-521 | Public Key Validation |

[1] Only the algorithms, modes, and key sizes specified in this table are used by the module. The CAVP certificate may contain more tested options than listed in this table.

| CAVP Cert | Algorithm and Standard | Mode/Method | Description / Key Size(s) / Key Strength(s) | Use / Function |
|---|---|---|---|---|
| A2137 | ECDSA SigGen (FIPS 186-4) | ECDSA SigGen | P-256, P-384, P-521 with SHA2-224, SHA2-256, SHA2-384, and SHA2-512 | Signature Generation |
| A2137 | ECDSA SigVer (FIPS 186-4) | ECDSA SigVer | P-256, P-384, P-521 with SHA-1, SHA2-224, SHA2-256, SHA2-384, and SHA2-512 | Signature Verification |
| A2137 | HMAC-SHA-1 [FIPS 198-1] | HMAC | HMAC-SHA-1 with λ=160 | For Protocols |
| A2137 | HMAC-SHA2-224 [FIPS 198-1] | HMAC | HMAC-SHA2-224 with λ=224 | For Protocols |
| A2137 | HMAC-SHA2-256 [FIPS 198-1] | HMAC | HMAC-SHA2-256 with λ=256 | For Protocols |
| A2137 | HMAC-SHA2-384 [FIPS 198-1] | HMAC | HMAC-SHA2-384 with λ=384 | For Protocols |
| A2137 | HMAC-SHA2-512 [FIPS 198-1] | HMAC | HMAC-SHA2-512 with λ=512 | For Protocols |
| A2137 | KAS-ECC-SSC SP800-56Ar3 | KAS | Ephemeral Unified Model: P-256/P-384/P-521 | Key Agreement, Shared Secret Computation |
| A2137 | KAS-FFC-SSC SP 800-56Ar3 | KAS | dhEphem: MODP-2048 | Key Agreement, Shared Secret Computation |
| A2137 | KDF IKEv2 [SP 800-135rev1] (CVL) | IKEv2 KDF | SHA2-256, SHA2-384, SHA2-512 | IKEv2 |
| A2137 | KDF SNMP [SP 800-135rev1] (CVL) | SNMPv3 KDF | Engine ID: 80001F88043030303030343935323630 | SNMPv3 |
| A2137 | KDF SSH [SP 800-135rev1] (CVL) | SSHv2 KDF | SHA-1, SHA2-256, SHA2-512 | SSH |
| A2137 | KDF TLS [SP 800-135rev1] (CVL) | TLS 1.0/1.1 KDF, TLS1.2 KDF | TLS v1.0/1.1<br>TLS v1.2 Hash Algorithm: SHA2-256, SHA2-384 | TLS |
| A2137 | RSA KeyGen (FIPS 186-4) | RSA KeyGen (FIPS 186-4) | 2048, 3072, and 4096 bits | Key Pair Generation |
| A2137 | RSA SigGen (FIPS 186-4) | RSA SigGen (FIPS 186-4) | (ANSI X9.31, RSASSA-PKCS1_v1-5, RSASSA-PSS): 2048, 3072, and 4096-bit with hashes SHA2-256/384/512 | Signature Generation |
| A2137 | RSA SigVer (FIPS 186-4) | RSA SigVer (FIPS 186-4) | (ANSI X9.31, RSASSA-PKCS1_v1-5, RSASSA-PSS): 2048, 3072, 4096-bit (per IG C.F) with hashes SHA-1 and SHA2-224+++/256/384/512 (Signature Verification)<br>+++ This Hash algorithm is not supported for ANSI X9.31 | Signature Verification |
| A2137 | SHA-1 [FIPS 180-4] | SHA | SHA-1 | Digital Signature Generation/Verification<br>Non-Digital Signature Applications (e.g. component of HMAC) |
| A2137 | SHA2-224 [FIPS 180-4] | SHA2 | SHA-224 | Digital Signature Generation/Verification |

| CAVP Cert | Algorithm and Standard | Mode/Method | Description / Key Size(s) / Key Strength(s) | Use / Function |
|---|---|---|---|---|
| | | | | Non-Digital Signature Applications (e.g. component of HMAC) |
| A2137 | SHA2-256 [FIPS 180-4] | SHA2 | SHA-256 | Digital Signature Generation/Verification<br>Non-Digital Signature Applications (e.g. component of HMAC) |
| A2137 | SHA2-384 [FIPS 180-4] | SHA2 | SHA-384 | Digital Signature Generation/Verification<br>Non-Digital Signature Applications (e.g. component of HMAC) |
| A2137 | SHA2-512 [FIPS 180-4] | SHA2 | SHA-512 | Digital Signature Generation/Verification<br>Non-Digital Signature Applications (e.g. component of HMAC) |
| A2137 | Safe Primes Key Generation [RFC 3526] | Safe Primes Key Generation | MODP-2048 | Safe Primes Key Generation |
| A2137 | Safe Primes Key Verification [RFC 3526] | Safe Primes Key Verification | MODP-2048 | Safe Primes Key Verification |
| AES Cert. #A2137 and HMAC Cert. #A2137 | KTS<br>[SP 800-38F] | SP 800-38A, FIPS 198-1, and SP 800-38F. KTS (key wrapping and unwrapping) per IG D.G. | 128, 192, and 256-bit keys providing 128, 192, or 256 bits of encryption strength | Key Wrapping. AES-CBC or AES-CTR with HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, or HMAC-SHA2-512 |
| AES-GCM Cert. #A2137 | KTS<br>[SP 800-38F] | SP 800-38D and SP 800-38F. KTS (key wrapping and unwrapping) per IG D.G. | 128 and 256-bit keys providing 128 or 256 bits of encryption strength | Key Wrapping. AES-GCM. |
| ESV Cert. #E130 | SP 800-90B | ESV | Palo Alto Networks RTC Entropy Source | Entropy |
| KAS-ECC-SSC Cert. #A2137, KDF IKEv2 Cert. #A2137 | KAS [SP 800-56Arev3] | SP 800-56Arev3. KAS-ECC per IG D.F Scenario 2 path (2). | P-256, P-384 curves providing 128 or 192 bits of encryption strength | Key Exchange with protocol KDF |
| KAS-ECC-SSC Cert. #A2137, KDF SSH Cert. #A2137 | 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 |
| KAS-ECC-SSC Cert. #A2137, KDF TLS Cert. #A2137 | 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 |
| KAS-FFC-SSC Cert. #A2137, KDF IKEv2 Cert. #A2137 | KAS [SP 800-56Arev3] | SP 800-56Arev3. KAS-FFC per IG D.F Scenario 2 path (2). | 2048-bit key providing 112 bits of encryption strength | Key Exchange with protocol KDF |
| KAS-FFC-SSC Cert. #A2137, KDF SSH Cert. #A2137 | KAS [SP 800-56Arev3] | SP 800-56Arev3. KAS-FFC per IG D.F Scenario 2 path (2). | 2048-bit key providing 112 bits of encryption strength | Key Exchange with protocol KDF |
| KAS-FFC-SSC Cert. #A2137, KDF TLS Cert. #A2137 | KAS [SP 800-56Arev3] | SP 800-56Arev3. KAS-FFC per IG D.F Scenario 2 path (2). | 2048-bit key providing 112 bits of encryption strength | Key Exchange with protocol KDF |

| CAVP Cert | Algorithm and Standard | Mode/Method | Description / Key Size(s) / Key Strength(s) | Use / Function |
|---|---|---|---|---|
| Vendor Affirmed | CKG<br>[ SP 800-133rev2] | Section 5.1, Section 5.2 | Cryptographic Key Generation; SP 800-133 and IG D.I. | Key Generation<br>Note: The seeds used for asymmetric key pair generation are produced using the unmodified/direct output of the DRBG |

*The module is compliant to IG C.H: GCM is used in the context of TLS, IPsec/IKEv2, and SSH:

- For TLS, The GCM implementation meets Scenario 1 of IG C.H: it is used in a manner compliant with SP 800-52 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 5288, 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.)
- For IPsec/IKEv2, The GCM implementation meets Scenario 1 of IG C.H: it is used in a manner compliant with RFCs 4106 and 7296 (RFC 5282 is not applicable, as the module does not use GCM within IKEv2 itself), and ensures when the module exhausts all possible values for a given session key that this triggers a rekey condition. During operational testing, the module was tested against an independent version of IPsec with IKEv2 and found to behave correctly.
- For SSH, the module meets Scenario 1 of IG C.H. The module conforms to RFCs 4252, 4253, and 5647. The fixed field is 4-byte in length and is derived using the SSH KDF; this ensures the fixed field is unique for any given GCM session. The invocation field is 8-byte in length and is incremented for each invocation of GCM; this prevents the IV from repeating until the entire invocation field space of 264 is exhausted, which can take hundreds of years. (In FIPS-CC Mode, SSH rekey is automatically configured at 1 GB of data or 1 hour, whichever comes first.)

In all 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 or SSH session, a new AES GCM key is established.

The module is compliant to IG C.F:

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.

The cryptographic module does not support Non-Approved Algorithms Allowed in the Approved Mode of Operation.

The cryptographic module supports the following non-Approved algorithms that are allowed for use in the Approved mode of operation:

*Table 4 - Non-Approved Algorithms Allowed in the Approved Mode of Operation with No Security Claimed*

| Algorithm | Caveat | Use / Function |
|---|---|---|
| MD5 | Only allowed as the PRF in TLSv1.1 per IG 2.4.AOnly allowed as the PRF in TLSv1.0 and v1.1 per IG 2.4.A | Message digest used in TLSv1.0 /v1.1 KDF only |

The cryptographic module supports the following non-approved algorithms not allowed for use in the approved mode of operation.

*Table 5 - Supported Protocols in the Approved Mode*

| Supported Protocols* |
|---|
| TLS v1.1, 1.2 |
| SSHv2 |
| SNMPv3 |
| IPsec and IKEv2 |

(*): These protocols have not been tested or reviewed by the CMVP or the CAVP.

(**): See vendor imposed security rule in Security Rules section

The module does not have any algorithms that fall under:

- Non-Approved Algorithms Not Allowed in the Approved Mode of Operation
- Non-Approved Algorithms

## Module Diagrams

Figures 1 - 5 depict the module and its interfaces. The cryptographic boundary includes the physical perimeter of the enclosure of the appliance with the physical kit installed and all logical components within.

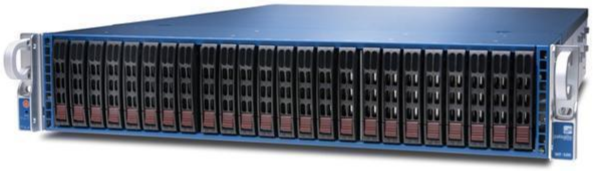

Figure 1 - Front view of WF-500

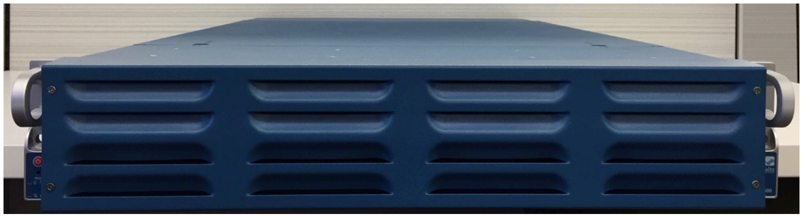

Figure 2 - Front view of WF-500 with opacity shield

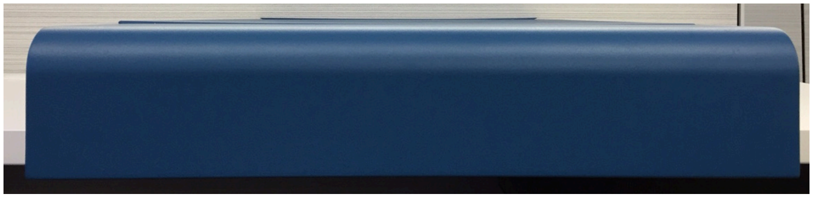

Figure 3 - Rear view of WF-500 with opacity shield

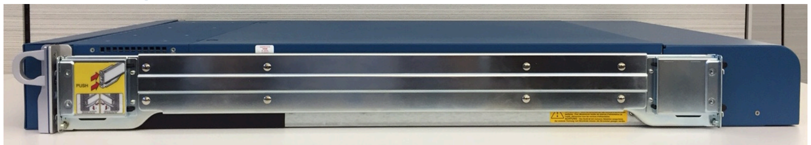

Figure 4 - Right side of WF-500 with opacity shields

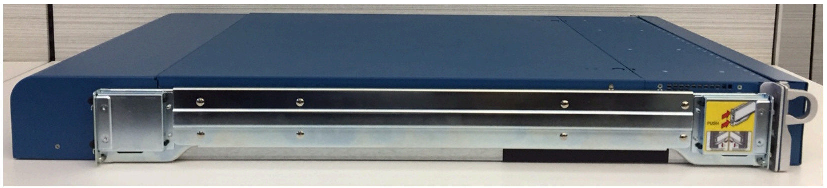

Figure 5 - Left side of WF-500 with opacity shields

# 3. Cryptographic Module Interfaces

The WF-500 provides the following ports and interfaces:

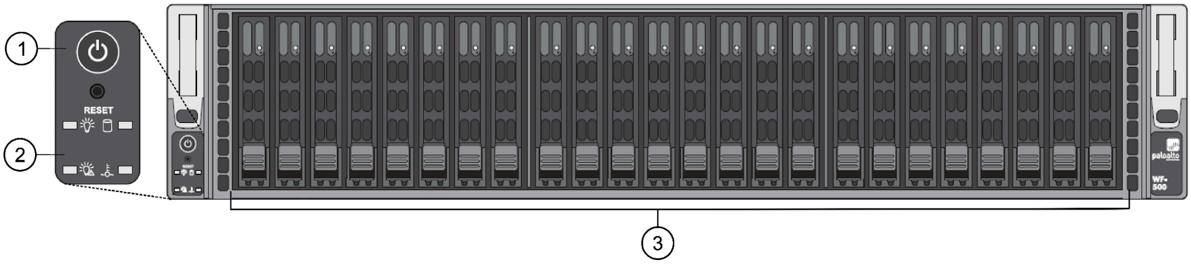

Figure 6 - Front Ports and Interfaces

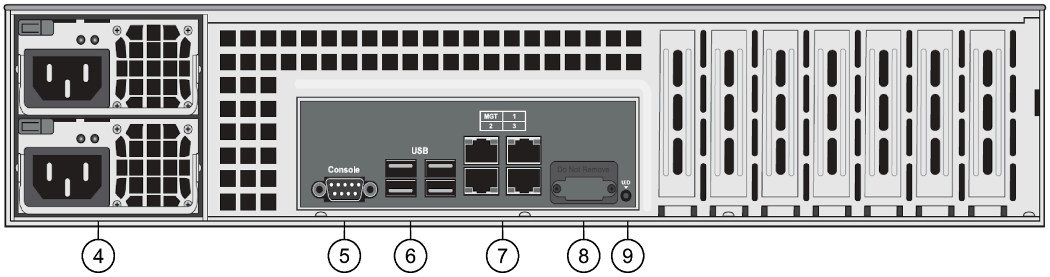

Figure 7 - Rear Ports and Interfaces

*Table 6 - Ports and Interfaces*

| Physical Port | | Logical Interface | Data that passes over port/interface |
|---|---|---|---|
| 1 | Power Button and Reset | Control input | None |
| 2 | Front LED Panel | Status output | LED information for module's status |
| 3 | Drive LEDs | Status output | LED information |
| 4 | Power | Power Input | None |
| 5 | DB9 | Data input, Control input, Data output, Status output, Control output | Console access *(Note: In the Approved mode, the Console port is only available as Status output)* |
| 6 | USB | Disabled except for power | None -- disabled except for power out |
| 7 | RJ45 | Data input, Control input, Data output, Status output, Control output | Used for TLS and SSH |
| | | Data input, Data Output | Used for TLS and SSH |
| | | Data input, Control input, Data output, Status output | SSH and or IKE/IPsec |
| | | Data input, Control input, Data output, Status output | SSH and or IKE/IPsec |
| 8 | UID Button with LED | Control input, Status output | LED information to help identify device |

NOTE: Port number 8 (VGA) is omitted intentionally because it is disabled and so N/A.

# 4. Roles, Services, and Authentication

### Services

When initialized into the Approved mode of operation, all authenticated services are accessed via SSH or TLS sessions. Approved and allowed algorithms, relevant CSPs and public keys related to these protocols are accessed to support the following services. CSP access by services is further described in the following tables.

The Crypto-Officer (CO) may access all services and has the ability to define multiple Crypto-Officer roles. The User role provides read-only access to the system via the System Audit service. The Peer-to-Peer VPN role consists in managing the establishment of VPN connections between several WildFire WF-500 modules.

*Table 7 – Roles, Service Commands, Input and Output*

| Role | Service | Input | Output |
|---|---|---|---|
| CO | Show Version | Query module for version | Module provides version |
| CO | System Operational Management | Configuring and managing networking parameter configuration, logging configuration, and other non-security relevant configuration via CLI | Confirmation of service via System Logs |
| CO | System Configuration Management | Configuring and managing cryptographic parameters and setting/modifying security policy, including creating User accounts and additional CO accounts via CLI | Confirmation of service via System Logs |
| CO | Data Analysis Management | Configure data submission, analysis and reporting functions via CLI | Confirmation of service via System Logs |
| CO | Check Status | Query status of the module via CLI | Module status information via CLI or System Logs |
| User | System Audit | View the System Logs via CLI | System Logs |
| Peer-to-Peer VPN | IKE/IPsec configuration | Initialize VPN connection | Confirmation of service via System Logs |
| Unauthenticated | Zeroize | Initialize Factory Reset via Maintenance Mode | Console Output |
| Unauthenticated | Self-Tests | Power removal | Console Output |
| Unauthenticated | Show Status | N/A | LEDs |

| | | | |
|---|---|---|---|
| CO | Configuration Management | Configuring and managing cryptographic parameters and setting/modifying communication | Confirmation of service via Configuration/System Logs |

## Assumption of Roles

The module supports distinct operator roles.  The cryptographic module enforces the separation of roles using unique authentication credentials associated with operator accounts.

The module supports concurrent operators with identity-based authentication.

The module does not provide a maintenance role or bypass capability.

*Table 8 – Roles and Authentication*

| Role | Authentication Method | Authentication Strength |
|---|---|---|
| Crypto-Officer (CO) | Username/password and/or certificate/public key-based authentication | Password-based<br>Minimum length is eight (8) characters[2] (95 possible characters). The probability that a random attempt will succeed or a false acceptance will occur is $1/(95^8)$ which is less than 1/1,000,000.  The probability of successfully authenticating to the module within one minute is $10/(95^8)$, which is less than 1/100,000.  The module's configuration supports at most ten failed attempts to authenticate in a one-minute period.<br>Certificate/Public key-based<br>The security modules support public-key based authentication using RSA 2048 and certificate-based authentication using RSA 2048, RSA 3072, RSA 4096, ECDSA P-256, P-384, or P-521.<br>The minimum equivalent strength supported is 112 bits.  The probability that a random attempt will succeed is $1/(2^{112})$ which is less than 1/1,000,000.  The probability of successfully authenticating to the module within a one minute period is $3{,}600{,}000/(2^{112})$, which is less than 1/100,000.  The module supports at most 60,000 new sessions per second to authenticate in a one-minute period. |
| User | Username/password and/or certificate/public key-based authentication | |
| Peer-to-peer VPN | Username/password and/or certificate-based authentication | Certificate/Public key-based<br>The security modules support public-key based authentication using RSA 2048 and certificate-based authentication using RSA |

[2] In FIPS-CC Mode, the module checks and enforces the minimum password length of eight (8) as specified in SP 800-63B. Passwords are securely stored hashed with salt value, with very restricted access control, and rate limiting mechanism for authentication attempts.

| | | |
|---|---|---|
| | | 2048, RSA 3072, RSA 4096, ECDSA P-256, P-384, or P-521.<br>The minimum equivalent strength supported is 112 bits. The probability that a random attempt will succeed is 1/($2^{112}$) which is less than 1/1,000,000. The probability of successfully authenticating to the module within a one minute period is 3,600,000/($2^{112}$), which is less than 1/100,000. The module supports at most 60,000 new sessions per second to authenticate in a one-minute period. |

## CSP Access Rights

The following table defines the access to CSPs and the different module services. While in the Approved mode, all authenticated services and CSPs are accessed via authenticated TLS or SSH sessions. Approved and allowed algorithms, relevant CSPs, and public keys related to these protocols are used to access the services as listed in Table 15. The modes of access shown in the table are defined as:

*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.*

*Table 9 – Approved Services*

| Service | Description | Approved Security Functions | | Keys and/or SSPs | Roles | Access rights to Keys and/or SSPs | Indicator |
|---|---|---|---|---|---|---|---|
| Show Version | Query the module to display the version | N/A | | N/A | CO | N/A | Version displayed via System Logs / CLI |
| System Operational Management | Perform system management functions including firmware updates, licensing, diagnostics and debug functions. | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) | | RSA Private Keys | CO | G/W/E | System Logs |
| | | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) | | ECDSA Private Keys | | G/W/E | |
| | | KAS | KDF TLS (CVL) | TLS Pre-Master Secret | | G/E/Z | |
| | | | KDF TLS (CVL) | TLS Master Secret | | G/E/Z | |
| | | | CKG, ECDSA KeyGen (FIPS 186-4), ECDSA KeyVer (FIPS 186-4), KAS-ECC-SSC, KAS-FFC-SSC, Safe Primes Key Generation, Safe | TLS DHE/ECDHE Private Components | | G/E/Z | |
| | | | | TLS DHE/ECDHE Public Components | | G/E/R/W/Z | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | Primes Key Verification | | | | |
| | | KTS | HMAC-SHA2-256<br>HMAC-SHA2-384 | TLS HMAC Keys | | G/E/Z | |
| | | | AES-CBC | TLS Encryption Keys | | G/E/Z | |
| | | KTS | AES-GCM | | | | |
| | | KAS | KDF SSH (CVL) | SSH DHE/ECDHE Private Components | | G/E/Z | |
| | | | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Safe Primes Key Generation, Safe Primes Key Verification | SSH DHE/ECDHE Public Components | | G/E/R/W/Z | |
| | | KTS | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512 | SSH Session Authentication Keys | | G/E/Z | |
| | | | AES-CBC,<br>AES-CTR | SSH Session Encryption Keys | | G/E/Z | |
| | | KTS | AES-GCM | | | | |
| | | N/A | | CO, User Password | | G/E/W | |
| | | Counter DRBG, ESV | | Entropy Input String | | G/E | |
| | | | | DRBG Seed | | | |
| | | | | DRBG V | | | |
| | | | | DRBG Key | | | |
| | | KAS | KDF IKEv2 (CVL) | IPSec/IKE DHE/ECDHE Public Components | | G/E/Z | |
| | | | CKG,<br>ECDSA KeyGen (FIPS 186-4),<br>ECDSA KeyVer (FIPS 186-4),<br>KAS-ECC-SSC,<br>KAS-FFC-SSC,<br>Safe Primes Key Generation, Safe Primes Key Verification | IPSec/IKE DHE/ECDHE Private Components | | G/E/Z | |
| | | KTS | HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 | IPSec/IKE Authentication Keys | | G/E/Z | |
| | | | AES-CBC | IPSec/IKE Session Keys | | | |
| | | KTS | AES-GCM | IPSec/IKE Session Keys | | G/E/Z | |
| | | N/A | | Protocol Secrets | | W/E | |
| | | RSA SigVer (FIPS 186-4) | | RSA Public Keys | | G/R/E/W | |
| | | ECDSA SigVer (FIPS 186-4) | | ECDSA Public Keys | | G/R/E/W | |
| | | RSA SigVer (FIPS 186-4) | | SSH Client Public Key | | W/E | |
| | | RSA SigVer (FIPS 186-4)<br>ECDSA SigVer (FIPS 186-4) | | SSH Host Public Key | | G/R/E/W | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | HMAC-SHA2-256,<br>ECDSA SigVer<br>(FIPS 186-4) | | Firmware Integrity Check Key | | E | |
| | | RSA SigVer (FIPS 186-4) | | Public key for firmware load test | | W/E | |
| System Configuration Management | Presents configuration options for management interfaces and communication for peer services.<br><br>Import, Export, Save, Load, revert and validate configurations and state.<br><br>Define access control methods via admin role profiles, configure administrators/users, and password profiles.<br><br>Configure operators and authentication profiles. | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) | | RSA Private Keys | CO | G/W/E | System Logs |
| | | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) | | ECDSA Private Keys | | G/W/E | |
| | | KAS | KDF TLS (CVL) | TLS Pre-Master Secret | | G/E/Z | |
| | | | KDF TLS (CVL) | TLS Master Secret | | G/E/Z | |
| | | | CKG,<br>ECDSA KeyGen (FIPS 186-4),<br>ECDSA KeyVer (FIPS 186-4),<br>KAS-ECC-SSC,<br>KAS-FFC-SSC,<br>Safe Primes Key Generation, Safe Primes Key Verification | TLS DHE/ECDHE Private Components | | G/E/Z | |
| | | | | TLS DHE/ECDHE Public Components | | G/E/R/W/Z | |
| | | KAS | KDF SSH (CVL) | SSH DHE/ECDHE Private Components | | G/E/Z | |
| | | | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Safe Primes Key Generation, Safe Primes Key Verification | SSH DHE/ECDHE Public Components | | G/E/R/W/Z | |
| | | KTS | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512 | SSH Session Authentication Keys | | G/E/Z | |
| | | | AES-CBC,<br>AES-CTR | SSH Session Encryption Keys | | G/E/Z | |
| | | KTS | AES-GCM | | | | |
| | | N/A | | CO, User Password | | G/E/W | |
| | | Counter DRBG, ESV | | Entropy Input String | | G/E | |
| | | | | DRBG Seed | | | |
| | | | | DRBG V | | | |
| | | | | DRBG Key | | | |
| | | KDF SNMP (CVL) | | SNMPv3 Authentication Secret | | W/E | |
| | | KDF SNMP (CVL) | | SNMPv3 Privacy Secret | | W/E | |
| | | HMAC-SHA-1<br>HMAC-SHA2-224<br>HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 | | Authentication Key | | G/E/Z | |
| | | AES-CFB128 | | Session Key | | G/E/Z | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | KAS | KDF IKEv2 (CVL) | IPSec/IKE DHE/ECDHE Public Components | | G/E/Z | |
| | | | CKG,<br>ECDSA KeyGen (FIPS 186-4),<br>ECDSA KeyVer (FIPS 186-4),<br>KAS-ECC-SSC,<br>KAS-FFC-SSC,<br>Safe Primes Key Generation, Safe Primes Key Verification | IPSec/IKE DHE/ECDHE Private Components | | G/E/Z | |
| | | KTS | HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 | IPSec/IKE Authentication Keys | | G/E/Z | |
| | | | AES-CBC | IPSec/IKE Session Keys | | | |
| | | KTS | AES-GCM | IPSec/IKE Session Keys | | G/E/Z | |
| | | N/A | | Protocol Secrets | | W/E | |
| | | RSA SigVer (FIPS 186-4)<br>ECDSA SigVer (FIPS 186-4) | | SSH Host Public Key | | G/R/E/W | |
| | | HMAC-SHA2-256,<br>ECDSA SigVer (FIPS 186-4) | | Firmware Integrity Check Key | | E | |
| Data Analysis Management | Configure data submission, analysis and reporting functions. | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) | | RSA Private Keys | CO | G/W/E | System Logs |
| | | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) | | ECDSA Private Keys | | G/W/E | |
| | | KAS | KDF TLS (CVL) | TLS Pre-Master Secret | | G/E/Z | |
| | | | KDF TLS (CVL) | TLS Master Secret | | G/E/Z | |
| | | | CKG,<br>ECDSA KeyGen (FIPS 186-4),<br>ECDSA KeyVer (FIPS 186-4),<br>KAS-ECC-SSC,<br>KAS-FFC-SSC,<br>Safe Primes Key Generation, Safe Primes Key Verification | TLS DHE/ECDHE Private Components | | G/E/Z | |
| | | | | TLS DHE/ECDHE Public Components | | G/E/R/W/Z | |
| | | KTS | HMAC-SHA2-256<br>HMAC-SHA2-384 | TLS HMAC Keys | | G/E/Z | |
| | | | AES-CBC | TLS Encryption Keys | | G/E/Z | |
| | | KTS | AES-GCM | | | | |
| | | KAS | KDF SSH (CVL) | SSH DHE/ECDHE Private Components | | G/E/Z | |
| | | | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Safe Primes Key Generation, Safe Primes Key Verification | SSH DHE/ECDHE Public Components | | G/E/R/W/Z | |

|  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|
|  |  | KTS | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512 | SSH Session Authentication Keys |  | G/E/Z |  |
|  |  |  | AES-CBC,<br>AES-CTR | SSH Session Encryption Keys |  | G/E/Z |  |
|  |  | KTS | AES-GCM |  |  |  |  |
|  |  | N/A |  | CO, User Password |  | G/E/W |  |
|  |  | Counter DRBG, ESV |  | DRBG Seed |  | G/E |  |
|  |  |  |  | DRBG V |  |  |  |
|  |  |  |  | DRBG Key |  |  |  |
|  |  |  |  | Entropy Input String |  |  |  |
| Check Status | Review system, configuration, debug logs, and show configurations. | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) |  | RSA Private Keys | CO | G/W/E | System Logs |
|  |  | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) |  | ECDSA Private Keys |  | G/W/E |  |
|  |  | KAS | KDF SSH (CVL) | SSH DHE/ECDHE Private Components |  | G/E/Z |  |
|  |  |  | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Safe Primes Key Generation, Safe Primes Key Verification | SSH DHE/ECDHE Public Components |  | G/E/R/W/Z |  |
|  |  | KTS | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512 | SSH Session Authentication Keys |  | G/E/Z |  |
|  |  |  | AES-CBC,<br>AES-CTR | SSH Session Encryption Keys |  | G/E/Z |  |
|  |  | KTS | AES-GCM |  |  |  |  |
|  |  | N/A |  | CO, User Password |  | G/E/W |  |
|  |  | Counter DRBG, ESV |  | DRBG Seed |  | G/E |  |
|  |  |  |  | DRBG V |  |  |  |
|  |  |  |  | DRBG Key |  |  |  |
|  |  |  |  | Entropy Input String |  |  |  |
|  |  | KDF SNMP (CVL) |  | SNMPv3 Authentication Secret |  | W/E |  |
|  |  | KDF SNMP (CVL) |  | SNMPv3 Privacy Secret |  | W/E |  |
|  |  | HMAC-SHA-1<br>HMAC-SHA2-224<br>HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 |  | Authentication Key |  | G/E/Z |  |
|  |  | AES-CFB128 |  | Session Key |  | G/E/Z |  |
| System Audit | Allows review of limited configuration and system status via logs, dashboard and configuration screens. Provides | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) |  | RSA Private Keys | CO | G/W/E | System Logs |
|  |  | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) |  | ECDSA Private Keys |  | G/W/E |  |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | no configuration commit capability. | KAS | KDF SSH (CVL) | SSH DHE/ECDHE Private Components | | G/E/Z | |
| | | | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Safe Primes Key Generation, Safe Primes Key Verification | SSH DHE/ECDHE Public Components | | G/E/R/W/Z | |
| | | KTS | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512 | SSH Session Authentication Keys | | G/E/Z | |
| | | | AES-CBC,<br>AES-CTR | SSH Session Encryption Keys | | G/E/Z | |
| | | KTS | AES-GCM | | | | |
| | | N/A | | CO, User Password | | G/E/W | |
| | | Counter DRBG, ESV | | Entropy Input String | | G/E | |
| | | | | DRBG Seed | | | |
| | | | | DRBG V | | | |
| | | | | DRBG Key | | | |
| IKE/IPsec Configuration | Configures IKE/IPsec setup for peer to peer VPN. | CKG<br>RSA KeyGen (FIPS 186-4)<br>RSA SigGen (FIPS 186-4) | | RSA Private Keys | Peer-to-Peer VPN | G/W/E | System Logs |
| | | CKG<br>ECDSA KeyGen (FIPS 186-4)<br>ECDSA SigGen (FIPS 186-4) | | ECDSA Private Keys | | G/W/E | |
| | | Counter DRBG | | Entropy Input String | | G/E | |
| | | | | DRBG Seed | | | |
| | | | | DRBG V | | | |
| | | | | DRBG Key | | | |
| | | KAS | KDF IKEv2 (CVL) | IPSec/IKE DHE/ECDHE Public Components | | G/E/Z | |
| | | | CKG,<br>ECDSA KeyGen (FIPS 186-4),<br>ECDSA KeyVer (FIPS 186-4),<br>KAS-ECC-SSC,<br>KAS-FFC-SSC,<br>Safe Primes Key Generation, Safe Primes Key Verification | IPSec/IKE DHE/ECDHE Private Components | | G/E/Z | |
| | | KTS | HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 | IPSec/IKE Authentication Keys | | G/E/Z | |
| | | | AES-CBC | IPSec/IKE Session Keys | | | |
| | | KTS | AES-GCM | IPSec/IKE Session Keys | | G/E/Z | |
| | | RSA SigVer (FIPS 186-4) | | RSA Public Keys<br>CA Certificates | | G/R/E/W | |
| | | ECDSA SigVer (FIPS 186-4) | | ECDSA Public Keys<br>CA Certificates | | G/R/E/W | |
| Zeroize | Destroys all keys in the module | N/A | | All Keys and SSPs | CO | Z | Console Output / Zeroization indicator |

| | | | | | | |
|---|---|---|---|---|---|---|
| Self-Tests | Run power up self-tests on demand by power cycling the module. | HMAC-SHA2-256,<br>ECDSA SigVer (FIPS 186-4) | Firmware Integrity Check Key | CO | E | System Logs |
| Show Status | View hardware status of the module via the LEDs. | N/A | N/A | All | N/A | LEDs |

## 5. Software/Firmware Security

The module performs the Firmware Integrity test by using HMAC-SHA-256 and ECDSA signature verification (HMAC and ECDSA Cert. #A2137) during the Pre-Operational Self-Test.  In addition, the module also conducts the firmware load test by using the Public Verification Key (RSA 2048 with SHA-256, Cert. #A2137) for the new validated firmware to be uploaded into the module via the System Operational Management service. The Firmware Integrity Verification key and Public key for Firmware Content Load Test used for the Firmware Integrity and Firmware Load test, respectively, are generated externally and delivered as part of the module firmware image.

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 firmware integrity test.

The module's executable code is in the form of the compiled firmware image loaded onto the module.

## 6. Operational Environment

The FIPS 140-3 Operational Environment requirements are not applicable. The operational environment is limited since the Module includes a firmware load service to support necessary updates. New firmware versions within the scope of this validation must be validated through the FIPS 140-3 CMVP. Any other firmware loaded into this module is out of the scope of this validation and requires a separate FIPS 140-3 validation.

# 7. Physical Security

### Physical Security Mechanisms

The multi-chip standalone module is production quality and contains standard passivation. Chip components are protected by an opaque enclosure. There are tamper-evident seals that are applied on the module by the Crypto-Officer, and any unused seals are to be controlled by the Crypto-Officer. The Crypto-Officer must ensure that the module surface is clean and dry before applying the seals. The seals prevent removal of the opaque enclosure without evidence, which should be inspected by the Crypto-Officer every 30 days for evidence of tampering. If the seals or opacity shields show evidence of tamper, the Crypto-Officer should assume that the module has been compromised and contact Customer Support.

**Note:** For ordering information, see Table 1 for physical kit part numbers and version. Opacity shields are included in the physical kits.

Refer to Appendix A for instructions regarding installation of the tamper seals and opacity shields. Tamper-evident seals must be pressed firmly onto the adhering surfaces during installation, and once applied, the Crypto-Officer shall permit 24 hours of cure time for all tamper-evident seals. The placement of the twelve (12) tamper-evident seals are shown in Appendix A.

### Operator Required Actions

The following table provides information regarding the various physical security mechanisms, and their recommended frequency of inspection/test.

*Table 10 - Physical Security Inspection Guidelines*

| Physical Security Mechanism | Recommended Frequency of Inspection/Test | Inspection/Test Guidance Details |
|---|---|---|
| Tamper-Evident Seals | 30 days | Verify integrity of tamper-evident seals in the locations specified in Appendix A. |
| Front and Rear Opacity Shields | 30 days | Verify that the front and rear opacity shields have not been deformed from their original shape, thereby reducing their effectiveness. |
| Vent Overlays | 30 days | Verify that the vent overlays have not been removed or deformed. All edges should maintain strong adhesion characteristics. |

## 8. Non-Invasive Security

There are currently no defined Approved non-invasive attack mitigation test metrics in SP 800-140F.

## 9. Sensitive Security Parameter Management

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

“TLS or SSH Session Key Encrypted” corresponds to the following KTS entries listed in the Approved Algorithms table:

- AES Cert. #A2137, HMAC Cert. #A2137
- AES-GCM Cert. #A2137

“IPSec/IKE, KAS SP 800-56A Rev. 3” corresponds to the following KAS entries listed in the Approved Algorithms table:

- KAS-ECC-SSC Cert. #A2137, KDF IKEv2 Cert. #A2137
- KAS-FFC-SSC Cert. #A2137, KDF IKEv2 Cert. #A2137

“SSH, KAS SP 800-56A Rev. 3” corresponds to the following KAS entries listed in the Approved Algorithms table:

- KAS-ECC-SSC Cert. #A2137, KDF SSH Cert. #A2137
- KAS-FFC-SSC Cert. #A2137, KDF SSH Cert. #A2137

“TLS, KAS SP 800-56A Rev. 3” corresponds to the following KAS entries listed in the Approved Algorithms table:

- KAS-ECC-SSC Cert. #A2137, KDF TLS Cert. #A2137
- KAS-FFC-SSC Cert. #A2137, KDF TLS Cert. #A2137

*Table 11 – SSPs*

| Key/SSP/Name/Type | Strength | Security Function and Cert. Number | Generation | Import/Export | Establishment | Storage | Zeroization | Use & Related Keys |
|---|---|---|---|---|---|---|---|---|
| CA Certificates | 112 bits minimum | RSA SigVer (FIPS 186-4)<br>ECDSA SigVer (FIPS 186-4)<br>Cert. #A2137 | DRBG, FIPS 186-4 | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | HDD – Zeroize Service<br>RAM - Zeroize at session termination | ECDSA/RSA Public key - Used to trust a root CA intermediate CA and leaf /end entity certificates<br>(RSA 2048, 3072, and 4096 bits)<br>(ECDSA P-256, P-384, and P-521) |
| RSA Public Keys | 112 bits minimum | RSA SigVer (FIPS 186-4) Cert. #A2137 | DRBG, FIPS 186-4 | TLS or SSH Session Key Encrypted or Plaintext<br>TLS handshake | N/A | HDD/RAM – plaintext | Zeroize Service | RSA public keys managed as certificates for the verification of signatures, establishment of TLS, operator authentication and peer authentication.<br>(RSA 2048, 3072, or 4096-bit) |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| RSA Private Keys | 112 bits minimum | RSA SigGen (FIPS 186-4)<br>Cert. #A2137 | DRBG, FIPS 186-4 | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | HDD – Zeroize Service<br>RAM - Zeroize at session termination | RSA Private keys for generation of signatures, authentication or key establishment.<br>(RSA 2048, 3072, or 4096-bit) |
| ECDSA Public Keys | 128 bits minimum | ECDSA SigVer (FIPS 186-4)<br>Cert. #A2137 | DRBG, FIPS 186-4 | TLS or SSH Session Key Encrypted or Plaintext<br>TLS handshake | N/A | HDD/RAM – plaintext | Zeroize Service | ECDSA public keys managed as certificates for the verification of signatures, establishment of TLS, operator authentication and peer authentication.<br>(ECDSA P-256, P-384, or P-521) |
| ECDSA Private Keys | 128 bits minimum | ECDSA SigGen (FIPS 186-4)<br>Cert. #A2137 | DRBG, FIPS 186-4 | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | HDD – Zeroize Service<br>RAM - Zeroize at session termination | ECDSA Private key for generation of signatures and authentication<br>(P-256, P-384, or P-521) |
| TLS DHE/ECDHE Private Components | 112 bits minimum | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Cert. #A2137 | DRBG, SP 800-56A Rev. 3 | N/A | N/A | RAM - plaintext | Zeroize at session termination | Ephemeral Diffie-Hellman private FFC or EC component used in TLS<br>(DHE 2048, ECDHE P-256, P-384, P-521) |
| TLS DHE/ECDHE Public Components | 112 bits minimum | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Cert. #A2137 | DRBG, SP 800-56A Rev. 3 | Plaintext - TLS handshake | N/A | N/A | Zeroize at session termination | Diffie_Hellman or EC Diffie-Hellman Ephemeral values used in key agreement<br>(DHE 2048, ECDHE P-256, P-384, P-521) |
| TLS Pre-Master Secret | N/A | KDF TLS Cert. #A2137, MD5 (No Security Claimed | KAS-ECC-SSC or KAS-FFC-SSC, SP 800-56A Rev. 3 | N/A | TLS, KAS SP 800-56A Rev. 3 | RAM – plaintext | Zeroize at session termination | Secret value used to derive the TLS Master Secret along with client and server random nonces |
| TLS Master Secret | N/A | KDF TLS Cert. #A2137, MD5 (No Security Claimed | KDF TLS | N/A | TLS, KAS SP 800-56A Rev. 3 | RAM – plaintext | Zeroize at session termination | Secret value used to derive the TLS session keys |
| TLS Encryption Keys | 128 bits minimum | AES-CBC or AES-GCM<br>Cert. #A2137 | KDF TLS | N/A | TLS, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | AES (128 or 256 bit) keys used in TLS connections (GCM; CBC) |
| TLS HMAC Keys | 160 bits minimum | HMAC-SHA2-256<br>HMAC-SHA2-384<br>Cert. #A2137 | TLS KDF (CVL) | N/A | TLS, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | HMAC keys used in TLS connections (SHA-1, 256, 384)<br>(160, 256, 384 bits) |
| SSH DHE/ECDHE Private Components | 112 bits minimum | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Cert. #A2137 | DRBG, SP 800-56A Rev. 3 | N/A | N/A | RAM - plaintext | Zeroize at session termination | Diffie Hellman or EC Diffie-Hellman private (DH Group 14, ECDH P-256, ECDH P-384, ECDH P-521) |
| SSH DHE/ECDHE Public Components | 112 bits minimum | KAS-ECC-SSC<br>KAS-FFC-SSC<br>Cert. #A2137 | DRBG, SP 800-56A Rev. 3 | Plaintext SSH handshake | N/A | RAM - plaintext | Zeroize at session termination | Diffie Hellman or EC Diffie-Hellman public component (DH Group |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 14, ECDH P-256, ECDH P-384, ECDH P-521) |
| SSH Host Public Key | 112 bits minimum | RSA SigVer (FIPS 186-4) ECDSA SigVer (FIPS 186-4)<br>Cert. #A2137 | DRBG, FIPS 186-4 | N/A | N/A | HDD/RAM – plaintext | Zeroize Service | SSH Host Public Key (RSA 2048, RSA 3072, RSA 4096, ECDSA P-256, P-384, or P-521) |
| SSH Client Public Key | 112 bits minimum | RSA SigVer (FIPS 186-4) Cert. #A2137 | N/A | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | Zeroize Service | Public RSA key used to authenticate client. (RSA 2048, 3072, and 4096 bits) |
| SSH Session Encryption Keys | 128 bits minimum | AES-CBC, AES-CTR, or AES-GCM Cert. #A2137 | N/A | N/A | SSH, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | Used in all SSH connections to the security module's command line interface.<br>(128, 192, or 256 bits: CBC or CTR)<br>(128 or 256 bits: GCM) |
| SSH Session Authentication Keys | 160 bits minimum | HMAC-SHA-1<br>HMAC-SHA2-256<br>HMAC-SHA2-512<br>Cert. #A2137 | N/A | N/A | SSH, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | Authentication keys used in all SSH connections to the security module's command line interface (HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-512) (160, 256, 512 bits) |
| IPSec/IKE DHE/ECDHE Private Components | 112 bits minimum | KAS-ECC-SSC KAS-FFC-SSC Cert. #A2137 | DBRG, SP 800-56A Rev. 3 | N/A | N/A | RAM - plaintext | Power cycle | Diffie-Hellman or EC Diffie-Hellman private component used in key establishment (DHE 2048, ECDHE P-256, P-384) |
| IPSec/IKE DHE/ECDHE Public Components | 112 bits minimum | KAS-ECC-SSC KAS-FFC-SSC Cert. #A2137 | DRBG, SP 800-56A Rev. 3 | N/A | N/A | RAM - plaintext | Power cycle | Diffie-Hellman or EC Diffie-Hellman public component used in key agreement (DHE 2048, ECDHE P-256, P-384) |
| IPSec/IKE Session Keys | 128 bits minimum | AES-CBC, AES-GCM Cert. #A2137 | N/A | N/A | IPSec/IKE, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | Used to encrypt IKE/IPSec data. These are AES CBC or GCM (128 or 256 bits). |
| IPSec/IKE Authentication Keys | 256 bits minimum | HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512<br>Cert. #A2137 | N/A | N/A | IPSec/IKE, KAS SP 800-56A Rev. 3 | RAM - plaintext | Zeroize at session termination | (HMAC-SHA-256, SHA-384 or SHA-512) Used to authenticate the peer in an IKE/IPSec tunnel connection. (256, 384, 512 bits) |
| CO, User Password | N/A | N/A | External | TLS or SSH Session Key Encrypted | N/A | HDD - a password hash | Zeroize Service | Authentication string with a minimum length of eight (8) characters. |
| Protocol Secrets | N/A | N/A | External | TLS or SSH Session Key Encrypted | N/A | HDD– Plaintext<br>RAM – Plaintext | Zeroize Service | Secrets used by RADIUS or TACACS+ (8 characters minimum) |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Entropy Input String | 256 bits | CKG (vendor affirmed), Counter DRBG Cert. #A2137 | Entropy as per SP 800-90B | N/A | N/A | RAM - plaintext | Power cycle | DRBG input string coming from the entropy source<br>Input length = 384 bits |
| DRBG Seed | 256 bits | CKG (vendor affirmed), Counter DRBG Cert. #A2137 | Entropy as per SP 800-90B | N/A | N/A | RAM - Plaintext | Power cycle | DRBG seed coming from the entropy source<br>Seed length = 384 bits |
| DRBG V | 128 bits | CKG (vendor affirmed), Counter DRBG Cert. #A2137 | Entropy as per SP 800-90B | N/A | N/A | RAM - plaintext | Power cycle | AES 256 CTR DRBG State (V) used in the generation of random values |
| DRBG Key | 256 bits | CKG (vendor affirmed), Counter DRBG Cert. #A2137 | Entropy as per SP 800-90B | N/A | N/A | RAM - plaintext | Power cycle | AES 256 CTR DRBG State (Key) used in the generation of random values |
| SNMPv3 Authentication Secret | N/A | KDF SNMP (CVL) Cert. #A2137 | N/A | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | Zeroize Service | Used to support SNMPv3 services (Minimum 8 characters) |
| SNMPv3 Privacy Secret | N/A | KDF SNMP (CVL) Cert. #A2137 | N/A | TLS or SSH Session Key Encrypted | N/A | HDD/RAM – plaintext | Zeroize Service | Used to support SNMPv3 services (Minimum 8 characters) |
| Authentication Key | 160 bits minimum | HMAC-SHA-1<br>HMAC-SHA2-224<br>HMAC-SHA2-256<br>HMAC-SHA2-384<br>HMAC-SHA2-512 Cert. #A2137 | SNMPv3 KDF (CVL) | N/A | N/A | HDD/RAM - Plaintext | Zeroize Service | HMAC–SHA-1/224/256/384/512 Authentication protocol key (160 bits) |
| Session Key | 128 bits minimum | AES-CFB128 Cert. #A2137 | SNMPv3 KDF (CVL) | N/A | N/A | HDD/RAM - Plaintext | Zeroize Service | Privacy protocol encryption key (AES 128/192/256 CFB128) |
| Public key for firmware content load test | 112 bits2048 bits | RSA SIgVer Cert. #A2137 | Factory preload | Import only, TLS or SSH Session Key Encrypted | N/A | HDD – Plaintext as part of firmware image | N/A | Used to authenticate firmware and content to be installed on the appliance (RSA 2048 with SHA-256) |
| Firmware Integrity Check Key | 256 bits | ECDSA SigVer Cert. #A2137, HMAC-SHA2-256 Cert. #A2137 | Factory preload | Import only, TLS or SSH Session Key Encrypted | N/A | HDD – Plaintext as part of firmware image | N/A | Used to check the integrity of all software code (HMAC-SHA-256* and ECDSA P-256)<br>*Keys used to perform power-up self-tests are not CSPs |

Note: SSPs are implicitly zeroized when power cycling and explicitly zeroized when using the zeroize service.

*Table 12 - Non-Deterministic Random Number Generation Specification*

| Entropy Source | Minimum number of bits of entropy | Details |
|---|---|---|
| Palo Alto Networks RTC Entropy Source | 256 bits | ESV Cert. #E130<br>When initialized per Section 11, the DRBG is seeded with 256 bits of entropy |

# 10. Self-Tests

The module design corresponds to the module security rules. This section documents the security rules enforced by the cryptographic module to implement the security requirements of a FIPS 140-3 Level 2 module.

1. The cryptographic module shall provide distinct operator roles. When the module has not been placed in a valid role, the operator shall not have access to any cryptographic services.
2. The cryptographic module shall clear previous authentications on power cycle.
3. The cryptographic module performs the following tests
    A. Pre-operational Self-Tests
        1. Firmware Integrity Test –verified with HMAC-SHA-256 and ECDSA P-256*
           *Note: the ECDSA and HMAC-SHA-256 KATs are performed prior to the Firmware integrity test
    B. Conditional self-tests
        1. Cryptographic algorithm self-tests
            a. AES 128-bit ECB Encrypt Known Answer Test*
            b. AES 128-bit ECB Decrypt Known Answer Test*
            c. AES 128-bit CMAC Known Answer Test*
            d. AES 256-bit GCM Encrypt Known Answer Test
            e. AES 256-bit GCM Decrypt Known Answer Test
            f. AES 192-bit CCM Encrypt Known Answer Test*
            g. AES 192-bit CCM Decrypt Known Answer Test*
            h. RSA 2048-bit PKCS#1 v1.5 with SHA-256 Sign Known Answer Test
            i. RSA 2048-bit PKCS#1 v1.5 with SHA-256 Verify Known Answer Test
            j. RSA 2048-bit Encrypt Known Answer Test*
            k. RSA 2048-bit Decrypt Known Answer Test*
            l. ECDSA P-256 with SHA-512 Sign Known Answer Test
            m. ECDSA P-256 with SHA-512 Verify Known Answer Test
            n. HMAC-SHA-1 Known Answer Test
            o. HMAC-SHA-256 Known Answer Test
            p. HMAC-SHA-384 Known Answer Test
            q. HMAC-SHA-512 Known Answer Test
            r. SHA-1 Known Answer Test
            s. SHA-256 Known Answer Test
            t. SHA-384 Known Answer Test
            u. SHA-512 Known Answer Test
            v. DRBG Instantiate/Generate/Reseed SP800-90A Known Answer Tests
            w. SP 800-90A Instantiate/Generate/Reseed Section 11.3 Health Tests
            x. KAS-FFC-SSC 2048-bit Known Answer Test
            y. KAS-ECC-SSC P-256 Known Answer Test
            z. SP 800-135 TLS 1.0/1.1 KDF KAT
            aa. SP 800-135 TLS 1.2 with SHA-256 KDF KAT
            bb. SP 800-135 SSH KDF KAT
            cc. SP 800-135 IKE KDF KAT
            dd. Continuous Random Number Generator (RNG) test – performed on DRBG
            ee. SP 800-90B RCT/APT Health Tests on Entropy Source

           *Note: Supported by the module cryptographic implementation, but only utilized for CAST
        2. Pairwise Consistency Self-Tests
            a. RSA Pairwise Consistency Test
            b. ECDSA/KAS-ECC Pairwise Consistency Test
            c. KAS-FFC Pairwise Consistency Test
        3. Software/firmware Load test
            a. Firmware Load Test – Verify RSA 2048 with SHA-256 signature on firmware at time of load

4. Critical Functions Tests
    a. SP 800-56A Rev. 3 Assurance Tests (Based on Sections 5.5.2, 5.6.2, and 5.6.3)

If any self-tests or conditional tests fail, the module will output 'FIPS-CC failure' and the specific test that failed.

1. Power-up self-tests shall not require any operator action.
2. The operator shall be capable of commanding the module to perform the power-up self-test by power cycling the module.
3. Data output shall be inhibited during power-up self-tests and error states.
4. Processes performing key generation and zeroization processes shall be logically isolated from the logical data output paths.
5. The module does not output intermediate key generation values.
6. Status information output from the module shall not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
7. There are no restrictions on which keys or CSPs are zeroized by the zeroization service.
8. The module maintains separation between concurrent operators.
9. The module does not support a maintenance interface or role.
10. The module does not have any external input/output devices used for entry/output of data.
11. The module does not allow the input or output of plaintext CSPs.
12. The module provides a warning, "Your device is still configured with the default admin account credentials. Please change your password prior to deployment." to inform the operator to change their default authentication data.

# 11. Life-cycle Assurance

The vendor provided life-cycle assurance documentation that describes configuration management, design, finite state model, development, testing, delivery + operation, end of life procedures, and guidance. For details regarding the secure installation, initialization, startup, and operation of the module, see section "Approved Mode of Operation" in Section 2.

Palo Alto Network provides an Administrator Guide for additional information noted in the "References" section of this Security Policy

## Vendor imposed security rules

In FIPS-CC mode, the following rules shall apply:

1. If the cryptographic module remains inactive in any valid role for the administrator specified time interval, the module automatically logs out the operator.
2. Once boot-up is complete, the WF-500 requires a minimum system uptime of 1 hour before the module can be used to ensure proper instantiation of the DRBG.
    A. Verify uptime via the following command: "show system info | match uptime"
    B. After this time, regenerate any items previously present such as the SSH keys using the following procedure:
        1. Login via CLI and issue the following command:
            a. debug system ssh-key-reset all
3. The module enforces a timed access protection mechanism that supports at most ten authentication attempts per minute. After the administrator-specified number of consecutive unsuccessful password

validation attempts have occurred, the cryptographic module shall enforce a wait period of at least one (1) minute before any more login attempts can be attempted.

4. In FIPS-CC mode, the following rules shall apply:
   A. The operator should not enable TLSv1.0 or use RSA for key wrapping; it is disabled by default.
      - Checked via CLI using "show shared" command

   B. If using RADIUS, it must be configured using TLS.
      - Checked via CLI using "show shared" command

Failure to follow these Security Rules will cause the module to operate in a non-compliant state.

# 12. Mitigation of Other Attacks

The module is not designed to mitigate any specific attacks outside the scope of FIPS 140-3. These requirements are not applicable.

# 13. References

[FIPS 140-3] FIPS Publication 140-3 Security Requirements for Cryptographic Modules

[AGD] WildFire Administrator's Guide
https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/wildfire/10-1/wildfire-admin/wildfire-admin.pdf

# 14. Definitions and Acronyms

AES – Advanced Encryption Standard
CA – Certificate Authority
CLI – Command Line Interface
CO – Crypto-Officer
CSP – Critical Security Parameter
CVL – Component Validation List
DB9 – D-sub series, E size, 9 pins
DES – Data Encryption Standard
DH – Diffie-Hellman
DRBG – Deterministic Random Bit Generator
EDC – Error Detection Code
ECDH – Elliptical Curve Diffie-Hellman
ECDSA – Elliptical Curve Digital Signature Algorithm
FIPS – Federal Information Processing Standard
HMAC – (Keyed) Hashed Message Authentication Code
KDF – Key Derivation Function
LED – Light Emitting Diode
RJ45 – Networking Connector
RNG –Random number generator
RSA – Algorithm developed by Rivest, Shamir and Adleman
SHA – Secure Hash Algorithm
SNMP – Simple Network Management Protocol
SSH – Secure Shell
TLS – Transport Layer Security

USB – Universal Serial Bus
VGA – Video Graphics Array
WF – WildFire

# Appendix A – WF-500 Physical Kit Installation Guide (12 Tamper-Evident Seals)

This section provides steps on how to install the Physical Kit on the WF-500 module.

Step 1:
Remove the two pull handles and front modules on the left and right side of the appliance by removing the three (3) screws located behind each handle/module. There is no need to disconnect the LED circuit board attached to the end of the ribbon cable. Retain these screws for Step 2.

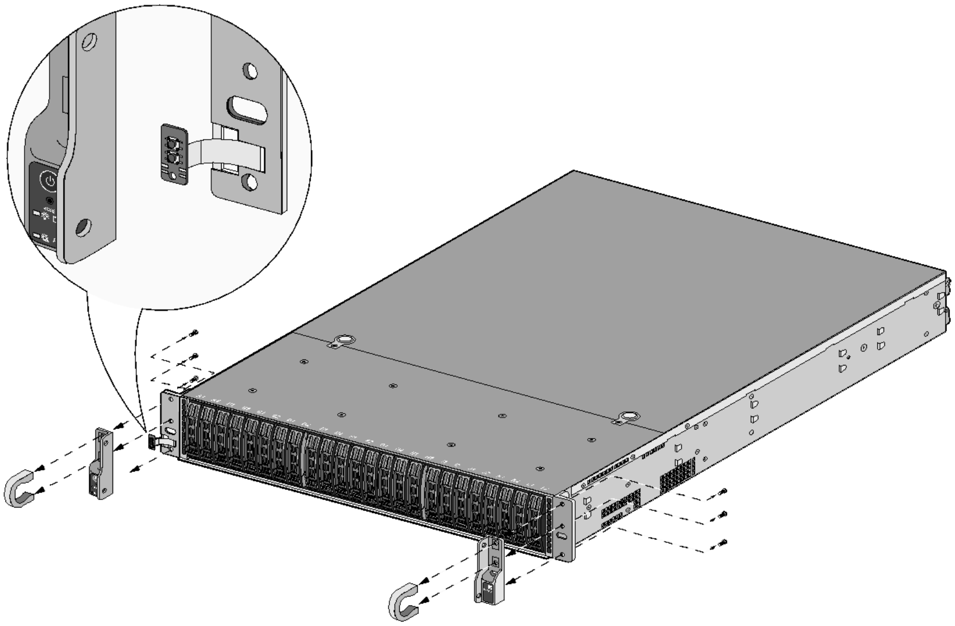

*Figure 8 – Remove Front Handles and Modules*

Step 2:
Attach the left and right front cover brackets to the appliance using the six (6) screws that were removed in Step 1. First attach the brackets using the bottom screws (one (1) on each side) as shown in Figure 9, ensuring that you feed the ribbon cable and LED circuit board through the left bracket. Replace the front modules and secure them using the middle and top screws on each side as shown in Figure 10.

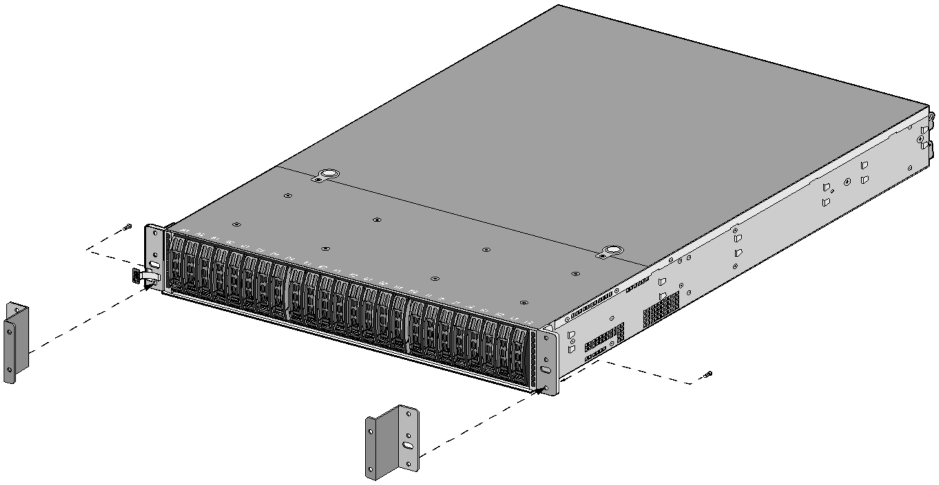

*Figure 9 – Secure the Front Brackets*

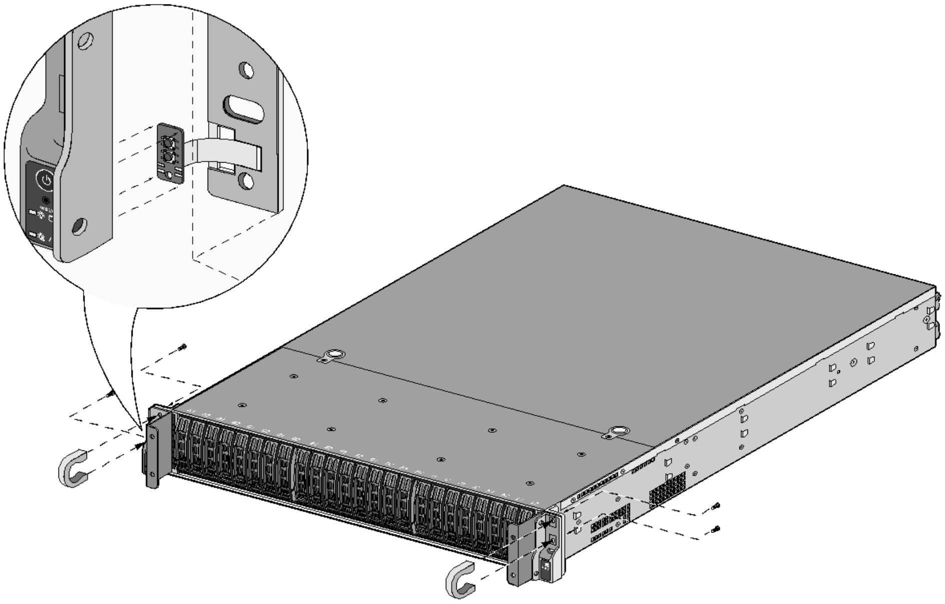

*Figure 10 - Attach Pull Handles and Front Modules*

Step 3:
Secure the front opacity shield to the right and left front brackets that you installed in Step 2. Use two (2) screws (provided) on each side.

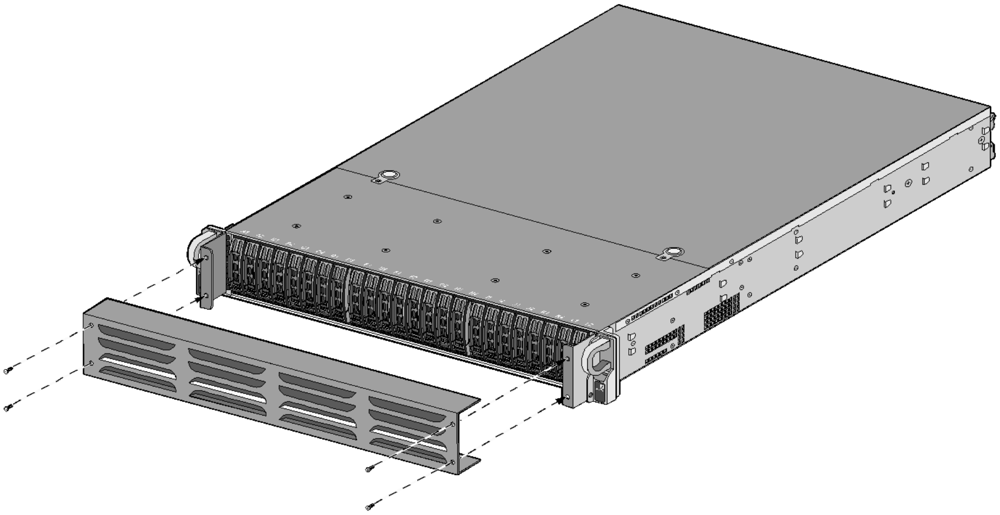

*Figure 11 – Install Front Opacity Shield*

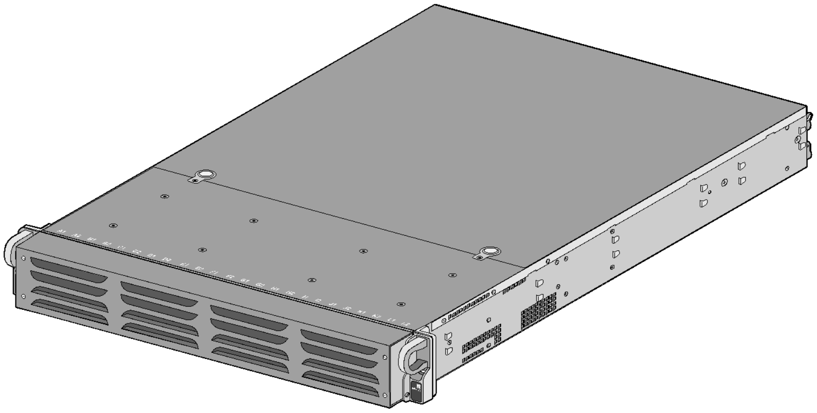

*Figure 12 – Front Opacity Shield Installed*

Step 4:
Attach the rear opacity shield tray to the appliance. First, remove the two (2) screws (shown in Figure 13) from the appliance and use these screws to secure the rear opacity shield tray.
**Note:** Install the back cables (power cords and network/management cables) because you will not be able to access these ports after the next step.

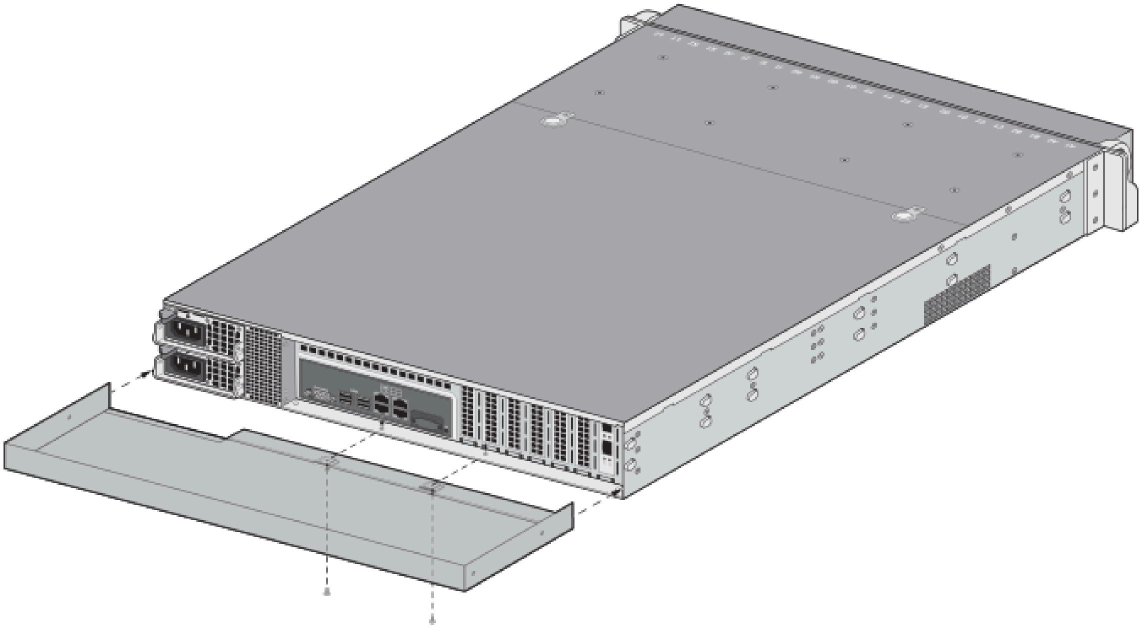

*Figure 13 – Install Rear Opacity Shield Tray*

Step 5:
Place the rear opacity shield on top of the rear opacity shield tray ensuring that you run the cables through the opening at the bottom.  Secure the opacity shields with two (2) screws (provided) on each side.

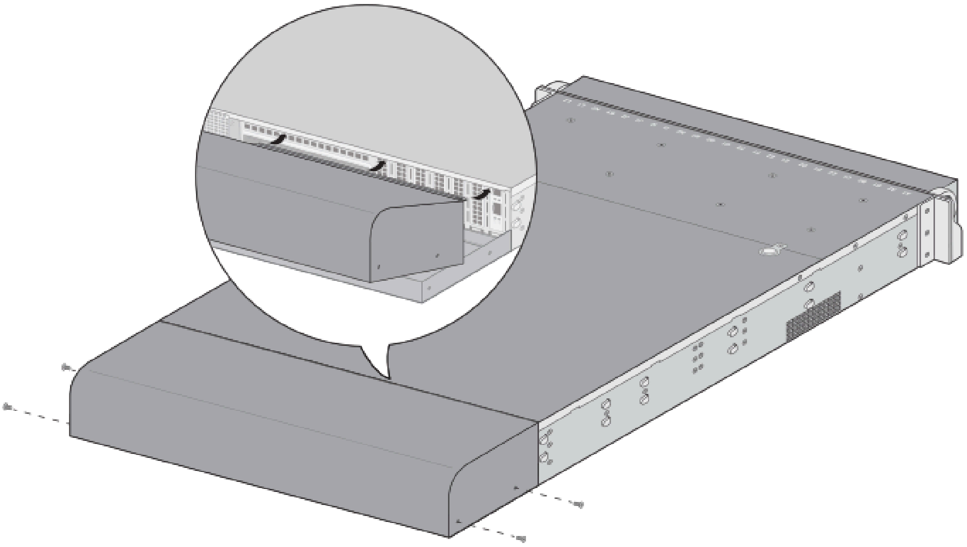

*Figure 14 – Install Rear Opacity Shield*

Step 6:
Cover the vent openings as shown in Figure 15 by applying one (1) overlay tamper-evident seal over the left side vent and one overlay tamper-evident seal over the right side vent.  Each overlay requires two (2) tamper-evident seals as shown in Figure 16.  Also apply one (1) additional tamper-evident seal as shown in Figure 16, #5.

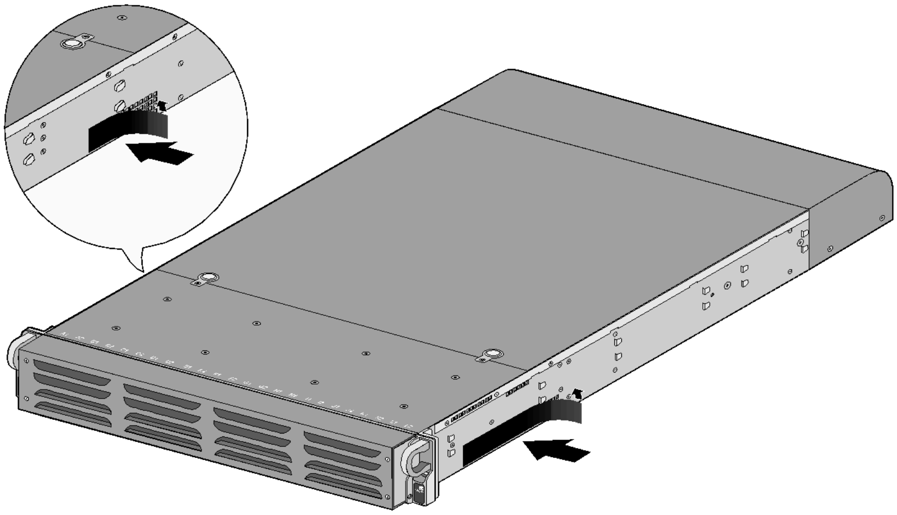

*Figure 15 – Apply Tamper-Evident Seals on Vent Overlays*

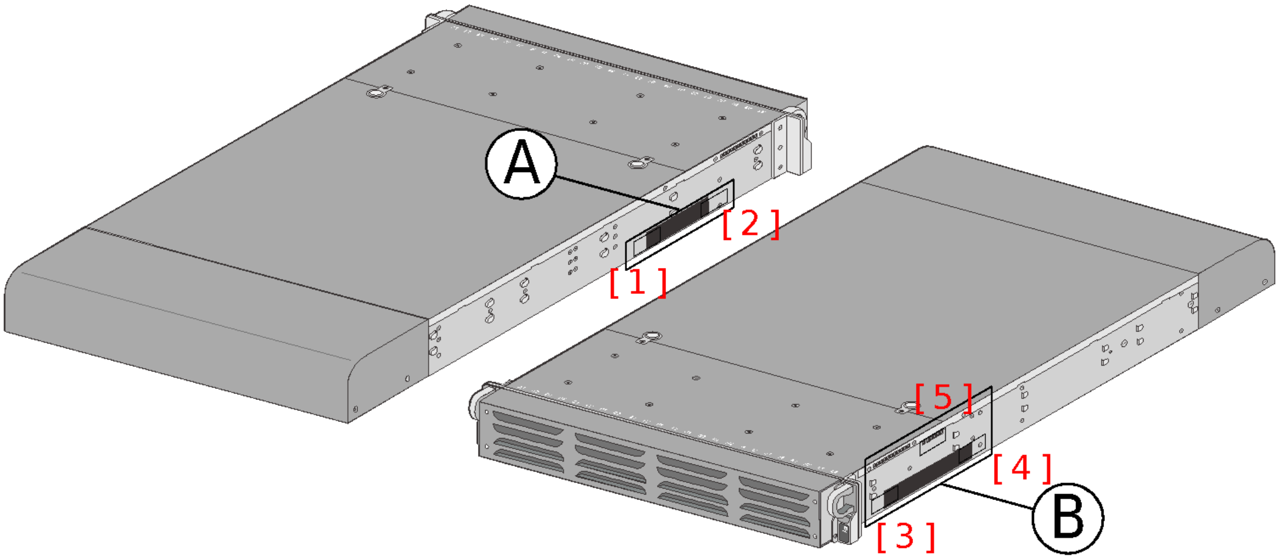

*Figure 16 – Apply Tamper-Evident Seals on Vent Overlays and Side Opening*

Step 7:
Attach the rail kit to the appliance as shown in Figure 17 and then add three (3) tamper-evident seals to the bottom of the appliance as shown in Figure 18.  One (1) tamper-evident seal prevents tampering of the front opacity shield connected to the bottom of the appliance and two (2) tamper-evident seals wrap around the upper and lower rear opacity shields to prevent tampering of the rear opacity shields.

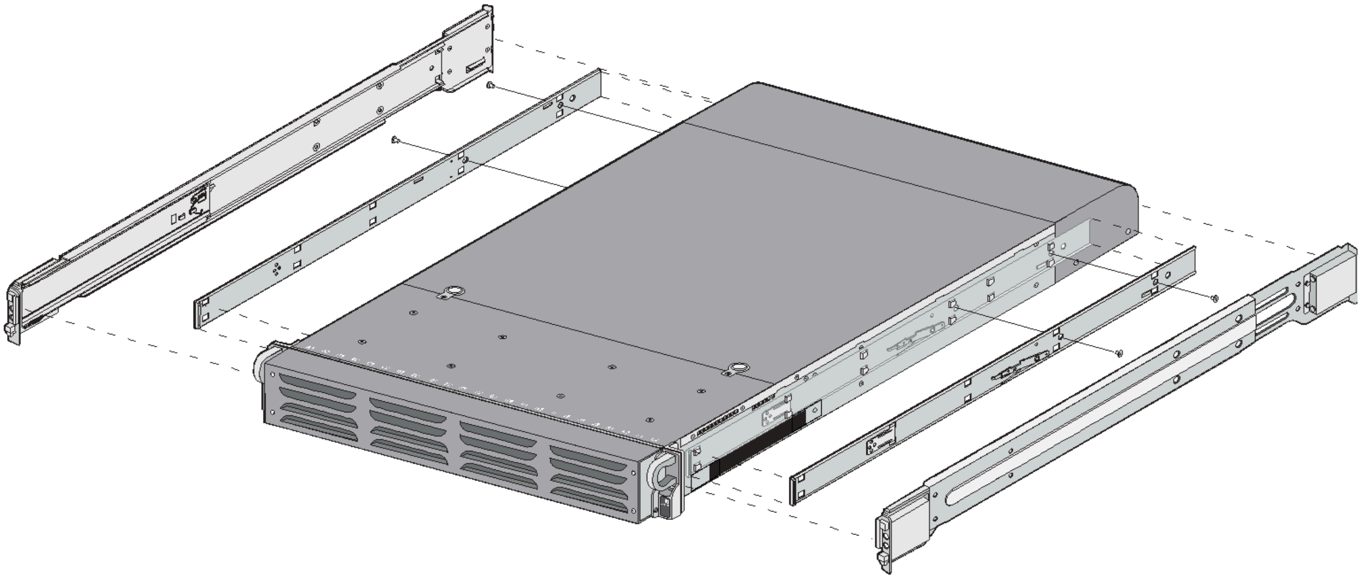

*Figure 17 – Install Rail Kit*

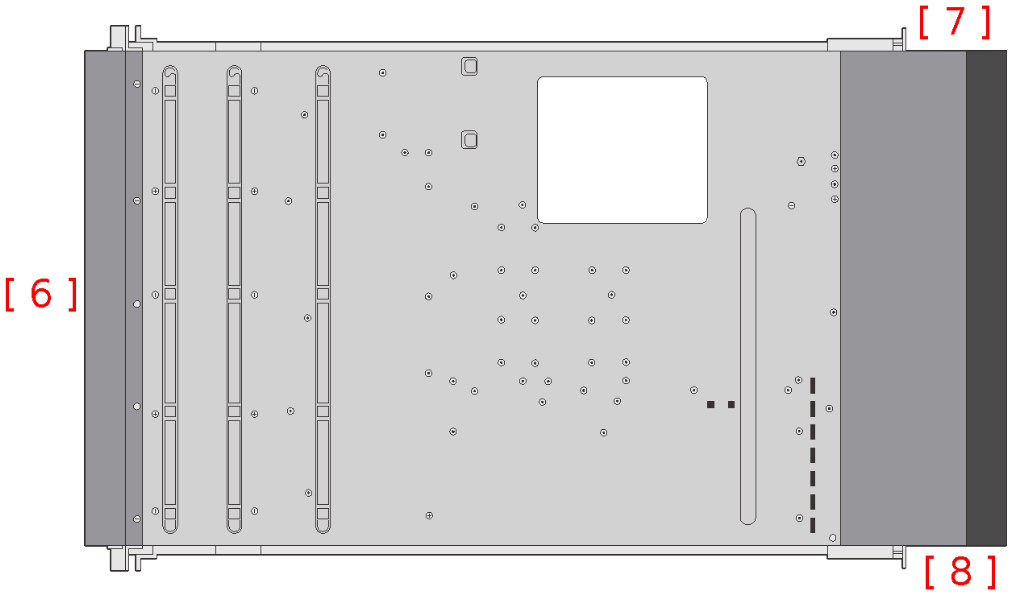

*Figure 18 – Apply Tamper-Evident Seals on the Bottom of the Appliance*

Step 8:
Place four (4) tamper seals on the top of the appliance. Two (2) tamper seals (#9 and #11) prevent tampering of the top front and rear opacity shields and two (2) tamper seals (#10 and #12) prevents someone from attempting to access the vent overlays by sliding the rail kit. This completes the physical kit installation.

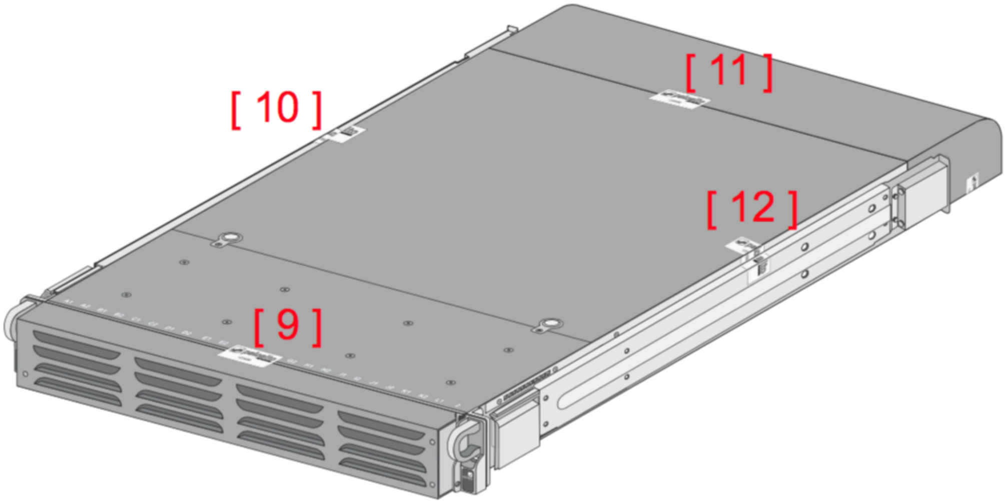

*Figure 19 – Apply Tamper-Evident Seals on the Top  of the Appliance*