National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme Validation Report

General Dynamics Mission Systems

FORTRESS Mesh Point ES210, ES520, ES820, ES2440

Report Number: CCEVS-VR-VID10667
Dated: March 11, 2016
Version: 1.0

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6940
Fort George G. Meade, MD 20755-6940

Acknowledgements

Validation Panel

Mr. Daniel Faigin

The Aerospace Corporation

El Segundo, CA

Mr. Luke Florer

The Aerospace Corporation

Chantilly, VA

Common Criteria Testing Laboratory

Scott Cutler, Brad Mitchell, Michael Baron

InfoGard Laboratories, Inc.

San Luis Obispo, CA

Table of Contents

1 Executive Summary

This report documents the NIAP validators' assessment of the CCEVS evaluation of the Fortress Mesh Point ES210, ES520, ES820 and ES2440, collectively referred to as the Target of Evaluation (TOE).

This report is intended to assist the end-user of this product with determining the suitability of this IT product in their environment. End-users should review both the Security Target (ST), which is where specific security claims are made, in conjunction with this Validation Report (VR), which describes how those security claims were evaluated.

The TOE is classified as a VPN Gateway Network Device. The TOE employs Mesh networking, which allows multiple TOEs to network within the operational environment. Only VPN gateway functionality is evaluated in *this* Security Target; WLAN functionality was evaluated in VIDs 10571, 10572, and 10573.

2 Identification of the TOE

Table 1 provides information needed to completely identify the product, including:

- The Target of Evaluation (TOE), the fully qualified identifier of the product as evaluated;
- The Security Target (ST), describing the security features, claims, and assurances of the product;
- The conformance result of the evaluation;
- The organizations and individuals participating in the evaluation.

Evaluation Scheme	United States Common Criteria Evaluation Validation Scheme
Evaluated Target of Evaluation	Fortress Mesh Point: • ES210-3 810-00020-01 • ES210-4 810-00029-01 • ES520-34 810-00022-01 • ES520-35 810-00015-01 • ES820-34 810-00030-01 • ES820-35 810-00023-01 • ES2440-0 810-00046-01 • ES2440-35 810-00051-01 • ES2440-3555 810-00037-01 • ES2440-34m 810-00061-01 • ES2440-3444m 810-00060-01 All running Software Version: 5.4.5.2157
Protection Profile	• Protection Profile for Network Devices v1.1, June 8, 2012

	• Security Requirements for Network Devices Errata #3, Version 1.0, November 3, 2014 • Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013
Security Target	FORTRESS Mesh Point ES210, ES520, ES820, ES2440 Security Target, Version 1.5, February 18, 2016
Dates of Evaluation	September 8, 2015 – February 19, 2016
Conformance Result	Pass
Common Criteria Version	CC Version 3.1r3, July 2009
Common Evaluation Methodology (CEM) Version	CEM Version 3.1r3, September 2012
Evaluation Technical Report (ETR)	Common Criteria Evaluation Technical Report, DOC ID: 16-2686-R-0003 V1.1, February 19, 2016
Sponsor/Developer	General Dynamics Mission Systems
Common Criteria Testing Lab (CCTL)	InfoGard Laboratories, Inc. NVLAP Lab Code: 100432-0
CCTL Evaluators	Brad Mitchell, Scott Cutler
CCEVS Validators	Daniel Faigin, Luke Florer

Table 1: Product Identification

3 Interpretations

The Evaluation Team performed an analysis of the international interpretations of the CC and the CEM and determined that none of the International interpretations issued by the Common Criteria Interpretations Management Board (CCIMB) were applicable to this evaluation.

The TOE is also compliant with all international interpretations with effective dates on or before August 27, 2015.

4 Security Policy

This section contains the product features and denotes that are within the logical boundaries of the TOE. The following Security Functions are supported by the TOE:

- Security Audit (FAU)
- Cryptographic Support (FCS)
- User Data Protection (FDP)
- Identification and Authentication (FIA)
- Security Management (FMT)
- Protection of the TSF (FPT)
- TOE Access (FTA)

- Trusted Path/Channels (FTP)
- Packet Filtering (FPF)

4.1 Security Audit

The TOE supports remote audit logging using the syslog standard with an external server. The TOE allows the user to filter audit logs via administrator identity, event type, and user interface.

The TOE will audit all events and information defined in [ST] Section 6.1.1, Table 8: Auditable Events. The TOE will also include the identity of the user that caused the event (if applicable), date and time of the event, type of event, and the outcome of the event.

The TOE protects storage of audit information from unauthorized deletion and prevents unauthorized modifications to the stored audit records. The TOE can transmit audit data to and receive data from an external IT entity using the IPsec protocol.

The TOE can audit packet filter firewall rules. An auditable rule can be added with the “add pktfilter” command.

It is not possible for the TOE to log every dropped packet because of CPU limitations and flash writes, which are very slow. However, the TOE counts every dropped packet in a specific per-interface counter. The TOE logs the dropped packets count for each interface.

The TOE keeps 3.5 Mbytes of local audit log data in a 20 Mbyte partition. No users can access this partition. Within this space are the current log file and the two most recent log files that have been rotated. These log files are rotated as they fill up.

The process for log rotation is as follows:

- Log files are filled by audit event logs as they are generated.
- When that log file is full (i.e., there is no room for additional logs) a new log file is created to record new audit events.
- Since there are only three log files in rotation, the TSF overwrites the oldest audit log file upon audit log rotation when all three audit log files are currently full.

When the TSF sends audit log data to the external syslog server, all data is encrypted with an IPsec tunnel. The log messages are sent when they are generated. The TOE uses Syslogd 1.5.0 compatible with RFC 3164. The granularity of the timestamps is 1 second. The syslogd process sends out UDP packets tunneled within the IPSec TCP tunnel, which guarantees order of transmission. Therefore, messages are sent in the order they are generated. If there is no link or the link goes down to the audit server, the TSF adds a “Communication error” to the local log.

4.2 Cryptographic Support

The TOE uses certified cryptographic algorithms to perform all cryptographic operation. These cryptographic algorithms implements all cryptographic primitives as validated by the CAVP program, as shown in the following table.

Algorithm	Cert #	Library Version	Functionality	Modes

Algorithm	Cert #	Library Version	Functionality	Modes
AES.	1520	FPGA 2.0	IPsec (ESP) WPA2	CBC (e/d: 128, 192, 256) CCM (KS: 128) GCM (e/d: 128, 192, 256)
	3506	SSL 2.1	IPsec (IKE) WPA2 TLS SSH	ECB (e/d: 128, 192, 256) CBC (e/d: 128, 192, 256) CFB8 (e/d: 128, 192, 256) CFB128 (e/d: 128, 192, 256) OFB (e/d: 128, 192, 256)
SHS	1358	FPGA 2.0	IPsec (ESP) WPA2	SHA-1 (byte-only) SHA-384 (byte-only)
	2891	SSL 2.1	IPsec (IKE) WPA2 TLS SSH	SHA-1 (byte-only) SHA-224 (byte-only) SHA-256 (byte-only) SHA-384 (byte-only) SHA-512 (byte-only)
HMAC	890	FPGA 2.0	IPsec (ESP) WPA2	HMAC-SHA1 HMAC-SHA384
		SSL 2.1	IPsec (IKE) WPA2 TLS SSH	HMAC-SHA1 HMAC-SHA224 HMAC-SHA256 HMAC-SHA384 HMAC-SHA512
ECDSA	716	SSL 2.1	IPsec WPA2 TLS SSH	FIPS186-4: SigVer: P-256: (SHA-1, 256) P-384: (SHA-1, 384)
	833			FIPS186-4: KeyGen: P-256, P-384

Algorithm	Cert #	Library Version	Functionality	Modes
	573		IPsec WPA2 TLS	ECDSA SigGen Component: P-256, P-384
RSA	1800	SSL 2.1	TLS SSH	FIPS186-2: ALG[RSASSA-PKCS1_V1_5] SIG(ver): 2048, SHS: SHA-1
	1967			FIPS186-2: Key Gen: 2048 SIG(gen): 2048, SHA-256, SHA-384
DRBG 800-90	874	SSL 2.1	IPsec (IKE) WPA2 TLS SSH	HMAC_Based DBRG: SHA-1, SHA-256, SHA-384, SHA-512
KAS	10	KAS 1.0	IPsec (IKE)	FFC: SHA-256 ECC: P-256, SHA-256, HMAC ED: P-384, SHA-284, HMAC
DSA	1053	SSL 2.1	IPsec (IKE) WPA2 SSH	FIPS186-Key Gen: (2048,224), (2048, 256), (3072, 256)

4.2.1 IPsec

The TOE uses IPsec VPN functionality to provide wireless (and wired) clients an encrypted and authenticated tunnel to the private network. The clients can be 3rd party devices, or other TOEs, provided the IPsec implementation supports compatible cipher suites. The TOE uses IPsec to secure communications to the RADIUS server, the Syslog server, and the NTP server. When establishing a tunnel, the TOE only operates in tunnel mode.

- AES128 and AES256 with modes CBC and GCM and 128/256 bit keys respectively.
- ECDSA with curves P-256 and P-384 for peer authentication to authorized IT entities.
- DH Groups: 14 (2048-bit MODP) 19 (256-bit Random ECP), and 20 (384-bit Random ECP) for key exchange

4.2.2 TLS

The TOE uses the TLS 1.0 protocol for securing communication with the GUI through HTTPS/TLS, as well as adding additional security in communicating with the RADIUS authentication server. The TOE provides TLS for the Web Server (https) services. The authentication server provides EAP-TLS for authentication for WPA2 via x.509 certificates. The TLS implementation allows the following cipher suites:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA

4.2.3 SSH

The TOE implements the SSH protocol using OpenSSH v5.8 P1. An administrative user can authenticate with SSH public key authentication and a user name and password or with just a user name and password. When establishing an SSH tunnel, the TOE allows the following ciphers:

- Public key algorithms
    - ecdsa-sha2-nistp256
    - ecdsa-sha2-nistp384
- Encryption algorithms
    - AES-CBC-128
    - AES-CBC-256
- Data integrity algorithms
    - HMAC-SHA1
    - HMAC-SHA1-96
- Key exchange
    - diffie-hellman-group14-SHA1
    - ecdh-sha2-nistp256
    - ecdh-sha2-nistp384

4.2.4 Zeroization

The TSF zeroizes all plaintext secret and private cryptographic keys and CSPs once they are no longer required.

The configuration database is stored in a file that has been hashed using SHA160. It is then encrypted using cipher block chaining. The key used to encrypt the configuration database is stored in I2C (i.e., it is set onto the EPROM when the box is manufactured). The key on the EPROM is never zeroized. This key is never used for communication. All encrypted keys that are decrypted have their memory usage zeroized after the usage is completed by writing all 0's.

The following is a list of the secret keys (keys used for symmetric encryption), private keys, and critical security parameters used to generate keys, all of which are stored in the configuration database in a flash file system:

- Administrative passwords
- WPA2 keys
- Authentication server keys
- Device Access ID
- Public/private key pairs
- X.509 certificates
- IPsec pre-shared keys

4.2.5 Cryptographic Key Generation

For cryptographic key generation of asymmetric keys, the TOE conforms to:

NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for finite field-based key establishment schemes

NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for elliptic curve-based key establishment schemes using NIST curves P-256, P-384

The TOE conforms to NIST SP800-56A 6.1.2.1 dhEphem, C(2, 0, FFC DH) and NIST SP800-56A 6.1.2.2 dhEphem, C(2, 0, ECC CDH). The TOE conforms to FIPS PUB 186-3 Appendix B.4.2.

4.2.6 Random Bit Generation

The source of entropy for the True Random Number Generator (TRNG) is thermal jitter. The basic principle of operation is that a slow oscillator samples a fast one, and it is the thermal jitter effects present on the slow oscillator that are "measured" as the source of random entropy. The raw entropy bits are produced from these measurements.

A standard von Neumann corrector is applied to the sampled bit stream to correct a small DC bias caused by the imperfect duty cycle of the fast oscillator.

The output of the post-processor is streamed into a continuously rotating ring buffer. This ring buffer is being constantly overwritten by newly generated bits that are continuously generated by the TRNG.

The TSF provides testing that consists of the minimum entropy test from NIST SP800-90, Appendix C. The lowest allowed min-entropy is 80% or 4.8 bits entropy per 6-bit sample. Anything less than that results in a failure of the FIPS test and the device is placed into a failed state. The continuity test catches repeat values. The TSF tests the actual "randomness" by doing a min-entropy test. The RBG is always seeded with a minimum of 256 bits of entropy.

4.3 User Data Protection

The TSF ensures that any previous information content of a resource is made unavailable upon the allocation of the resource to all objects. When the TOE is constructing a protocol data unit (PDU), it makes any previous information unavailable when it is allocated for the next PDU. The PDU is not padded as a part of normal packet processing. Data passing into the system is copied from the driver that initially received that data into a PDU buffer of exactly the right size; therefore there is no need to perform padding or zeroization actions.

For IPsec:

- Only IPsec-tunnel mode is supported, so the original IP header is encrypted.
- The decrypted IPHDR.length must be less than or equal to the encrypted IPHDR.length.
- The frames are protected with a MIC.

4.4 Identification and Authentication

The TSF supports passwords consisting of alphanumeric and special characters. The TSF also allows administrators to set a minimum password length and supports passwords with 15 characters or more.

The TSF requires all administrative-users to authenticate before allowing the user to perform any actions other than:

- Viewing the warning banner
- Receiving and sending Mesh Viewer Protocol (MVP) packets every 30 seconds on port 4949

The TOE logs all unsuccessful authentication attempts as well as the interface from which the attempt originated.

The TOE can be administered with a CLI through SSH or a web based GUI over HTTP/TLS. For SSH public key authentication the TOE supports using either the ECDSA-SHA2-NISTP256 or ECDSA-SHA2-NISTP384 for public key algorithm.

A successful authentication is determined by either a successful username and password combination, or additionally required public key/certificate for SSH/TLS respectively.

When a user is entering their password information, the password is obscured such that no observer could read the password off the screen. This is done by using a circle to represent all characters while accessing the local (console via RS-232) administrative interface. The admission can be handled by either a local authenticator or a RADIUS server. No passwords are ever stored as clear text. For remote authentication, the TOE must have a connection to the RADIUS server. Communications to the RADIUS server are secured using an IPsec tunnel and the TLS protocol.

An administrative user is required to re-authenticate when that user changes their own password, and following a TSF-initiated locking as described in any of the FTA_SSL requirements in the [ST].

The TOE uses pre-shared keys for IPsec and WPA2.

A user can use X.509 certificates for TLS and IPsec authentication.

4.5 Security Management

The two remote administrative interfaces are the GUI, via TLS/HTTPS protocol, and the CLI console, via SSH protocol or local console, interfaces. These allow the administrator to perform all security functionality as required by this PP. Through the administrative interfaces, the

following roles of: administrator, maintenance, and log viewer, can access their allowed privileges and are maintained by the TSF.

The TOE maintains the Role of Authorized Administrator. This allows the administrator to administer the TOE either locally or remotely through a CLI/GUI. For users that are not administrative users, there are no TSF commands or TSF data that is available to that user except the pre-authentication access banner. Once a wireless client successfully authenticates with WPA2-PSK or EAP-TLS, that user can only elicit data through the TOE using the general WLAN functionality. This prevents any unprivileged configuration of the TOE or viewing of TSF data.

No passwords are stored in clear text. All passwords are stored as a hashed SHA-256 digest. A salt value used in conjunction with the digest cannot be seen by the user. The entire configuration database is then encrypted using cipher block chaining (AES256-CBC) with a master key. There are no clear-text keys stored that must be zeroized.

4.6 Protection of the TSF

Protection of Cryptographic Operations:

The TSF prevents the reading of secret and private keys. The TOE stores symmetric keys only in RAM, never on persistent media. While in RAM, symmetric keys are kept in an encrypted format and are decrypted/re-encrypted every time they are used. The TOE stores pre-shared keys and private keys in an encrypted data file (DBP module). The TOE admin interface does not provide any mechanism to view sensitive data (PSK, passwords, or keys) once stored. The configuration backup command permits the pre-shared/private keys to be exported. The sensitive material (PSK, keys, passwords) are encrypted with AES256 using an administrator provided password.

Reliable Time:

The TOE has the capability to obtain reliable time from a remote Network Time Protocol (NTP) server, or is maintained by the system internal clock once set by an Administrator. For auditing, session establishment, SA lifetimes and X.509 certificate revocation, the internal clock is used. The connection to the NTP server is protected by an IPsec tunnel.

The TOE runs a suite of self-tests during the initial start-up (upon power on) to demonstrate the correct operation of the TSF. Most of the time, related functions in the TOE rely on timers that count the number of “ticks” since an arbitrary point in the past. Each tick is 10ms. The TOE runs a power on test to ensure that is true. There is also a continuous test that monitors to ensure the value returned never jumps backwards. Being connected to an NTP server ensures that the system time is accurate for the time related functions that uses a timestamp (audit log and X509 certificate revocation).

Trusted Updates:

The TOE provides a means to verify firmware/software updates to the TOE using a digital signature mechanism and published hash prior to installing those updates.

Users can query the firmware/software version of the TOE and an authorized administrator can initiate updates to the TOE. When performing the update, the TOE verifies the integrity of the update with either a digital signature or a published hash. For digital signatures, the TOE compares the update files' signature using a certificate that comes pre-loaded on the device. As part of the build process, the update image is signed with a private key by the TOE developer. In this system, the "authorized source" is defined as the holder of the private key, thus making Fortress the only authorized source for updated images. This is done with either RSA 2048 or ECDSA 256/384. For published hashes, the update files are hashed with SHA-512 and the hash and update files are encrypted together with AES-256. The TOE decrypts the image, calculates the hash and compares the hash with the one provided. Only if the signature/hash is correct, will the image be installed. If an update is unsuccessful, a message is delivered to the user. Since the update process attempts to update a different partition than what is currently being run, the current active partition remains the same and the user continues to run the same code that was being run before the upgrade attempt was made.

If the TOE fails the integrity check of a new image update, it will not install the new software image. If the TOE fails the integrity check of the current image while rebooting, it will not load the image.

Self-Tests:

The TOE runs a suite of self-tests on start-up The failure of any critical security component self-test causes the TOE to enter an error state during which no traffic may flow. The following is a list of self-tests performed by the TOE:

- RAM Test
- Flash Test
- Firmware Integrity Test
- EEPROM Test
- I2C Test
- MDIO Test
- PCI Test
- IDE Test
- RTC Test
- Watchdog Test
- IRQ Test
- FPGA Test
- TPM Test

The TOE performs the following run-time DRBG self-tests:

- Instantiate Function
- Generate Function
- Reseed Function
- Uninstantiate Function

These self-tests are essentially known answer tests that verifies that for a known input the calculated output matches the expected output. For all these known answer tests, the proper execution of the error handling is also verified before the test is considered "passed".

For key material and user data, the most critical security-related tests, such as the TPM test, the FPGA test, and any of the FIPS required tests, causes the box to stop operation as soon as the failure is detected.

Once the TOE has completed the boot process, the entire suite of known answer tests and continuous tests are run. All tests must pass before the TOE begins handling user data or the administrator is able to log in.

When configured to run in FIPS[1] mode, upon a self-test failure, the TOE logs the failure and causes the box to reboot. If the failure is a persistent failure the box will be stuck in rolling reboot without any packets passing from then on.

User Data:

A successful authentication is determined by either a successful username and password combination, or additionally required public key/certificate for SSH/TLS respectively. A failure to find a public key and/or incorrect password will result in a failed authentication attempt. When a user is entering their password information, the password is obscured such that no observer could read the password off the screen. This is done by using a circle to represent all characters while accessing the local (console via RS-232) administrative interface. The admission can be handled by either a local authenticator or a RADIUS server. In the local case, passwords entered are converted into a SHA-256 digest using a salt value. This is compared to the digest value for that user. No passwords are ever stored as clear text. For remote authentication, the TOE must have a connection to the RADIUS server. Communications to the RADIUS server are secured using an IPsec tunnel and the TLS protocol. An administrative user is required to re-authenticate when that user changes their password.

4.7 TOE Access

For TOE administration, the GUI (TLS/HTTPS), CLI (SSH) and local console CLI are available. Prior to an administrative user authenticating, that user is presented with an access display banner that displays an advisory notice and consent warning message regarding unauthorized use of the TOE. The TOE, for both local and remote interactive sessions, will terminate the session after an Authorized Administrator-specified period of session inactivity. The TOE allows Administrator-initiated termination of the Administrator's own interactive session.

4.8 Trusted Path/Channels

Trusted Channel

The TOE permits the TSF, or the authorized IT entities to initiate communication via the trusted channel. The TSF trusted channel provides assured identification of its endpoints, protects the channel data from disclosure, and detects modification of the channel data. This trusted channel is provided via IPsec, and protects RADIUS, syslog, and NTP.

Trusted Path

The TOE permits remote administrators to initiate communication via the trusted path. For TOE administration, the GUI, SSH (CLI) and local console CLI are available. The GUI and the remote

[1] Does not imply FIPS 140-2 Validation by the CMVP

CLI interfaces are secured using TLS/HTTPS and SSH respectively. TLS is not included for all IT entities because they are already secured within the IPSec tunnel. The TOE requires the use of the trusted path for initial administrator authentication and all remote administration actions.

4.9 Packet Filtering

The TOE performs Packet Filtering on network packets processed by the TOE. When the TOE starts-up it takes the actions listed in the "Protection of the TSF" section listed above (self-tests). Those actions include tests at firmware boot time and tests at software boot time. No packets flow during firmware or software boot until all of the software known answers and entropy tests have passed.

The mechanism that prevents a packet from flowing is a global value that contains the FIPS Status. If that Status is anything other than "OK", the Frame Processor will not put any interfaces into the Forwarding state. There are two mechanisms that prevent packets from flowing before the TSF is fully initialized and ready to start processing packets. While the TSF is booting, each interface initializes in the disabled/down state where it will not process any packets. The TSF loads and sets the packet processing and packet filtering rules for each interface prior to enabling the interface. Additionally, the Frame Processor (packet forwarding and filtering engine) does not begin processing any packets until the global FIPS[2] status value is set to "OK". The FIPS Status is only set to "OK" once the startup self-tests have completed successfully.

Each time a packet comes into an interface (ingress) or is about to be sent out an interface (egress), the TOE applies the Packet Filtering rules. The TOE uses "flows" to optimize packet forwarding. A flow for IPv4/IPv6 frames is identified by:

- Source Address
- Destination Address
- Protocol
- Source Port (TCP/UDP)
- Destination Port (TCP/UDP)

For performance reasons, the TOE does not issue a log every time a packet is dropped. Instead, a counter is kept in each flow. The log will note how many packets have matched during the polling cycle (8 seconds). The log contains all the identifying information of the flow and the rule.

IPv4, IPv6, TCP, and UDP in the software are implemented in the Linux kernel stack. Interoperability testing is performed with Windows and other Linux distributions. For this reason, it has been determined that the TOE conforms to:

- RFC 791 (IPv4)
- RFC 2460 (IPv6)

[2] Does not imply FIPS 140-2 Validation by the CMVP.

- RFC 793 (TCP)
- RFC 768 (UDP)

5 TOE Security Environment

5.1 Secure Usage Assumptions

The following assumptions are made about the usage of the TOE:

A.NO_GENERAL_PURPOSE	It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available to the TOE, other than those services necessary for the operation, administration and support of the TOE.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

5.2 Threats Countered by the TOE

The TOE is designed to counter the following threats:

T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress- or egress-based actions.
T.NETWORK_ACCESS	Unauthorized access may be achieved to services on a protected network from outside that network, or alternately services outside a protected network from inside the protected network
T.NETWORK_MISUSE	Access to services made available by a protected network might be used counter to Operational Environment policies.
T.REPLAY_ATTACK	If malicious or external IT entities are able to gain access to the network, they may have the ability to capture information traversing throughout the network and send them on to the intended receiver.
T.DATA_INTEGRITY	A malicious party attempts to change the data being sent – resulting in loss of integrity.

5.3 Organizational Security Policies

The TOE enforces the following OSPs:

P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

5.4 Security Objectives for the TOE

O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource

	is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
O.ADDRESS_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination addresses.
O.AUTHENTICATION	The TOE will provide a means to authenticate the user to ensure they are communicating with an authorized external IT entity.
O.CRYPTOGRAPHIC_FUNCTIONS	The TOE will provide means to encrypt and decrypt data as a means to maintain confidentiality and allow for detection and modification of TSF data that is transmitted outside of the TOE.
O.FAIL_SECURE	Upon a self-test failure, the TOE will shutdown to ensure data cannot be passed while not adhering to the security policies configured by the administrator.
O.PORT_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination transport layer ports.

5.5 Clarification of Scope

All evaluations (and all products) have limitations, as well as potential misconceptions that need clarifying. This text covers some of the more important limitations and clarifications of this evaluation. Note that:

1. As with any evaluation, this evaluation only shows that the evaluated configuration meets the security claims made, with a certain level of assurance (the assurance activities specified in the claimed PPs and performed by the evaluation team).
2. This evaluation covers only the specific software version identified in this document, and not any earlier or later versions released or in process.
3. The evaluation of security functionality of the product was limited to the functionality specified in the claimed ST. Any additional security related functional capabilities of the product were not covered by this evaluation.
4. This evaluation did not specifically search for, nor attempt to exploit, vulnerabilities that were not “obvious” or vulnerabilities to objectives not claimed in the ST. The CEM

defines an “obvious” vulnerability as one that is easily exploited with a minimum of understanding of the TOE, technical sophistication and resources.

5. This evaluation covers only the VPN Gateway functionality specified by the VPN EP to the NDPP and the basic network device functionality specified by the NDPP v1.1 Err 3. All use of the following features and capabilities has not been evaluated and no conclusions should be drawn from this evaluation about their correctness:
    - MeshPoint protocol
    - Fortress MSP
    - Power over Ethernet
    - USB host connectors or other USB ports
    - WLAN functionality (evaluated in VIDs 10571, 10572, and 10573)
    - GPS
    - DHCP server
    - DNS services
    - QoS
    - VLANs
    - Device Access Control
    - Fortress Mesh Viewer Protocol
    - Layer 2 link management (e.g. Spanning Tree Protocol, etc.)

6 Architectural Information

The TOE is classified as a Network Device for Common Criteria purposes. The TOE is made up of hardware and software components.

6.1 Architecture Overview

The TOE consists of hardware and software components.

6.1.1 TOE Hardware

6.1.1.1 ES210

The ES210 acts as a layer 2 bridge with VPN functionality and a wireless access point. The ES210 can operate at the given frequencies and data link protocols listed above in Section 1.2. The physical boundaries of the ES210 are the interfaces of the TOE module, listed below:

- RJ45 10/100BT Ethernet Port (2)
    - Provides a port for the user to access the network as well as allows access to the management functionality with administrative user authentication. The only difference between the two ports is that the port labeled “Ethernet1/WAN” is encrypted by default while the other is not.
- 3 Pin Con-X Serial Connector (3 pin mil-spec round connector)
    - Local CLI management interface
- 2 Pin Con-X Power Connector (2 pin mil-spec round connector)

    - Provides power to the ES210
- RP-TNC Antenna Connector (1)
    - For the various antenna options described in [ST] Section 1.4.1.1
- SMA Connector
    - GPS antenna

Indicators are used to allow the operator to have a quick indication of the state of the ES210:

- Power: Indicates the power status of the TOE
- Battery: Indicates the charge state of the battery
- Ethernet1/Ethernet 2 – Link/Activity: Indicates the status and activity of the Ethernet port
- Radio activity: Indicates activity on that radio position

The ES210 also has the following physical button controls:

- Power On/Off: Allows the device to be powered
- Blackout Mode: Turns off all LED indicators
- RF Kill: Turns all radio transmissions off
- Zeroize: Restores factory defaults

6.1.1.2 ES520

The ES520 acts as a layer 2 bridge with VPN functionality and a wireless access point. The ES520 can operate at the given frequencies and data link protocols listed in [ST] Section 1.4.1.1. The physical boundaries of the ES520 are the interfaces of the TOE module:

- RJ45 10/100BT Ethernet Port (8)
    - Provides a port for the user to access the network as well as allows access to the management functionality with administrative user authentication. The only difference between the two ports is that the port labeled (WAN) is encrypted by default while the other is not.
- USB Host Connector
    - This is excluded in the CC evaluated configuration.
- 10/100BT WAN Port (1)
    - Provides a port for the user to access the network as well as allows access to the management function with administrative user authentication.
- 3 Pin Con-X Serial Connector (3 pin mil-spec round connector)
    - Local CLI management interface
- DC Power Input Connector
    - Provides power to the ES520
- N-type Antenna Connector (2)
    - For the various antenna options described in Section Radio Configurations

Indicators are used to allow the operator to have a quick indication of the state of the ES520:

- Power: Indicates the power status of the TOE
- Clr: Excluded

- Status 1: Indicates system status
- Status 2: Excluded
- Fail: Excluded
- Radio1/Radio2 (Upper): Indicates the activity on the radio
- Radio1/Radio2 (Lower): Excluded

The ES520 also has the following controls:

- Reset Button: Power cycles the TOE.

6.1.1.3 ES820

The ES820 acts as a layer 2 bridge with VPN functionality and a wireless access point. The ES820 can operate at the given frequencies and data link protocols listed above in [ST] Section 1.4.1.1. The physical boundaries of the ES820 are the interfaces of the TOE module:

- MIL Connector, includes the following interfaces:
    - RJ45 10/100BT Ethernet Port (2)
        - Provides a port for the user to access the network as well as allows access to the management functionality with administrative user authentication. The only difference between the two ports is that the port labeled (WAN) is encrypted by default while the other is not.
    - USB
        - This is excluded in the CC evaluated configuration.
    - Serial
        - Local CLI management interface
    - All LED indicators
    - All Controls
- 3 Pin Con-X Serial Connector (3 pin mil-spec round connector)
    - Supplies power to the TOE
- N-type Antenna Connector (2)
    - For the various antenna options described in Section Radio Configurations

Indicators are used to allow the operator to have a quick indication of the charge state of the ES820. The following indicators are available through the MIL connector:

- Power: Indicates the power status of the TOE
- Status: Excluded
- Ethernet1/Ethernet 2 – Link/Activity: Indicates the status and activity of the Ethernet port
- Radio activity: Indicates activity on that radio position

The ES820 has the following input functions by means of the MIL connector:

- Power On/Off: Allows the device to be powered
- Blackout Mode: Turns off all LED indicators
- RF Kill: Turns all radio transmissions off
- Reset: Power cycles the device
- Zeroize: Restores factory defaults

6.1.1.4 ES2440

The ES2440 acts as a layer 2 bridge with VPN functionality and a wireless access point. The ES2440 can operate at the given frequencies and data link protocols listed above in [ST] Section 1.4.1.1. The physical boundaries of the ES2440 are the interfaces of the TOE module:

- RJ45 10/100/1000BT Ethernet Port (3)
  - Provides a port for the user to access the network as well as allows access to the management functionality with administrative user authentication. The only difference between the first port and the other two ports is that the port labeled (Ethernet1/WAN/POE) allows power over Ethernet (802.3af), and the others do not.
- RJ45 Serial Connector
  - Local CLI management interface
- 2 Pin Con-X Power Connector (2 pin mil-spec round connector)
  - Provides power to the ES2440
- N-type Antenna Connector (8)
  - For the various antenna options described in Section Radio Configurations
- SMA Connector
  - GPS antenna

Indicators are used to allow the operator to have a quick indication of the state of the ES2440:

- Power: Indicates the power status of the TOE
- Ethernet1/Ethernet 2/Ethernet3 link/activity – Link/Activity: Indicates the status and activity of the Ethernet port
- Radio1/Radio2/Radio3/Radio4 activity: Indicates activity on that radio position

The ES2440 also has the following physical button controls:

- Recessed Button: Restores factory defaults

6.1.2 TOE Software

Software Version: 5.4.5.2157

6.1.3 Operational Environment Requirements

Table 2 below identifies components that must be present in the Operational Environment to support the operation of the TOE:

Component	Description

Syslog Server	• Compatible with RFC 3164 • Supporting IPsec as defined in [ST] Section 6.1.2.8 "FCS_IPSEC_EXT.1 IPsec"
NTP Server	• V4 conformant to RFC 5905 with a SHA-1 authentication[3]

Remote Console	• **GUI access**
	  ○ Firefox v3.6 to 14
	  ○ IE version 7-9
	  ○ Compatible with HTTPS implementing:
	  ▪ HTTPS protocol that complies with RFC 2818
	  ▪ TLS 1.0 (RFC 2246)
	  ○ Compatible with TLS using the following:
	  ▪ Mandatory cipher suites:
	  • TLS_RSA_WITH_AES_128_CBC_SHA
	  ▪ Optional cipher suites:
	  • TLS_RSA_WITH_AES_256_CBC_SHA
	  • TLS_DHE_RSA_WITH_AES_128_CBC_SHA
	  • TLS_DHE_RSA_WITH_AES_256_CBC_SHA
	• **SSH**
	  ○ V2 client compatible with the list of required ciphers (as listed in [ST] Section 6.1.2.10 "FCS_SSH_EXT.1 SSH").

Local Console	RS-232 Console Port compatible with the following enumeration settings: • bits per second: 9600 • data bits: 8 • parity: none • stop bits: 1 • hardware flow control: none
Ethernet	Ethernet Client Hardware Requirements: • 10BASE-T/100BASE-TX Base Ethernet
Wireless	Wireless Client Hardware/Firmware Requirements: • Wireless 2.4GHz, 4.4GHz, 4.9GHz, or 5.0GHz, IEEE 802.11 a/b/g/n (depending on radio utilization) • WPA2 (a security protocol and security certification program developed by the Wi-Fi Alliance to secure wireless computer networks) Antennae: • ES210 and ES2440 Specific (not in ES520, 820): GPS antenna with SMA connector • Wi-Fi Antenna with N-type connector • Capable of transmitting and receiving on the required frequency as described in [ST] Section 1.4.1.1

[3] SHA-1 authentication for NTP was not evaluated and therefore cannot claim any cryptographic security.

Authentication Server	• RADIUS server (compatible with RFC 2865, supporting IPsec as defined in [ST] Section 6.1.2.8 "FCS_IPSEC_EXT.1 IPsec")
VPN Client	• Using an IPsec client that is compatible with the requirements defined in [ST] Section 6.1.2.8 "FCS_IPSEC_EXT.1 IPsec"

Table 2: Operational Environment Components

7 Documentation

This section details the documentation that is (a) delivered to the customer, or (b) was used as evidence for the evaluation of the TOE. The "Guidance Documentation" is delivered with the TOE and was included within the scope of the evaluation. Those documents are the only ones that should be trusted for configuring or administering the TOE.

7.1 Guidance Documentation

Document	Revision	Date
Fortress Common Criteria Operational Guidance	1.8	February 19, 2016
Fortress Mesh Point Software CLI Guide	009-00036 00v5.4.5	N/A
Fortress Mesh Point Software GUI Guide	009-00035-00v5.4.5	N/A

7.2 Test Documentation

Document	Revision	Date
Independent Test Report, 15-2686-R-0025	V1.3	February 19, 2016

7.3 Vulnerability Assessment Documentation

Document	Revision	Date
Independent Test Report, 15-2686-R-0025	V1.3	February 19, 2016

7.4 Security Target

Document	Revision	Date
FORTRESS Mesh Point ES210, ES520, ES820, ES2440 Security Target	Version 1.5	February 18, 2016
CC Entropy Description	N/A	August 2015

8 IT Product Testing

This section describes the testing efforts of the Developer and the Evaluation Team.

8.1 Evaluation Team Independent Testing

The CCTL (InfoGard Laboratories, Inc.) generated the testing plan and designed the testing activities specified in the Protection Profile for Network Devices v1.1, June 8, 2012 and the Security Requirements for Network Devices Errata #3, November 3, 2014 and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013, and generated automated and manual tests to execute the designed test plan.

The evaluation team verified the product conformities during the period September 8, 2015 – January 26, 2016 at the CCTL according to the FORTRESS Mesh Point ES210, ES520, ES820, ES2440 Security Target, Version 1.5, February 18, 2016, and ran the tests specified in the Protection Profile for Network Devices v1.1, June 8, 2012, the Security Requirements for Network Devices Errata #3, November 3, 2014, and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013 documents.

The test configurations and tools used to evaluate the TOE are described in the Assurance Activity Report (AAR) Section 5, “Testing Environment”.

8.2 Vulnerability Analysis

All testing assurance activities and vulnerability assessment (AVA_VAN) activities were performed against the TOE by the CCTL. A thorough report of vulnerability assessment activities may be found in the AAR Section 4.

The CCTL developed a custom testing environment for ND-based evaluations that uses several virtual machines, isolated networks, and smart switches in order to meet the requirements stated by the testing assurance activities.

Based on discussions in both [TRRT2] and [TD13], the evaluator reduced the scope of testing to the following components:

IPv4

- Type of Service
- Total Length
- Identification
- Flags
- Fragment Offset
- Time to Live
- Transport Layer Protocol

IPv6

- Traffic Class
- Flow Label
- Next Header
- Hop Limit

The evaluator used a custom tool based on Scapy, "fuzzIPv4.py," to perform IPv4 fuzz testing and confirmed that no erratic or unusual behavior occurred on the TOE as a result of the fuzzing, and that the TOE dropped all traffic generated by the fuzzing tool by capturing traffic on the destination of the fuzz traffic and verifying no packets were received.

The evaluator used a customer tool based on Scapy, “fuzzIPv6.py”, to perform IPv6 fuzz testing and confirmed that no erratic or unusual behavior occurred on the TOE as a result of the fuzzing, and that the TOE dropped all traffic generated by the fuzzing tool by capturing traffic on the destination of the fuzz traffic and verifying no packets were received.

9 Results of the Evaluation

The evaluation was carried out in accordance with the Common Criteria Evaluation and Validation Scheme (CCEVS) processes and procedures. The TOE was evaluated against the criteria contained in the Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 3. The evaluation methodology used by the Evaluation Team to conduct the evaluation is the Common Methodology for Information Technology Security Evaluation, Version 3.1 Revision 3.

InfoGard has determined that the TOE meets the security criteria in the Security Target, which claims compliance with the Protection Profile for Network Device Protection Profile, Version 1.1, June 8, 2012, the Security Requirements for Network Devices Errata #3, November 3, 2014, and the Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013.

A team of Validators, on behalf of the CCEVS Validation Body, monitored the evaluation. The evaluation was completed on February 19, 2016.

10 Validator Comments/Recommendations

Please note that the proprietary “Fortress Mobile Security Protocol (MSP)” must be disabled to place the TOE into the evaluated configuration. Instructions for doing so are found in the Fortress Mesh Point Software CLI Guide, sections 3.9 and 4.1.

Also, the versions of browsers shown as requirements in the Operational Environment are all no longer supported by their developers, including security support. Users of the TOE should only utilize current browsers unless additional mitigating precautions are taken.

As was noted in the Clarification of Scope section of this report, the devices provide more functionality than was covered by the evaluation. Only the functionality claimed in the SFR’s in the Security Target was evaluated. All other functionality provided by the devices needs to be assessed separately and no further conclusions should be drawn as to their effectiveness, nor can any claims be made relative to their security based upon this evaluation.

11 Security Target

FORTRESS Mesh Point ES210, ES520, ES820, ES2440 Security Target, Version 1.5, February 18, 2016

12 Terms

12.1 Acronyms

CC	Common Criteria
CCTL	Common Criteria Testing Laboratory
CSP	Critical Security Parameters
DAC	Discretionary Access Control
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standards Publication 140-2
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
I/O	Input/Output
MIB	Management Information Base
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
OCSP	Online Certificate Status Protocol
PP	Protection Profile
SF	Security Functions
SFR	Security Functional Requirements
SSH	Secure Shell
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions

13 Bibliography

[1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated July 2009, Version 3.1 Revision 3, CCMB-2009-07-001.

[2] Common Criteria (CC) for Information Technology Security Evaluation – Part 2: Security functional components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-002.

[3] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, July 2009, Version 3.1, Revision 3, CCMB-2009-07-003.

[4] Common Methodology for Information Technology Security Evaluation – Evaluation methodology, July 2009, Version 3.1, Revision 3, CCMB-2009-07-004.

[5] Protection Profile for Network Devices, June 8, 2012, Version 1.1.

[6] Security Requirements for Network Devices Errata #3, November 3, 2014.

[7] Network Device Protection Profile Extended Package VPN Gateway, Version 1.1, April 12, 2013.