Assurance Activities Report
For a Target of Evaluation

Forescout v8.3

Assurance Activities Report (AAR)
Version 1.0

July 19, 2022

For
Security Target (Version 2.0)

Evaluated by:

Booz | Allen | Hamilton

delivering results that endure

Cyber Assurance Testing Laboratory
1100 West Street
Laurel, MD 20707
NIAP Lab # 200423

Prepared for:
National Information Assurance Partnership
Common Criteria Evaluation and Validation Scheme

The Developer of the TOE:
Forescout Technologies, Inc.,
190 West Tasman Drive,
San Jose, CA 95134 USA

The Author of the Security Target:
Booz Allen Hamilton,
1100 West Street,
Laurel, 20707 USA

The TOE Evaluation was sponsored by:
Forescout Technologies, Inc.,
190 West Tasman Drive,
San Jose, CA 95134 USA

Evaluation Personnel:
Christopher Gugel, CC Technical Director
Christopher Rakaczky

Applicable Common Criteria Version

Common Criteria for Information Technology Security Evaluation, April 2017 Version 3.1 Revision 5

Common Evaluation Methodology Version

Common Criteria for Information Technology Security Evaluation, Evaluation Methodology, April 2017 Version 3.1 Revision 5

Table of Contents

1 Purpose

The purpose of this document is to serve as a non-proprietary attestation that this evaluation has satisfied all of the TSS, AGD, ATE and AVA Assurance Activities required by the Protection Profiles/Extended Packages to which the TOE claims exact conformance.

2 TOE Summary Specification Assurance Activities

The evaluation team completed the testing of the Security Target (ST) '*Forescout v8.3 Security Target v2.0'* and confirmed that the TOE Summary Specification (TSS) contains all Assurance Activities as specified by the 'Collaborative Protection Profile for Network Devices Version 2.2e' (NDcPP). The evaluators were able to individually examine each SFR's TSS statements and determine that they comprised sufficient information to address each SFR claimed by the TOE as well as meet the expectations of the NDcPP Assurance Activities.

Through the evaluation of ASE_TSS.1-1, described in the ETR, the evaluators were able to determine that each SFR was described in enough detail to demonstrate that the TSF addresses the SFR. However, in some cases the Assurance Activities that are specified in the claimed source material instruct the evaluator to examine the TSS for a description of specific behavior to ensure that each SFR is described to an appropriate level of detail. The following is a list of each SFR, the TSS Assurance Activities specified for the SFR, and how the TSS meets the Assurance Activities. Additionally, each SFR is accompanied by the source material NDcPP that defines where the most up-to-date TSS Assurance Activity was defined.

FAU_GEN.1 – *"For the administrative task of generating/import of, changing, or deleting of cryptographic keys as defined in FAU_GEN.1.1c, the TSS should identify what information is logged to identify the relevant key.*

For distributed TOEs the evaluator shall examine the TSS to ensure that it describes which of the overall required auditable events defined in FAU_GEN.1.1 are generated and recorded by which TOE components. The evaluator shall ensure that this mapping of audit events to TOE components accounts for, and is consistent with, information provided in Table 1, as well as events in Tables 2, 4, and 5 (where applicable to the overall TOE). This includes that the evaluator shall confirm that all components defined as generating audit information for a particular SFR should also contribute to that SFR as defined in the mapping of SFRs to TOE components, and that the audit records generated by each component cover all the SFRs that it implements."

Section 8.1.1 of the ST includes an example audit record for the generating/import of, changing, or deleting of cryptographic keys. The audit record contains the fingerprint of the key being acted upon. The record also provides the "Issued To" and "Issued By" information to also help in the identification of the certificate.

FAU_GEN.2 – *"The TSS and Guidance Documentation requirements for FAU_GEN.2 are already covered by the TSS and Guidance Documentation requirements for FAU_GEN.1."*

FAU_STG_EXT.1 – *"The evaluator shall examine the TSS to ensure it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.*

The evaluator shall examine the TSS to ensure it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access.

The evaluator shall examine the TSS to ensure it describes whether the TOE is a standalone TOE that stores audit data locally or a distributed TOE that stores audit data locally on each TOE component or a distributed TOE that contains TOE components that cannot store audit data locally on themselves but need to transfer audit data to other TOE components that can store audit data locally. The evaluator shall examine the TSS to ensure that for distributed TOEs it contains a list of TOE components that store audit data locally. The evaluator shall examine the TSS to ensure that for distributed TOEs that contain components which do not store audit data locally but transmit their generated audit data to other components it contains a mapping between the transmitting and storing TOE components.

The evaluator shall examine the TSS to ensure that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE shall also be detailed in the TSS.

The evaluator shall examine the TSS to ensure that it details whether the transmission of audit information to an external IT entity can be done in real-time or periodically. In case the TOE does not perform transmission in real-time the evaluator needs to verify that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.

For distributed TOEs the evaluator shall examine the TSS to ensure it describes to which TOE components this SFR applies and how audit data transfer to the external audit server is implemented among the different TOE components (e.g. every TOE components does its own transfer or the data is sent to another TOE component for central transfer of all audit events to the external audit server).

For distributed TOEs the evaluator shall examine the TSS to ensure it describes which TOE components are storing audit information locally and which components are buffering audit information and forwarding the information to another TOE component for local storage. For every component the TSS shall describe the behaviour when local storage space or buffer space is exhausted."

Section 8.1.2 of the TSS states the TOE is standalone product and is responsible for storing and sending its own generated audit records. The TSF provides near real-time forwarding of the audit trail to an external audit server in the operational environment using a TLS channel.

Section 8.1.2 of the TSS states for record storage and overflow protection:

- The application layer audit is stored in the TOE's database (DB). The DB, as part of the installation, determines a maximum size for audit store based on hard drive availability. The TSF automatically initiates a FIFO (first-in-first-out) database purge function, when 75 percent of this threshold is reached for the internal database which stores the application-level audit to prevent the audit store from becoming full.

- When the OS log file reaches the maximum size, the log file is closed and renamed sequentially (i.e. audit.log.1, audit.log.2). Therefore, with 5 audit logs and a maximum file size of 50MB each this would result in 5*50MB= 250MB of total audit space required for the OS logs. Once the number of log files reaches its configured maximum amount, the oldest log file is automatically deleted, and the remaining log files roll over in order to allow the new file to be created for the new audit records.

Section 8.1.2 of the TSS states for record protection against unauthorized access that the TOE does not provide a means for any user to manually delete or manipulate the audit logs stored at the OS level or those in the internal DB. The management interfaces (Console or CLI) do not allow the audit records to be modified or deleted. The audit functionality starts automatically with the TOE and cannot be disabled by any means.

FCS_CKM.1 – *"The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE. If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme."*

Section 8.2.1 of the TSS states the support for two key generations schemes:

- RSA scheme that meets FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 and supports using a key size of 2048 bits. The RSA scheme is used to create keys for TLS connections.

- FFC schemes Diffie-Hellman group 14 that meets RFC 3526, Section 3 and supports using a key size of 2048 bits. The Diffie-Hellman key establishment scheme is used to create keys for SSH.

FCS_CKM.2 – TD0580 – *"The evaluator shall ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme. It is sufficient to provide the scheme, SFR, and service in the TSS.*

The intent of this activity is to be able to identify the scheme being used by each service. This would mean, for example, one way to document scheme usage could be:

Scheme	*SFR*	*Service*
RSA	*FCS_TLSS_EXT.1*	*Administration*
ECDH	*FCS_SSHC_EXT.1*	*Audit Server*
ECDH	*FCS_IPSEC_EXT.1*	*Authentication Server*

The information provided in the example above does not necessarily have to be included as a table but can be presented in other ways as long as the necessary data is available."

Section 8.2.2 of the TSS states that RSA key establishment scheme is used as well as Diffie-Hellman group 14 schemes which match the FCS_CKM.1 claims. The RSA schemes is used to establish keys for TLS in support of remote administration using the Console, authentication requests to the external authentication server, and exporting of audit data (FCS_TLSC_EXT.1 and FCS_TLSS_EXT.1).

Additionally, the TSS states that Diffie-Hellman group 14 schemes are used to establish keys for SSH for support of remote administration (FCS_SSHS_EXT.1). The Diffie-Hellman group 14 support is provided by the OpenSSL cryptographic module and is implemented by using the KexAlgorithms parameter as specified in RFC 3526 Section 3. SSH is also forced to only work with DH group 14. This is hardcoded with no ability for an administrator to modify the settings.

FCS_CKM.4 – *"The evaluator examines the TSS to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case. For the purpose of this Evaluation Activity the relevant keys are those keys that are relied upon to support any of the SFRs in the Security Target. The evaluator confirms that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for). In particular, if a TOE claims not to store plaintext keys in non-volatile memory then the evaluator checks that this is consistent with the operation of the TOE.*

The evaluator shall check to ensure the TSS identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs).

Note that where selections involve 'destruction of reference' (for volatile memory) or 'invocation of an interface' (for non-volatile memory) then the relevant interface definition is examined by the evaluator to ensure that the interface supports the selection(s) and description in the TSS. In the case of non-volatile memory, the evaluator includes in their examination the relevant interface description for each media type on which plaintext keys are stored. The presence of OS-level and storage device-level swap and cache files is not examined in the current version of the Evaluation Activity.

Where the TSS identifies keys that are stored in a non-plaintext form, the evaluator shall check that the TSS identifies the encryption method and the key-encrypting-key used, and that the key-encrypting-key is either itself stored in an encrypted form or that it is destroyed by a method included under FCS_CKM.4.

The evaluator shall check that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below). Note that reference may be made to the Guidance Documentation for description of the detail of such cases where destruction may be prevented or delayed.

Where the ST specifies the use of "a value that does not contain any CSP" to overwrite keys, the evaluator examines the TSS to ensure that it describes how that pattern is obtained and used, and that this justifies the claim that the pattern does not contain any CSPs."

Section 8.2.3 of the TSS includes the Crypto key destruction table (Table 22), which lists the following keys: Diffie-Hellman Shared Secret, Diffie-Hellman private-exponent, SSH Session Key, SSH Server Host Private Key, and TLS Server Host Certificate Private Key along with their origin, storage location, and relevant destruction situations.

Section 8.2.3 states that for RAM storage the key is destroyed by a single direct overwrite consisting of Zeroes. (0x00). If the read-verify fails, the process is repeated. The key is zeroized immediately after it is no longer needed and when the TOE is shutdown or reinitialized. For system local storage the TSS section Table 22 describes two instances: (1) (SSH Server Host Private Key) the generation of a new certificate will only be accomplished during a reinstallation of the product where all files would be overwritten which would in effect also destroy the abstraction that represented the key and (2) (TLS Server Host Certificate Private Key) Private key is deleted when generation of a new certificate are imported or when certificates are removed. The TOE will invoke an interface, provided as part of the TSF, that instructs a part of the TSF to destroy the abstraction that represents the key (i.e. delete the resource). The description includes the APIs called by the TOE to perform zeroization function on volatile memory and describes the destruction of keys on the filesystem.

The ST does not select "destruction of reference" or "invocation of an interface". There are no claims for storage of keys in an encrypted form. Section 8.2.3 states there are no known instances where key destruction does not happen as defined. The ST does not select "a value that does not contain any CSP".

FCS_COP.1/DataEncryption – *"The evaluator shall examine the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption."*

Section 8.2.4 of the TSS identifies the key sizes and modes supported by the TOE for data encryption/decryption. Specifically, the AES modes are CTR, CBC, and GCM. The cryptographic key sizes are 128 and 256 bits for each AES mode per cryptographic support are as follows:

- OpenSSL supports:
  - TLS communication: AES-CBC-128, AES-CBC-256, AES-GCM-256
  - SSH communication: AES-CTR-128, AES-CTR-256, AES-GCM-128, AES-GCM-256
  - CTR DRBG: AES-CTR-256
- Bouncy Castle supports:
  - TLS communication: AES-CBC-128, AES-CBC-256, AES-GCM-256.

FCS_COP.1/SigGen – *"The evaluator shall examine the TSS to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services."*

Section 8.2.5 of the TSS specifies RSA as the cryptographic algorithm with key size (modulus) 2048 bits used by the TOE for digital signature services generation and verification. This is applicable to both cryptographic libraries being implemented.

FCS_COP.1/Hash – *"The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS."*

Section 8.2.6 of the TSS states that the SHA-1, SHA-256, SHA-384, and SHA-512* cryptographic hashing functions are used for password hashing of all passwords stored on the TOE (FPT_APW_EXT.1), trusted updates digital signature verification (FPT_TUD_EXT.1), and TSF self-testing hash value check verification (FPT_TST_EXT.1).

*Only OpenSSL provides the SHA-512 hashing support. Meaning:

- OpenSSL supports: SHA-1, SHA-256, SHA-384, and SHA-512
- Bouncy Castle supports: SHA-1, SHA-256, and SHA-384

FCS_COP.1/KeyedHash – *"The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used."*

Section 8.2.7 of the TSS states that for HMAC-SHA-1, the key length is 160 bits, the block size is 512 bits, and output MAC length is 160 bits. For HMAC-SHA-256, the key length is 256 bits, the block size is 512 bits, and output MAC length is 256 bits. For HMAC-SHA-384, the key length is 384 bits, the block size is 1024 bits, and output MAC length is 384 bits. For HMAC-SHA-512*, the key length is 512 bits, the block size is 1024 bits, and output MAC length is 512 bits.

*Only OpenSSL provides HMAC-SHA-512 keyed-hashing message authentication. Meaning:

- OpenSSL supports: HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512
- Bouncy Castle supports: HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-384.

FCS_RBG_EXT.1 – *"The evaluator shall examine the TSS to determine that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min-entropy contained in the combined seed value."*

Section 8.2.8 of the TSS states that the TOE implementation of Bouncy Castle uses a hash deterministic random bit generator (Hash_DRBG). The TOE implementation of OpenSSL uses a counter mode random be generator (CTR DRBG). Both DRBG used by the TOE are in accordance with ISO/IEC 18031:2011. There is no ability to specify the use of an alternative DRBG. The TSS specifies that the entropy pool is software-based from four different noise sources (interrupt events, disk events, keyboard event, and Central Processing Unit (CPU) cycle event). The min-entropy contained in the combined seed value is 256 bits.

FCS_SSHS_EXT.1.1 – This SFR does not contain any NDcPP TSS Assurance Activities.

FCS_SSHS_EXT.1.2 – TD0631 – *"The evaluator shall check to ensure that the TSS contains a list of supported public key algorithms that are accepted for client authentication and that this list is consistent with signature verification algorithms selected in FCS_COP.1/SigGen (e.g., accepting EC keys requires corresponding Elliptic Curve Digital Signature algorithm claims).*

The evaluator shall confirm that the TSS includes the description of how the TOE establishes a user identity when an SSH client presents a public key or X.509v3 certificate. For example, the TOE could verify that the SSH client's presented public key matches one that is stored within the SSH server's authorized_keys file.

If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, then the evaluator shall confirm its role in the authentication process is described in the TSS."

Section 8.2.9 lists that ssh-rsa is the only public key algorithm supported. The use of RSA public keys for user authentication is consistent with FCS_COP.1/SigGen claim of being able to verify RSA keys.

If the SSH client's presented public key does not match a stored key on the TOE, the TOE will consider this a failed authentication attempt and the connection will not be established. In the case of password-based authentication attempt, the presented user credentials are verified using the TOE's native authentication mechanism. If the presented user credentials cannot be verified, then the connection will not be established.

FCS_SSHS_EXT.1.3 – *"The evaluator shall check that the TSS describes how "large packets" in terms of RFC 4253 are detected and handled."*

Section 8.2.9 of the TSS states that the SSH implementation detects packets greater than 32,768 bytes and when detected will be dropped.

FCS_SSHS_EXT.1.4 – *"The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that optional characteristics are specified, and the encryption algorithms supported are specified as well. The evaluator shall check the TSS to ensure that the encryption algorithms specified are identical to those listed for this component."*

Section 8.2.9 of the TSS correctly lists aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com as the supported encryption algorithms, which is consistent with the SFR definition and thus, the only option is to choose one of these encryption algorithms for transport implementation.

FCS_SSHS_EXT.1.5 – TD0631 – *"The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the SSH server's host public key algorithms supported are specified and that they are identical to those listed for this component."*

Section 8.2.9 lists that ssh-rsa is the only public key algorithm supported. The use of RSA for host public key is consistent with FCS_COP.1/SigGen claim of being able to verify RSA keys.

FCS_SSHS_EXT.1.6 – *"The evaluator shall check the TSS to ensure that it lists the supported data integrity algorithms, and that that list corresponds to the list in this component."*

Section 8.2.9 of the TSS correctly lists hmac-sha1, hmac-sha2-256, hmac-sha2-512, and implicit for data integrity algorithms which is consistent with the SFR definition.

FCS_SSHS_EXT.1.7 – *"The evaluator shall check the TSS to ensure that it lists the supported key exchange algorithms, and that that list corresponds to the list in this component."*

Section 8.2.9 of the TSS indicates that diffie-hellman-group14-sha1 is the only allowed key exchange method within the evaluated configuration which is consistent with the SFR definition.

FCS_SSHS_EXT.1.8 – *"The evaluator shall check that the TSS specifies the following:*

a) Both thresholds are checked by the TOE.

b) Rekeying is performed upon reaching the threshold that is hit first."

Section 8.2.9 of the TSS states that the TSF enforces the connection to be rekeyed after no longer than one hour, and no more than one gigabyte of transmitted data, whichever threshold is reached first. The SSH rekey time and size threshold parameters are administratively configurable via the CLI. One hour and one gigabyte are the maximum settings allowed for the rekey threshold parameters in the evaluated configuration.

FCS_TLSC_EXT.1.1 – *"The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified. The evaluator shall check the TSS to ensure that the ciphersuites specified include those listed for this component."*

Section 8.2.10 of the TSS states that the following ciphersuites are supported for TLS version 1.2:

- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

which are consistent with the SFR definition.

FCS_TLSC_EXT.1.2 – *"The evaluator shall ensure that the TSS describes the client's method of establishing all reference identifiers from the administrator/application-configured reference identifier, including which types of reference identifiers are supported (e.g. application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported.*

Note that where a TLS channel is being used between components of a distributed TOE for FPT_ITT.1, the requirements to have the reference identifier established by the user are relaxed and the identifier may also be established through a "Gatekeeper" discovery process. The TSS should describe the discovery process and highlight how the reference identifier is supplied to the "joining" component. Where the secure channel is being used between components of a distributed TOE for FPT_ITT.1 and the ST author selected attributes from RFC 5280, the evaluator shall ensure the TSS describes which attribute type, or combination of attributes types, are used by the client to match the presented identifier with the configured identifier. The evaluator shall ensure the TSS presents an argument how the attribute type, or combination of attribute types, uniquely identify the remote TOE component; and the evaluator shall verify the attribute type, or combination of attribute types, is sufficient to support unique identification of the maximum supported number of TOE components.

If IP addresses are supported in the CN as reference identifiers, the evaluator shall ensure that the TSS describes the TOE's conversion of the text representation of the IP address in the CN to a binary representation of the IP address in network byte order. The evaluator shall also ensure that the TSS describes whether canonical format (RFC 5952 for IPv6, RFC 3986 for IPv4) is enforced."

Section 8.2.10 of the TSS indicates that the Common Name and Subject Alternative Name (DNS Name only) are the only reference identifiers in the certificate that is part of that validation. The TOE will only support a wildcard in the left-most label (e.g. *.example.com). All other usages of a wildcard will cause a failure in the connection. The TOE does not support URI, IP addresses or service name reference identifiers or pinned certificates.

FCS_TLSC_EXT.1.3 – This SFR does not contain any NDcPP TSS Assurance Activities.

FCS_TLSC_EXT.1.4 – *"The evaluator shall verify that TSS describes the Supported Elliptic Curves/Supported Groups Extension and whether the required behaviour is performed by default or may be configured."*

Per Section 8.2.11, N/A – No ECDHE or DHE ciphers are used as part of the evaluated configuration.

FCS_TLSS_EXT.1.1 – *"The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified. The evaluator shall check the TSS to ensure that the ciphersuites specified are identical to those listed for this component."*

Section 8.2.11 of the TSS states that the following ciphersuites are supported for TLS version 1.2:

- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

which are consistent with the SFR definition.

FCS_TLSS_EXT.1.2 – *"The evaluator shall verify that the TSS contains a description of how the TOE technically prevents the use of old SSL and TLS versions."*

Section 8.2.11 of the TSS states that all connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 are denied. When the TOE receives a TLS connection request with the wrong (unsupported) version, it returns a Fatal Alert: Handshake failure message and terminates the connection.

FCS_TLSS_EXT.1.3 – *"If using ECDHE or DHE ciphers, the evaluator shall verify that the TSS describes the key agreement parameters of the server Key Exchange message.*

Per Section 8.2.11, N/A – No ECDHE or DHE ciphers are used as part of the evaluated configuration.

FCS_TLSS_EXT.1.4 – TD0569 *"The evaluator shall verify that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246) and/or if session resumption based on session tickets is supported (RFC 5077).*

If session tickets are supported, the evaluator shall verify that the TSS describes that the session tickets are encrypted using symmetric algorithms consistent with FCS_COP.1/DataEncryption. The evaluator shall verify that the TSS identifies the key lengths and algorithms used to protect session tickets.

If session tickets are supported, the evaluator shall verify that the TSS describes that session tickets adhere to the structural format provided in section 4 of RFC 5077 and if not, a justification shall be given of the actual session ticket format.

If the TOE claims a (D)TLS server capable of session resumption (as a single context, or across multiple contexts), the evaluator verifies that the TSS describes how session resumption operates (i.e. what would trigger a full handshake, e.g. checking session status, checking Session ID, etc.). If multiple contexts are used the TSS describes how session resumption is coordinated across those contexts. In case session establishment and session resumption are always using a separate context, the TSS shall describe how the contexts interact with respect to session resumption (in particular regarding the session ID). It is acceptable for sessions established in one context to be resumable in another context."

Section 8.2.11 of the TSS states that session resumption is not supported.

FIA_AFL.1 – *"The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.*

The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking)."

Section 8.3.1 of the TSS states that the TSF uses a configurable counter to track consecutive failed authentication attempts, and that a user account will be locked out when the counter reaches its defined threshold. A valid login that happens prior to the failure counter reaching its threshold will reset the counter to zero.

For Console user accounts that are locked: A user with a locked account cannot login into the Console application until another Console Security Administrator manually unlocks the account via the Console. A locked Console user account can be manually unlocked by a Console Security Administrator by navigating to the "Tools" > "Options" > "CounterACT User Profiles" page in the Console, selecting the locked user account, and pressing the activated "Unlock" button. Additionally, a CLI Security Administrator may unlock a Console user account using the "fstool unlock_console_user <user-id>" command.

For CLI user accounts that are locked: A user with a locked account cannot login to either the remote CLI or local console until a CLI Security Administrator manually unlocks the account using the "fstool user faillock reset <locked username>" command or when the CLI configured time limit set by a CLI Security Administrator has elapsed. A CLI user account cannot be unlocked via the Console.

Multiple Console and CLI Security Administrator accounts are required to prevent complete user lockout. During installation and configuration of the TOE, the "Admin" user must create at least one new Console Security Administrator account and the "cliadmin" user must be used to create at least one new CLI Security Administrator account. These new Security Administrator accounts will provide the ability to unlock accounts that have been locked due to reaching the failed number of authentication attempts threshold.

FIA_PMG_EXT.1 – "The evaluator shall examine the TSS to determine that it contains the lists of the supported special character(s) and minimum and maximum number of charters supported for administrator passwords."

Section 8.3.2 of the TSS lists the set of supported special characters of "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")" and a minimum password length ranging from 15 to 30 characters supported for administrator passwords.

FIA_UAU.7 – This SFR does not contain any NDcPP TSS Assurance Activities.

FIA_UAU_EXT.2 – *"Evaluation Activities for this requirement are covered under those for FIA_UIA_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA_UIA_EXT.1."*

FIA_UIA_EXT.1 – *"The evaluator shall examine the TSS to determine that it describes the logon process for each logon method (local, remote (HTTPS, SSH, etc.)) supported for the product. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a "successful logon".*

The evaluator shall examine the TSS to determine that it describes which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration.

For distributed TOEs the evaluator shall examine that the TSS details how Security Administrators are authenticated and identified by all TOE components. If not, all TOE components support authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2, the TSS shall describe how the overall TOE functionality is split between TOE components including how it is ensured that no unauthorized access to any TOE component can occur.

For distributed TOEs, the evaluator shall examine the TSS to determine that it describes for each TOE component which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration. For each TOE component that does not support authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2 the TSS shall describe any unauthenticated services/services that are supported by the component."

Section 8.3.4 of the TSS states that users are authenticated on the local console (physical) and remote CLI (SSH) using a native username/password. SSH connections also support public key-based authentication. When authenticating via the Console application, the TOE can be configured to request an authentication decision from an Active Directory server or use the native username/password credential authentication mechanism. Access is only granted once the user provides a valid username/password that is verified using Active Directory or native username/password credential authentication mechanism.

Section 8.3.4 of the TSS states that the display and acknowledgement of a warning banner is the only TOE functionality available prior to identification and authentication.

FIA_X509_EXT.1/Rev – *"The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). It is expected that revocation checking is performed when a certificate is used in an authentication step and when performing trusted updates (if selected). It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected).*

The TSS shall describe when revocation checking is performed and on what certificates. If the revocation checking during authentication is handled differently depending on whether a full certificate chain or only a leaf certificate is being presented, any differences must be summarized in the TSS section and explained in the Guidance."

Section 8.3.5 of the TSS states that X.509v3 certificates are used to support authentication to external IT entities that use TLS. The certificate path is validated from the root CA when the CA Certificate response is received. Additionally, the TSF validates the certificate revocation status using the Online Certificate Status Protocol (OCSP) as specified in RFC 6960. This includes the leaf certificate and all intermediate certificates received. The certificate path is validated by ensuring that all the CA certificates have the basicConstraints extension and the CA flag set to TRUE, and the certificate path must terminate with a trusted CA certificate. The extendedKeyUsage field is validated according to the following rules:

- Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
- Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
- Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
- OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

FIA_X509_EXT.2 – *"The evaluator shall check the TSS to ensure that it describes how the TOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.*

The evaluator shall examine the TSS to confirm that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The evaluator shall verify that any distinctions between trusted channels are described. If the requirement that the administrator is able to specify the default action, then the evaluator shall ensure that the guidance documentation contains instructions on how this configuration action is performed."

Section 8.3.5 of the TSS states that X.509v3 certificates are used for TLS authentication. The TSS goes on to state that for the TOE to have its own certificate, a Certificate Request is generated as specified in RFC 2986 containing the public key and "Common Name". For authenticating to the audit and Active Directory servers, trusted certificates must be installed into the TOE's certificate trust store.

Section 8.3.5 of the TSS states that when the TSF cannot determine the validity of a certificate, the TSF will not accept the certificate and not establish a connection. The TSF does not provide a mechanism to override the validation decision.

FIA_X509_EXT.3 – *"If the ST author selects "device-specific information", the evaluator shall verify that the TSS contains a description of the device-specific fields used in certificate requests."*

N/A - The ST author did not select "device-specific information".

FMT_MOF.1/ManualUpdate – "For distributed TOEs see chapter 2.4.1.1. There are no specific requirements for non-distributed TOEs."

FMT_MTD.1/CoreData – *"The evaluator shall examine the TSS to determine that, for each administrative function identified in the guidance documentation; those that are accessible through an interface prior to administrator log-in are identified. For each of these functions, the evaluator shall also confirm that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users.*

If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator shall examine the TSS to determine that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted."

Section 8.4.1 of the TSS describes that the TOE only allows a user to view a warning banner prior to any authentication. The TSS also states that the TOE uses role-based access control (RBAC) to restrict access to the functions that manage TSF data. The available functionality that is presented to an authenticated user is based on the group of permissions and the privileges associated with the permissions. RBAC restriction is applicable to all of the following NDcPP scoped functionality:

- Configure Banner Text
- Configure Idle Session Timeout
- Initiate Manual Update
- Configure Failed Lockout Threshold
- Configure Lockout Duration
- Configure audit server information for audit data transmission
- Configure thresholds for SSH rekeying
- Re-enable Administrator accounts
- Configure System Time
- Manage trusted public keys database
- Manage the TOE's trust store and designate X.509v3 certificates as trust anchors

FMT_SMF.1 – *"The evaluator shall examine the TSS, Guidance Documentation and the TOE as observed during all other testing and shall confirm that the management functions specified in FMT_SMF.1 are provided by the TOE. The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface).*

The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface. The evaluator shall ensure the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.

For distributed TOEs with the option 'ability to configure the interaction between TOE components' the evaluator shall examine that the ways to configure the interaction between TOE components is detailed in the TSS and Guidance Documentation. The evaluator shall check that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS and Guidance Documentation."

Section 8.4.1 of the TSS describes the PP-scoped administrative actions based on the administrative interfaces. The table provided, clearly indicates local CLI is a physical connection vs the remote CLI being SSH and remote Console using TLS. This is consistent throughout the ST and AGD. The AGD instructions include differentiators as to whether the administrator should use the Console application or the CLI interface. The AGD also includes specifics on how a setting affects local CLI, remote CLI, or remote Console.

FMT_SMR.2 – *"The evaluator shall examine the TSS to determine that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE."*

Section 8.4.2 of the TSS details the TOE supported roles and any restrictions of the roles involving administration of the TOE by enforcing role-based access control (RBAC) to limit access to TSF functions and data based on the set of permissions bound to the subject. The TOE has one predefined Console administrative user called "Admin". The "Admin" account is assigned the "administrator" role and these permissions cannot be modified or customized. A customized password must be created during installation by the customer. The "Admin" account is used to create additional Console Security Administrators.

A Console Security Administrator must assign permissions when creating an additional Console user. These permissions may be modified later by a Console Security Administrator or Console user with the correct permissions. A Console User's set of permissions are customized by adding and subtracting specific permissions to allow/disallow the user TOE functionality. To create an additional Console Security Administrator, all the permissions must be selected and assigned to the user.

Additionally, the TOE has one predefined CLI administrative role called "cliadmin". CLI roles and permissions cannot be modified or customized at any time. A customized password must be created during installation by the customer. The "cliadmin" account is used to create additional CLI Security Administrators.

FPT_APW_EXT.1 – *"The evaluator shall examine the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS shall also detail passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note."*

Section 8.5.1 of the TSS states that Console passwords are hashed using SHA-256 and then encrypted using AES-256, and that CLI passwords are hashed using SHA-512. The TOE provides no function or mechanism to view plaintext password data and the password data is not recoverable.

FPT_SKP_EXT.1 – *"The evaluator shall examine the TSS to determine that it details how any pre-shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS shall describe how they are protected/obscured."*

Section 8.5.2 of the TSS states that the TOE does not provide means to view pre-shared, symmetric, or private keys. Volatile memory used to store secret keys, private keys, and secret key data is not accessible by administrators and neither is the file system of the OS. The TSS also states that data keys stored on the TOE are encrypted using AES-256. There are no keys stored in plaintext.

FPT_STM_EXT.1 – *"The evaluator shall examine the TSS to ensure that it lists each security*

function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions."

Section 8.5.3 of the TSS states that time is maintained using its own internal clock that can be set manually by an CLI Security Administrator. The date and time are used by the TOE for audit record timestamps, tracking the inactivity of administrative sessions (local and remote), and checking the validation of X.509v3 certificates.

FPT_TST_EXT.1 – *"The evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" shall be used). The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.*

For distributed TOEs the evaluator shall examine the TSS to ensure that it details which TOE component performs which self-tests and when these self-tests are run."

Section 8.5.4 of the TSS states that upon startup of the TOE, multiple Power-On Self Tests (POSTs) are run. These tests provide environmental monitoring of the hardware and software. Details of the tests are located in Table 24 of the ST which shows the component, validation, and a fail result. Each fail result is described in detail and the reaction of the TOE if this occurs. Self-tests also include verification of the integrity check of the software and cryptographic modules. The self-tests will also be run on service restarts and are available for manual execution by a CLI Security Administrator. Additionally, this section states, "These tests are sufficient to validate the correct operation of the TSF because they verify that the software has not been tampered with and that the underlying hardware does not have any anomalies that would cause the software to be executed in an unpredictable or inconsistent manner."

FPT_TUD_EXT.1 – *"The evaluator shall verify that the TSS describe how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS needs to describe how and when the inactive version becomes active. The evaluator shall verify this description.*

The evaluator shall verify that the TSS describes all TSF software update mechanisms for updating the system firmware and software (for simplicity the term 'software' will be used in the following although the requirements apply to firmware and software). The evaluator shall verify that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS shall detail this mechanism instead of the digital signature verification mechanism. The evaluator shall verify that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.

If the options 'support automatic checking for updates' or 'support automatic updates' are chosen from the selection in FPT_TUD_EXT.1.2, the evaluator shall verify that the TSS explains what actions are involved in automatic checking or automatic updating by the TOE, respectively.

For distributed TOEs, the evaluator shall examine the TSS to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component. Alternatively, this description can be provided in the guidance documentation. In that case the evaluator should examine the guidance documentation instead.

If a published hash is used to protect the trusted update mechanism, then the evaluator shall verify that the trusted update mechanism does involve an active authorization step of the Security Administrator, and that

download of the published hash value, hash comparison and update is not a fully automated process involving no active authorization by the Security Administrator. In particular, authentication as Security Administration according to FMT_MOF.1/ManualUpdate needs to be part of the update process when using published hashes."

Section 8.5.5 of the TSS states that a Console Security Administrator can query the TOE for the currently executing version of the TOE software by going to the top menu bar, click the "Help" drop down menu, and then click "About Forescout".

Section 8.5.5 of the TSS states that the TOE does not automatically check for or download an update itself nor does it connect to the update server directly. When an update is available, a Security Administrator must download the update package to the management workstation. Once the update is on the management workstation an administrator must manually initiate the installation via the Console.

For an Appliance Model: Upon execution of the upgrade command the Console Security Administrator has the choice of the following option:

- Upload and Upgrade - Upload the file to the device and begin the upgrade.

For an Enterprise Manager Model: Upon execution of the upgrade command the Console Security administrator has the choice of the following three options:

- Upload Only – Upload the file to the device but do not begin the upgrade
- Upload and Upgrade – Upload the file to the device and begin the upgrade
- Upgrade – Upgrade the device from the previously uploaded file.

The Console uploads the update package over the existing TLS path that is already established between the Console and the TOE appliance. Only one upgrade package can be uploaded to the TOE device at a time. A second attempt to upload an upgrade package will result in the administrator being warned that this will overwriting the existing upgrade package. The following provide more details to each of the installation options identified above:

> Upload Only – The TSF automatically verifies the update's digital signature during the upload process. The TSF uses a locally stored public key (on the appliance) to verify update package authenticity. This key is installed as part of the initial software installation and cannot be modified or changed by an administrator. The TSF will delete the uploaded file if the digital signature is determined to be invalid for any reason. There is no means for an administrative override to continue the upload. Once the upload is complete and the digital signature is valid, the Console indicates its success.
>
> Upgrade Only – When there is an upgraded package available, a Console Security Administrator can select the Upgrade Only option to initiate the installation. The TOE will re-verify the digital signature prior to initiating the installation. The TSF will not continue with the installation if the digital signature is determined to be invalid for any reason. There is no means for an administrative override to continue the installation. Once the device has been upgraded, the device will reboot automatically, the upload storage area is emptied (meaning another upgrade package can now be uploaded but not installed), and the current operating version will be updated to reflect the recent upgrade version.
>
> Upload and Upgrade – The TSF automatically verifies the update's digital signature during the upload process. Once the upload is complete and the digital signature is valid, the installation will begin. The TOE will re-verify the digital signature prior to initiate the installation. The TSF will not continue with the installation if the digital signature is determined to be invalid for any reason. There is no means for an administrative override to continue the installation. Once the device has been upgraded, the device will reboot automatically, the upload storage area is emptied (meaning another upgrade package can now be uploaded but not installed), and the current operating version will be updated to reflect the recent upgrade version.

The TOE does not claim *'support automatic checking for updates'* or *'support automatic updates'*.

The TOE does not claim the use of published hashes.

FTA_SSL_EXT.1 – *"The evaluator shall examine the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings."*

Section 8.6.1 of the TSS states when a local session is inactive for the configured period of time, the TOE will terminate the session. The inactivity timer is configured by a Console Security Administrator via the Console and is set in minutes or hours.

FTA_SSL.3 – *"The evaluator shall examine the TSS to determine that it details the administrative remote session termination and the related inactivity time period."*

Section 8.6.2 of the TSS states the TOE will terminate a remote session due to inactivity according to the configuration threshold set by a Console Security Administrator. The inactivity timer is configured by a Console Security Administrator via the Console and is set in minutes or hours.
.

FTA_SSL.4 – *"The evaluator shall examine the TSS to determine that it details how the local and remote administrative sessions are terminated."*

Section 8.6.3 of the TSS states any user accessing the TOE is capable of terminating their own session. A Console user terminates their own current session by clicking the "exit" command from the File menu. A CLI user terminates their own current session by typing "quit" at the command line.

FTA_TAB.1 – *"The evaluator shall check the TSS to ensure that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS). The evaluator shall check the TSS to ensure that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file)."*

Section 8.6.4 of the TSS states there are three possible administrative ways to log into the TOE: locally via physical connection to access the CLI, remotely via SSH connection to access the CLI, and remotely using the Console which establishes a TLS connection. When logging in locally or remotely, the pre-authentication banner is displayed and is viewed prior to authentication. The authentication banner is administratively customizable by a Console Security Administrator via the Console.

FTP_ITC.1 – *"The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. The evaluator shall also confirm that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST."*

Section 8.7.1 of the TSS states the TOE provides the ability to secure data in transit to and from the Operational Environment using TLS. The TOE acts as a TLS client to support the following capabilities:

- to export audit data to an audit server
- authenticate users via an Active Directory server

Section 8.7.1 of the TSS also states, "The TOE appliance's TLS client implementation is conformant to FCS_TLSC_EXT.1. TLS communications use X5.09v3 certificates to support authentication."

This protocol is consistent with the claims made in the ST.

FTP_TRP.1/Admin – *"The evaluator shall examine the TSS to determine that the methods of remote TOE administration are indicated, along with how those communications are protected. The evaluator shall also confirm that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST."*

Section 8.7.2 of the TSS states that remote administration is secured by using SSH and TLS protocols.

- The Console establishes the TLS connection to the TOE appliance on behalf of the user for remote administration. The TOE appliance is acting as a TLS Server and is conformant to FCS_TLSS_EXT.1. The Console is using the host platforms TLS client capabilities.

- A user can connect to the TOE appliance using SSH to remotely manage the TOE appliance via the CLI (remote console). The TOE appliance's SSH server implementation is conformant to FCS_SSHS_EXT.1.

3 Operational Guidance Assurance Activities

The evaluation team completed the testing of the Operational Guidance, which includes the review of the *Forescout v8.3 Supplemental Administrative Guidance* (AGD) document, and confirmed that the Operational Guidance contains all Assurance Activities as specified by the 'Collaborative Protection Profile for Network Devices, version 2.2e (NDcPP)'. The evaluators reviewed the NDcPP to identify the security functionality that must be discussed for the operational guidance. This is prescribed by the Assurance Activities for each SFR and the AGD SARs. The evaluators have listed below each of the SFRs defined in the NDcPP that have been claimed by the TOE (some SFRs are conditional or optional) as well as the AGD SAR, along with a discussion of where in the operational guidance the associated Assurance Activities material can be found.

FAU_GEN.1 – *"The evaluator shall check the guidance documentation and ensure that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, shall be provided from the actual audit record).*

The evaluator shall also make a determination of the administrative actions related to TSF data related to configuration changes. The evaluator shall examine the guidance documentation and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP. The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding guidance documentation satisfies the requirements related to it."

Section 8 of the AGD provides a table of auditable events that is consistent with the auditable events table in the NDcPP for the claimed SFRs. This table includes examples of audit records for different situations that are associated with the requirement including all audit events defined in Table 1 of the NDcPP. Section 8 provides an example of an audit record before this table and breaks it down into the individual fields that are prescribed by FAU_GEN.1.2. From this example, the relationship between the audit logs shown in the table and the required fields can be determined clearly.

"*The evaluator shall also make a determination of the administrative actions related to TSF data related to configuration changes. The evaluator shall examine the guidance documentation and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP. The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding guidance documentation satisfies the requirements related to it."*

The AGD was developed with the intent to provide the specific guidance for managing TOE functionality or a pointer to the necessary documentation as defined by the Intended Audience statement in Section 2: "This document is intended for administrators responsible for installing, configuring, and/or operating Forescout. Guidance provided in this document allows the reader to deploy the product in an environment that is consistent with the configuration that was evaluated as part of the product's Common Criteria (CC) testing process. It also provides the reader with instructions on how to exercise the security functions that were claimed as part of the CC evaluation. The reader is also expected to be familiar with the general operation of the Forescout product. This supplemental guidance includes references to Forescout's standard documentation set for the product and does not explicitly reproduce materials located there. The reader is also expected to be familiar with the Forescout v8.3 Security Target and the general CC terminology that is referenced in it. This document references the Security Functional Requirements (SFRs) that are defined in

the Security Target document and provides instructions for how to perform the security functions that are defined by these SFRs. The Forescout product as a whole provides a great deal of security functionality but only those functions that were in the scope of the claimed PP are discussed here. Any functionality that is not described here or in the Forescout Security Target was not evaluated and should be exercised at the user's risk." Thus, the evaluation team has determined that only the commands located within the AGD and the specific pointers to other documents are considered to be security relevant for this evaluation. Therefore, through the course of the remainder of the operational guidance assurance activities the evaluation team confirmed that all portions of the AGD with administrative commands were for the initial setup and configuration of the TOE, or directly in the support of a Common Criteria requirement. As part of the testing effort the evaluation team also cross compared the commands needed to execute the test cases with the commands defined in the AGD. Through this review, the evaluation team determined that since the AGD documents and/or provides the necessary pointer for all security relevant commands that were executed by the evaluation team in performing the independent testing, that the subset of the commands defined or referenced to in the AGD are all of the security relevant commands necessary to enforce the SFRs specified in the PP.

FAU_GEN.2 – "*The TSS and Guidance Documentation requirements for FAU_GEN.2 are already covered by the TSS and Guidance Documentation requirements for FAU_GEN.1."*

See FAU_GEN.1.

FAU_STG_EXT.1 – "*The evaluator shall also examine the guidance documentation to ensure it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.*

Section 6.7 of the AGD describes how to establish the trusted channel from the TOE to the external audit server (syslog server). This section also includes the protocol, version of the protocol and the configuration steps on the TOE to communicate with the syslog server.

"The evaluator shall also examine the guidance documentation to determine that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. For example, when an audit event is generated, is it simultaneously sent to the external server and the local store, or is the local store used as a buffer and "cleared" periodically by sending the data to the audit server."

Section 6.7 of the AGD states that all audit records are stored locally and automatically sent to the remote syslog server as soon as they are generated.

"The evaluator shall also ensure that the guidance documentation describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behaviour of the TOE for each possible configuration. The description of possible configuration options and resulting behaviour shall correspond to those described in the TSS."

Section 8.1 of the AGD describes the behavior of the local audit storage once the storage is full. The application layer events and the OS log files are described separately because they are stored in different locations on the TOE. The behavior of the audit storage is different between the two and both are described in detail. These descriptions are consistent with TSS section 8.1.2 of the Forescout Security Target.

FCS_CKM.1 - *"The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_CKM.2 – *"The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s)."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_CKM.4 – "*A TOE may be subject to situations that could prevent or delay key destruction in some cases. The evaluator shall check that the guidance documentation identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used). The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer.*

For example, when the TOE does not have full access to the physical memory, it is possible that the storage may be implementing wear-levelling and garbage collection. This may result in additional copies of the key that are logically inaccessible but persist physically. Where available, the TOE might then describe use of the TRIM command3 and garbage collection to destroy these persistent copies upon their deletion (this would be explained in TSS and Operational Guidance)."

Section 6.9 states that there are no known instances where key destruction does not happen as defined by the Security Target. This is consistent with TSS description.

FCS_COP.1/DataEncryption – *"The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_COP.1/SigGen – *"The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_COP.1/Hash - *"The evaluator checks the AGD documents to determine that any configuration that is required to configure the required hash sizes is present."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_COP.1/KeyedHash – *"The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_RBG_EXT.1 – *"The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

FCS_SSHS_EXT.1.1 – This SFR does not contain any NDcPP AGD Assurance Activities.

FCS_SSHS_EXT.1.2 – This SFR does not contain any NDcPP AGD Assurance Activities.

FCS_SSHS_EXT.1.3 – This SFR does not contain any NDcPP AGD Assurance Activities.

FCS_SSHS_EXT.1.4 – *"The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements)."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. This limits the SSH encryption algorithms, public key algorithms, MAC algorithms and key exchange methods to only what is being claimed in the ST. Section 6.5 details a list of all of the SSH algorithms and it is consistent with the claims in the ST.

FCS_SSHS_EXT.1.5 - *"The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements)."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. This limits the SSH encryption algorithms, public key algorithms, MAC algorithms and key exchange methods to only what is being claimed in the ST. Section 6.5 details a list of all of the SSH algorithms and it is consistent with the claims in the ST.

FCS_SSHS_EXT.1.6 – *"The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed data integrity algorithms are used in SSH connections with the TOE (specifically, that the "none" MAC algorithm is not allowed)."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. This limits the SSH encryption algorithms, public key algorithms, MAC algorithms and key exchange methods to only what is being claimed in the ST. Section 6.5 details a list of all of the SSH algorithms and it is consistent with the claims in the ST. Section 6.5 also states the following, "NOTE: The MAC algorithms defined above are the only ones included in the evaluated configuration and thus, the "none" MAC algorithm is never allowed for SSH."

FCS_SSHS_EXT.1.7 – *"The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed key exchange algorithms are used in SSH connections with the TOE."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. This limits the SSH encryption algorithms, public key algorithms, MAC algorithms and key exchange methods to only what is being claimed in the ST. Section 6.5 details a list of all of the SSH algorithms and it is consistent with the claims in the ST.

FCS_SSHS_EXT.1.8 – *"If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, then the evaluator shall check that the guidance documentation describes how to configure those thresholds. Either the allowed values are specified in the guidance documentation and must not exceed the limits specified in the SFR (one hour of session time, one gigabyte of transmitted traffic) or the TOE must not accept values beyond the limits specified in the SFR. The evaluator shall check that the guidance documentation describes that the TOE reacts to the first threshold reached."*

Section 6.5 if the AGD states "SSH session key thresholds for time and amount of transmitted data can be configured by a CLI Security Administrator executing the following steps:

1. Authenticate to the TOE via the CLI using a Security Administrator account.
2. Execute the following command: ssh -t server -o RekeyLimit "1G 1h

This settings shown in this command are the maximum and default values for the evaluated configuration. In this configuration, the TOE has been configured to rekey when one hour has elapsed or one gigabyte of data has been transmitted using a key; whichever occurs first. The first argument in the quotation marks specifies the maximum amount of data that may be transmitted before the session key is renegotiated. This value is defined in bytes and may have a suffix of 'K', 'M', or 'G' to indicate Kilobytes, Megabytes, or Gigabytes, respectively. The second argument in the quotation marks specifies the maximum amount of time that may pass before the session key is renegotiated. This value is defined with a suffix of 's', 'm', or 'h' to indicate seconds, minutes, or hour respectively."

FCS_TLSC_EXT.1.1 – *"The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. Section 6.7 and 6.8 lists each of the ciphersuites supported by the TOE while in FIPS mode as well as the TLS protocol version. This list is consistent with the description in the TSS.

FCS_TLSC_EXT.1.2 – *"The evaluator shall ensure that the operational guidance describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by the TOE includes support for IP addresses, the evaluator shall ensure that the operational guidance provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.*

Where the secure channel is being used between components of a distributed TOE for FPT_ITT.1, the SFR selects attributes from RFC 5280, and FCO_CPC_EXT.1.2 selects "no channel"; the evaluator shall verify the guidance provides instructions for establishing unique reference identifiers based on RFC5280 attributes."

Sections 6.7 and 6.8 contain the instructions for setting the reference identifier for certification validation when using TLS for both Syslog and Active Directory. These instructions include steps for connecting to Active Directory servers as well as syslog servers. Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.

Section 6.6 of the AGD provides information and configuration steps for Certificate Management. It contains information on creating the server certificate and the mandatory fields CN= fully qualified domain name (FQDN), Key length=2048, and signature algorithm= SHA-256. For certificates that need to be imported to handle audit server and authentication server connections it clearly indicates. The Common Name and Subject Alternative Name (FQDN) are the only reference identifiers in the certificate that is part of that validation. The TOE will only support a wildcard in the left-most label (e.g. *.example.com). All other usages of a wildcard will cause a failure in the connection. The TOE does not support URI, IP addresses or service name reference identifiers or pinned certificates.

FCS_TLSC_EXT.1.3 – This SFR does not contain any NDcPP AGD Assurance Activities.

FCS_TLSC_EXT.1.4 – *"If the TSS indicates that the Supported Elliptic Curves/Supported Groups Extension must be configured to meet the requirement, the evaluator shall verify that AGD guidance includes configuration of the Supported Elliptic Curves/Supported Groups Extension."*

The TOE does not support Elliptic Curve Extensions. Therefore, this Assurance Activity is not applicable.

FCS_TLSS_EXT.1.1 – *"The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements)."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. Section 6.2 explains that once the TOE has been installed and placed into FIPS mode the TLS server functions are automatic and will only accept TLS v1.2 and the 4 ciphers declared in the ST.

FCS_TLSS_EXT.1.2 – *"The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. Section 6.2 explains that once the TOE has been installed and placed into FIPS mode the TLS server functions are automatic and will only accept TLS v1.2 and the 4 ciphers declared in the ST.

FCS_TLSS_EXT.1.3 – *"The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations. Section 6.2 explains that once the TOE has been installed and placed into FIPS mode the TLS server functions are automatic and will only accept TLS v1.2 and the 4 ciphers declared in the ST.

FCS_TLSS_EXT.1.4 – TD0569 "*The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance."*

Section 6.2 of the AGD states "The TOE does not claim support session resumption or session tickets." Therefore, this Assurance Activity is not applicable.

FIA_AFL.1 – *"The evaluator shall examine the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described.*

The evaluator shall examine the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1."

Section 7.2 of the AGD describes the actions that the TOE enforces for failed login attempts. The AGD includes instructions for configuring the TOE's authentication lockout mechanism attributes for both Console and CLI users. This section gives step by step instructions in order to set the maximum failed attempts and the lockout time before the user is able to authenticate again. This section also identifies the difference in enforcement activities based on the interfaces.

Multiple Console and CLI Security Administrator accounts are required to prevent complete user lockout. During installation and configuration of the TOE, the "Admin" user must create at least one new Console Security Administrator account and the "cliadmin" user must be used to create at least one new CLI Security Administrator account. These new Security Administrator accounts will provide the ability to

unlock accounts that have been locked due to reaching the failed number of authentication attempts threshold. Section 7.2 provides instruction on unlocking accounts using the Console and CLI interfaces.

FIA_PMG_EXT.1 – *"The evaluator shall examine the guidance documentation to determine that it:*

a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and

b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported."

Section 7.4 of the AGD provides instruction on how to configure the TOE to prevent insecure passwords. The section describes setting the minimum length of passwords to 15 characters as well as provides composition guidance to ensure strong passwords are used through using a mixture of uppercase, lowercase, numeric, and special character sets. Section 7.4 also provides a list of special characters that are supported by the TOE for a secure password.

FIA_UAU.7 – *"The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed."*

Section 7.1.2 of the AGD states that the TOE does not echo progress when entering a password for a local connection. Suppression of password entry is an automatic behavior that is not configured nor can it be modified.

FIA_UIA_EXT.1 – "*The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps (e.g., establishing credential material such as preshared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator shall ensure the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator shall determine that the guidance documentation provides sufficient instruction on limiting the allowed services."*

Section 6.5 of the AGD describes the preparatory steps to configure SSH public-key authentication for users authenticating to the remote CLI.
Section 6.8 describes the necessary steps to configure the TOE to communicate with an Active Directory server for Console authentication.
Section 7.1.1 and 7.1.2 describes how to authenticate to the TOE using the Console and the Remote CLI.
Section 7.5 provides instructions for setting the warning banner which is the only pre-authentication function available.

FIA_UAU_EXT.2 – "*Evaluation Activities for this requirement are covered under those for FIA_UIA_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA_UIA_EXT.1."*

See FIA_UIA_EXT.1.

FIA_X509_EXT.1/Rev – *"The evaluator shall also ensure that the guidance documentation describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate."*

Section 6.6 of the AGD states the TOE performs certificate validity checking for all outbound TLS connections as part of the connection process. Successful certificate validation is required in order to successfully negotiate a connection. This sections further describes all of the fields and extensions that are checked, their expected values in order to successfully validate the certificate, and revocation checking via

OCSP as specified in RFC 6960. When the TSF cannot determine the validity of a certificate, the TSF will not accept the certificate and not establish a connection. The TSF does not provide a mechanism to override the validation decision. The TOE does not validate the entries of any other fields or extensions not described above and would therefore be considered trivially satisfied as part of the X.509 certificate validation.

FIA_X509_EXT.2 – *"The evaluator shall also ensure that the guidance documentation describes the configuration required in the operating environment so the TOE can use the certificates. The guidance documentation shall also include any required configuration on the TOE to use the certificates. The guidance document shall also describe the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel."*

Section 6.6 of the AGD describes how to import certificates for the purpose of validating presented server certificates from the Active Directory and audit servers. The description also describes that the CN and SAN are the only reference identifiers that as part of the validation. Additional information is provided that states the TOE will only support a wildcard in the left-most label (e.g. *.example.com). All other usages of a wildcard will cause a failure in the connection. The TOE does not support URI, IP addresses, service name reference identifiers, or pinned certificates. Additionally, when the TSF cannot determine the validity of a certificate, the TSF will not accept the certificate and not establish a connection. The TSF does not provide a mechanism to override the validation decision.

FIA_X509_EXT.3 – *"The evaluator shall check to ensure that the guidance documentation contains instructions on requesting certificates from a CA, including generation of a Certificate Request. If the ST author selects "Common Name", "Organization", "Organizational Unit", or "Country", the evaluator shall ensure that this guidance includes instructions for establishing these fields before creating the Certification Request."*

Section 6.6 of the AGD contains instructions for generating a certificate signing request and includes instructions on completing the fields of the request, specifically the "Common Name" – fully qualified domain name (FQDN) only, Key Length – RSA 2048, and Signature algorithm – SHA-256.

FMT_MOF.1/ManualUpdate – *"The evaluator shall examine the guidance documentation to determine that any necessary steps to perform manual update are described. The guidance documentation shall also provide warnings regarding functions that may cease to operate during the update (if applicable).*

For distributed TOEs the guidance documentation shall describe all steps how to update all TOE components. This shall contain description of the order in which components need to be updated if the order is relevant to the update process. The guidance documentation shall also provide warnings regarding functions of TOE components and the overall TOE that may cease to operate during the update (if applicable)."

Section 6.3 of the AGD provides step by step instructions for manually updating the TOE. The TOE will normally reboot after an upgrade (thus making the TOE unavailable for a few minutes). It also instructs the reader to refer to the Forescout Administration Guide.

FMT_MTD.1/CoreData – *"The evaluator shall review the guidance documentation to determine that each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.*

If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates,

the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator shall also review the guidance documentation to determine that it explains how to designate a CA certificate a trust anchor."

The AGD was developed with the intent to provide the specific guidance for managing TOE functionality or a pointer to the necessary documentation as defined by the Intended Audience statement in Section 2: "This document is intended for administrators responsible for installing, configuring, and/or operating Forescout. Guidance provided in this document allows the reader to deploy the product in an environment that is consistent with the configuration that was evaluated as part of the product's Common Criteria (CC) testing process. It also provides the reader with instructions on how to exercise the security functions that were claimed as part of the CC evaluation. The reader is expected to be familiar with the general operation of the Forescout product. This supplemental guidance includes references to Forescout's standard documentation set for the product and does not explicitly reproduce materials located there.
The reader is also expected to be familiar with the Forescout Security Target and the general CC terminology that is referenced in it. This document references the Security Functional Requirements (SFRs) that are defined in the Security Target document and provides instructions for how to perform the security functions that are defined by these SFRs. The Forescout product as a whole provides a great deal of security functionality but only those functions that were in the scope of the claimed PP are discussed here. Any functionality that is not described here or in the Forescout Security Target was not evaluated and should be exercised at the user's risk." Thus, the evaluation team has determined that only the commands located within the AGD and the specific pointers to other documents are considered to be security relevant for this evaluation. Therefore, through the course of the remainder of the operational guidance assurance activities the evaluation team confirmed that all portions of the AGD with administrative commands were for the initial setup and configuration of the TOE, or directly in the support of a Common Criteria requirement. As part of the testing effort the evaluation team also cross compared the commands needed to execute the test cases with the commands defined in the AGD. Through this review, the evaluation team determined that since the AGD documents and/or provides the necessary pointer for all security relevant commands that were executed by the evaluation team in performing the independent testing, that the subset of the commands defined or referenced to in the AGD are all of the security relevant commands necessary to enforce the SFRs specified in the PP.

Section 7.3 of the AGD describes how the TOE can be configured to limit privileges of different users within the constraints of the Security Administrator role. As shown in the other Guidance Assurance Activities, configuration and secure usage of the TSF's cryptographic, I&A, management, and update functions are discussed in various places throughout the AGD document (or in other administrative documents that are referenced by the AGD). The privilege levels and guidance surrounding their use in enforcing strict separation of duties is considered to be secure administration of the user privilege level parameters and would ensure that only Security Administrators have access to the allowed TOE functions.

Section 6.6 of the AGD goes into detail on how to manage X.509 certificates. This section identifies how to generate the TOE server certificate, import certificates for use with operational entities such as the syslog server and AD server, and how to designate as a trusted certificate. These steps were followed for testing and were found to be complete. This section also details the validation requirements for a X.509 certificate being presented to the TOE for trusted communications to occur.

FMT_SMF.1 *"The evaluator shall examine the TSS, Guidance Documentation and the TOE as observed during all other testing and shall confirm that the management functions specified in FMT_SMF.1 are provided by the TOE. The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface).*

The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface. The evaluator shall ensure the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.

For distributed TOEs with the option 'ability to configure the interaction between TOE components' the evaluator shall examine that the ways to configure the interaction between TOE components is detailed in the TSS and Guidance Documentation. The evaluator shall check that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS and Guidance Documentation."

The TSS specifies in Section 8.4.1 FMT_SMF.1 the management functions and identifies which management functions are available through the local CLI, remote CLI and Console interfaces. The ST defined management functions align with those discovered in the AGD (per the Section defined below) and performed during testing.

- Ability to administer the TOE locally (local CLI) and remotely (Console and remote CLI) – Section 7.1
- Ability to configure the access banner for the Console, Remote CLI and local CLI via the Console – Section 7.5
- Ability to configure the session inactivity time before session termination for the Console, Remote CLI and local CLI via the Console – Section 7.6.2
- Ability to update the TOE, and to verify the updates using a digital signature prior to installing those updates via the Console – Section 7.8
- Ability to configure the authentication failure parameters for FIA_AFL.1 for the Console, Remote CLI and local CLI via the Console – Section 7.2.1
- Ability to configure thresholds for SSH rekeying; - Section 6.5
- Ability to modify the behaviour of the transmission of audit data to an external IT entity – Section 6.7
- Ability to re-enable a locked Administrator account via the Console – Section 7.2.2
- Ability to set the time which is used for time-stamps via the local and remote CLI – Section 7.7
- Ability to manage the trusted public keys database via local and remote CLI – Section 6.5
- Manage the TOE's trust store and designate X.509v3 certificates as trust anchors via the Console– Sections 6.6

The TSS section 8.6.4 one of the three possible ways to authenticate to the TOE is locally via a physical connection to the serial port. Section 7.1 of the AGD states "Local users can gain access to the TOE by connecting via the keyboard/video ports or a serial port and a terminal emulator which accesses the local console (local CLI) and requires authenticating with their native username/password combination." The AGD clarifies any difference in behavior for the local CLI such as the ability to unlock a Console or CLI account that is locked by a CLI Security Administrator versus a Console Security Administrator can only unlock a Console user.

FMT_SMR.2 – *"The evaluator shall review the guidance documentation to ensure that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration."*

Section 6.1 of the AGD describes how to perform initial configuration of the TOE locally, including enabling FIPS Mode. Once configured, it can continue to be administered in this manner. Section 6.5 of the AGD discusses the configuration of the TOE to ensure that SSH is enabled. It also discusses the configuration of the SSH server to specify the set of algorithms and that the "none" MAC is not allowed for the SSH trusted path for the use of protecting remote CLI management communications. Section 6.8 discusses the configuration necessary to set up authentication using remote Active Directory credentials. Section 6.1 of the AGD indicates that when the TOE is in FIPS mode that it uses the cryptography described in the ST for all claimed cryptographic operations. These instructions are a part of the initial configuration to allow for local and remote administration of the TOE. Authenticating to the TOE locally and remotely is described in section 7.1.

FPT_APW_EXT.1 – This SFR does not contain any NDcPP AGD Assurance Activities.

FPT_SKP_EXT.1 – This SFR does not contain any NDcPP AGD Assurance Activities.

FPT_STM_EXT.1 – *"The evaluator examines the guidance documentation to ensure it instructs the administrator how to set the time. If the TOE supports the use of an NTP server, the guidance documentation instructs how a communication path is established between the TOE and the NTP server, and any configuration of the NTP client on the TOE to support this communication."*

Section 7.7 of the AGD describes how to set the time manually. The TOE does not claim NTP support.

FPT_TST_EXT.1 – *"The evaluator shall also ensure that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.*

For distributed TOEs the evaluator shall ensure that the guidance documentation describes how to determine from an error message returned which TOE component has failed the self-test."

Section 6.4 of the AGD describes the Power-On Self Tests, including the kinds of tests and failure scenarios. This is consistent with Section 8.5.4 of the ST. The AGD states that if errors are present in the self-tests the administrator should reboot the TOE. If further errors persist then the administrator needs to contact Forescout support. This section also states that the self-tests are performed during start-up and also can be done manually by a CLI Security Administrator.

FPT_TUD_EXT.1 – *"The evaluator shall verify that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation needs to describe how to query the loaded but inactive version.*

The evaluator shall verify that the guidance documentation describes how the verification of the authenticity of the update is performed (digital signature verification or verification of published hash). The description shall include the procedures for successful and unsuccessful verification. The description shall correspond to the description in the TSS.

If a published hash is used to protect the trusted update mechanism, the evaluator shall verify that the guidance documentation describes how the Security Administrator can obtain authentic published hash values for the updates.

For distributed TOEs the evaluator shall verify that the guidance documentation describes how the versions of individual TOE components are determined for FPT_TUD_EXT.1, how all TOE components are updated, and the error conditions that may arise from checking or applying the update (e.g. failure of signature verification, or exceeding available storage space) along with appropriate recovery actions. . The guidance documentation only has to describe the procedures relevant for the Security Administrator; it does not need to give information about the internal communication that takes place when applying updates.

If this was information was not provided in the TSS: For distributed TOEs, the evaluator shall examine the Guidance Documentation to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component.

If this was information was not provided in the TSS: If the ST author indicates that a certificate-based mechanism is used for software update digital signature verification, the evaluator shall verify that the Guidance Documentation contains a description of how the certificates are contained on the device. The evaluator also ensures that the Guidance Documentation describes how the certificates are installed/updated/selected, if necessary."

Step 3 in Section 6.3 of the AGD describes how to query the currently active version of the TOE. There is no option to for a delayed activation of an update.

Step 4 in Section 6.3 of the AGD describes how the TOE verifies the authenticity of the update using a digital signature. Part d in step 4 states the behavior of the TOE if the update fails verification. This is consistent with the TSS section 8.5.5.

Published hash is not used for verification of the updates.

The TOE does not claim use of a certificate based mechanism for software update digital verification.

FTA_SSL_EXT.1 – *"The evaluator shall confirm that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period."*

Section 7.6.2 of the AGD describes the steps to configure the TOE to terminate remotes sessions due to inactivity. The inactivity configuration for each interface is configured via the Console.

FTA_SSL.3 – *"The evaluator shall confirm that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination."*

Section 7.6.2 of the AGD describes the steps to configure the TOE to terminate remotes sessions due to inactivity. The inactivity configuration for each interface is configured via the Console.

FTA_SSL.4 – *"The evaluator shall confirm that the guidance documentation states how to terminate a local or remote interactive session."*

Section 7.6.1 of the AGD describes how to manually terminate a local or remote session on the TOE.

FTA_TAB.1 – *"The evaluator shall check the guidance documentation to ensure that it describes how to configure the banner message."*

Section 7.5 of the AGD describes how to configure the login banner for the local CLI, remote CLI and the Console. The banner for each of these interfaces is configured via the Console.

FTP_ITC.1 – *"The evaluator shall confirm that the guidance documentation contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.
Section 6.6 of the AGD provides information on X.509 Certificate Management and how to import certificates.
Sections 6.7 and 6.8 of the AGD provide information on how to establish the allowed protocols for remote Syslog and Active Directory entities and how the TOE handles TLS session interruption and recovery.

FTP_TRP.1/Admin – *"The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method."*

Section 6.1 of the AGD describes how to place the TOE into FIPS mode of operation. Once the TOE is in FIPS mode it only uses the cryptography described in the ST for all claimed cryptographic operations.
Section 6.2 of the AGD describes how to install the Console Management Application and the fact that the TOE only uses TLS1.2 and the ciphers are dictated by being in FIPS mode.
Section 6.5 of the AGD discusses the configuration of the TOE to ensure that ssh is enabled per the evaluated configuration.
Section 6.6 of the AGD provides information on X.509 Certificate Management describes how to generate the TOE's TLS server certificate.
Section 7.1 of the AGD describes how to establish both remote Console and remote CLI administrative sessions to the TOE.

4 Test Assurance Activities (Test Report)

The following sections demonstrate that all ATE Assurance Activities for the TOE have been met. This evidence has been presented in a manner that is consistent with the "Reporting for Evaluations Against NIAP-Approved Protection Profiles" guidance that has been provided by NIAP. Specific test steps and associated detailed results are not included in this report in order for it to remain non-proprietary. The test report is a summarized version of the test activities that were performed as part of creating the Evaluation Technical Report (ETR).

4.1 Platforms Tested and Composition

The evaluation team set up a test environment for the independent functional testing that allowed them to perform all test assurance activities across three models and over the relevant interfaces. The testing performed has an overlap between the tested models and interfaces to validate that the TOE performs the same regardless of the specific model.

The selection of models for testing was based upon ensuring that the software image provided for all four models produced the same results when tested. The 5140 (FS2), CEM-10 (FS4), CT-R (FS5), and 4130 (FS6) were deployed in the test laboratory as a representative set of the TOE's models. These models were used for the execution of the independent functional testing and vulnerability testing.

The evaluation team performed testing of the TSF functionality across all of the sampled models as well as each of the three available management interfaces (local console, remote CLI, remote Console). The full set of tests were replicated amongst the models and the tests were developed to stimulate each applicable TSF relevant interface. The testing performed on each interface of each sampled model, with the same logical interface SFR functionality, validated that the internal processing of the TOE would produce the same results regardless of the specific model or interface used to initiate or perform the processing. The testing is consistent with the use of the interfaces defined within the ST and AGD. Thus, the testing of the interfaces was based upon testing SFR functionality related to user actions over each interface.

The evaluation team conducted all testing activities of the TOE at the Booz Allen CCTL facility in Laurel, MD between July of 2021 and June of 2022.

Test Configuration 1:

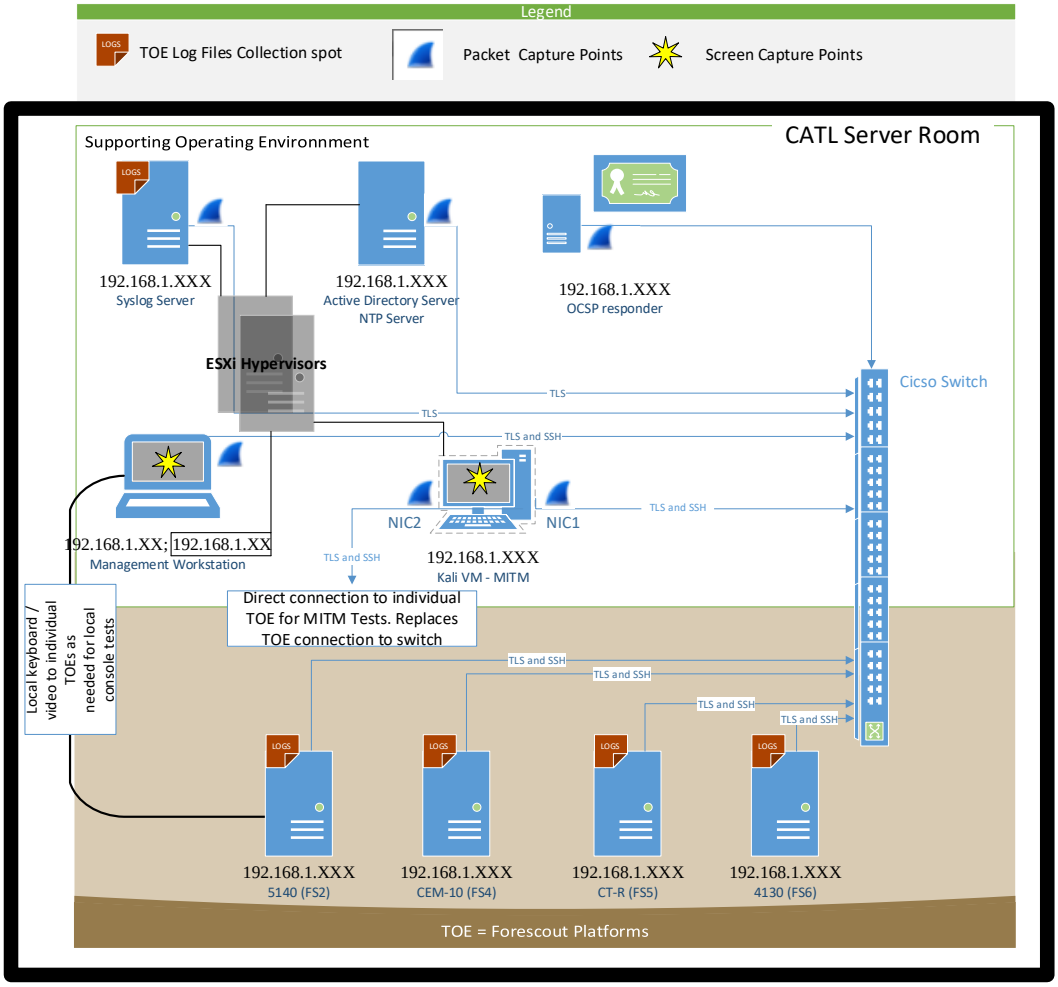
The TOE platforms (FS2, FS4, FS5, FS6) were configured to communicate with the following environment components:

- Function: Audit server
  - Platform: ProLiant DL380e Gen8
  - OS: Linux syslog04 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux
  - Protocols: TLS Server
  - Tools:
    - Syslog Server for recording of syslog data (rsyslogd 8.1901.0) (aka 2019.01)
    - Tcpdump 4.9.3

- Function: Active Directory
  - Platform: ProLiant DL380e Gen8
  - OS: Windows server 2012 R2
  - Protocols: TLS Server
  - Tools
    - WireShark: version 2.4.10

    .
- Function: OCSP Responder

- Platform: ProLiant DL380e Gen8
- OS: Linux catlsvcs 3.16.0-4-amd64 #1 SMP Debian 3.16.51-3 (2017-12-13) x86_64 GNU/Linux
- Protocols: HTTP
- Tools:
  - OpenSSL 1.0.1t
    - OCSP Responder for the tests related to certificate validation.



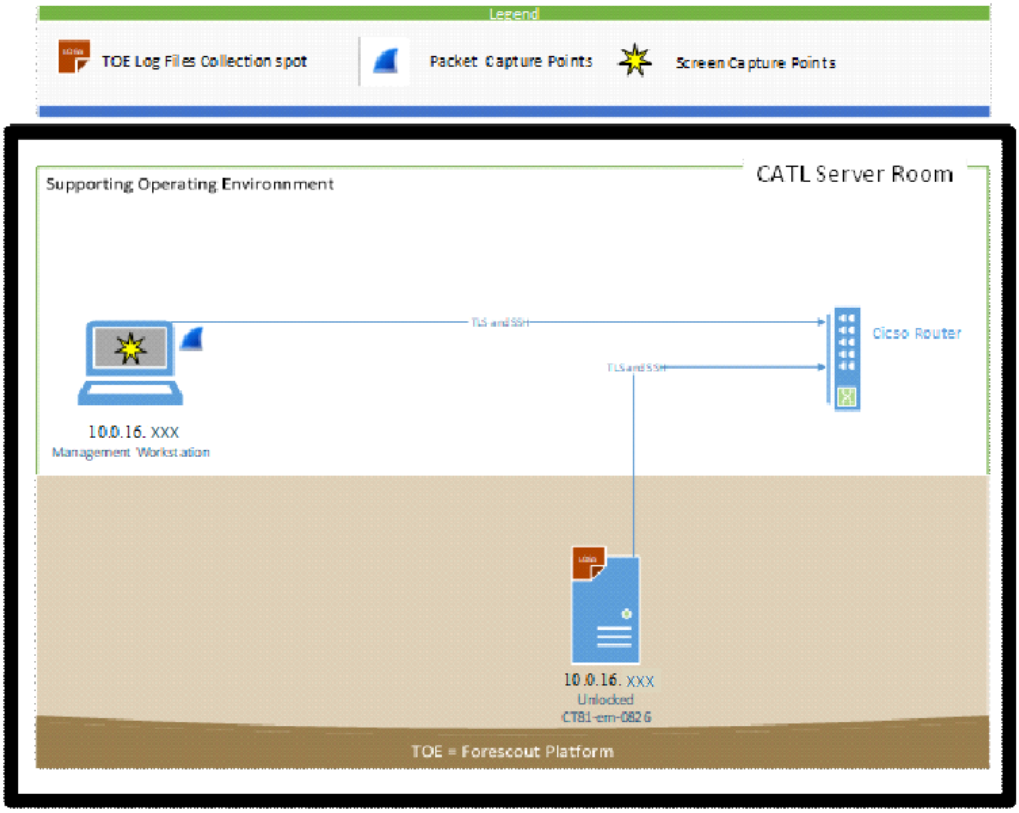
- Function: Management Workstations for local and remote administration and MITM testing. The following machines were used as the Management Workstation on the following IP addresses:
  - Platform 1 and 2: HP EliteBook Laptop with
    - OS: Windows 10 Version 20H2
    - Protocols: TLS Client, SSH Client
    - Interfaces 1,2, 3 and 8
    - Tools:
      - Forescout Console Application v8.3
      - WireShark: version 3.4.9 (platform 1) version 3.2.5 (platform 2)
      - Firefox Quantum: version 68.0.1
      - Google Chrome: version 102.0.5005.63
      - PuTTY SSH Client: version .73
      - Bitvise SSH Client 8.47

- Platform: ProLiant DL380e Gen8
  - OS: Linux kali1 4.15.0-kali2-amd64 #1 SMP Debian 4.15.11-1kali1 (2018-03-21) x86_64 GNU/Linux
  - Protocols: TLS Client, SSH Client
  - Tools:
    - Tcpdump: version 4.9.3
    - Libpcap version 1.8.1
    - OpenSSL version 1.0.2k

Test Configuration 2:

Special Test Case: FAU_STG_EXT.1 part b (Test 004 of the Test Plan)

The developer of the TOE had to provide a build with an unlocked embedded OS in order to perform Test Case 004. In order to emphasize the difference in the TOE environments, a separate test network was configured for this test. This network contained the TOE and the test workstation as the two parts of this tests did not require any other environmental support. See network configuration 2 below.



4.2 Omission Justification

TOE Functional Differences –
The Security Target does describe that there are two configurations (CEM and Appliance) of the TOE and that both would be tested. However, a review of the Security Target does not describe any variations of functionality on a per TOE model basis (i.e. 5140 vs CEM-10 vs CT-R). There is only one PP scoped functional difference described in the ST. This difference involves the trusted update functionality. The CEM configuration allows for a delayed update mechanism and the appliance configuration does not.

Therefore:

- All machines configured as an Appliance are considered equivalent with regards TOE functional differences.
- All machines configured as an CEM would be considered equivalent with regards TOE functional differences.

TOE Management Interface Differences –
Every model, whether configured as a CEM or Appliance, has the same dedicated Ethernet Management Port and Serial Port. These ports were used for all testing performed by the evaluation team. Therefore, all models are considered equivalent with regards to the TOE Management Interfaces.

Differences in TOE Software Binaries –
The same software binary image is installed on all models. The evaluation team has determined that there is no difference in the software binaries that would impact the testing of the SFRs claimed within this evaluation. Therefore, all models are considered equivalent with regards to the TOE software binary.

Platform/Hardware Dependencies –
The evaluation team assessed the platform and hardware dependencies of the TOE models and determined that the only differences that could be impactful are based upon the TOE models' processor. Therefore, the evaluation team determined at least one model from each of these four processor groups must be tested:

- Intel Celeron J19XX (Bay Trail)
- Intel Xeon E5 26XX v3 (Haswell)
- Intel Xeon Silver 41XX (Skylake) or Intel Xeon Gold 51XX (Skylake) or Intel Xeon Gold 61XX (Skylake)
- Gen 8 Intel® Core™ i5-8500T

Conclusion:

Based upon the evaluation team's analysis, there are only two differences between TOE models that can be impactful: configuration (CEM vs Appliance) and processor. The evaluation team has made the following conclusions regarding which models need to be tested and those covered by equivalency:

- All functionality needs to be fully tested against at least one TOE model configured as a CEM and one TOE model configured as an Appliance.
  - Satisfied by testing model CEM-10 Rev40 configured as the Enterprise Manager
  - Satisfied by testing models 5140, 4130, CT-R configured as Appliances
- At least one model from each of these five processor groups must be tested:
  - Intel Celeron J19XX (Bay Trail)
    - Satisfied by testing model CT-R (FS5)
    - Model 5110 is considered equivalent
  - Intel Xeon E5 26XX v3 (Haswell)
    - Satisfied by testing model CEM-10 Rev40 (FS4)
    - Models CEM-05 Rev40, CEM-25 Rev40, CEM-50 Rev40, CEM-100 Rev40, CEM-200 Rev40, CT-100 Rev40, CT-1000 Rev40,CT-2000 Rev40, CT-4000 Rev40, and CT-10000 Rev40 are considered equivalent
  - Intel Xeon Silver 41XX (Skylake) or Intel Xeon Gold 51XX (Skylake) or Intel Xeon Gold 61XX (Skylake)
    - Satisfied by testing model 5140 (FS2)
    - Models CEM-05 Rev50, CEM-10 Rev50, CEM-25 Rev50, CEM-50 Rev50, CEM-100 Rev50, CEM-150 Rev50, CEM-200 Rev50, CT-100 Rev50, CT-1000 Rev50, CT-2000 Rev50, CT-4000 Rev50, CT-10000 Rev50, 5120, and 5160 are considered equivalent
  - Gen 8 Intel® Core™ i5-8500T
    - Satisfied by testing model 4130 (FS6)

4.3 Test Cases

The evaluation team completed the functional testing activities within the laboratory environment. The evaluation team conducted a set of testing that includes all ATE Assurance Activities as specified by the 'collaborative Protection Profile for Network Devices Version 2.2e' (NDcPP) for the SFRs claimed in the Security Target. The evaluators reviewed the NDcPP to identify the security functionality that must be verified through functional testing. This is prescribed by the Assurance Activities for each SFR.

If an SFR is not listed, one of the following conditions applies:

- The Assurance Activity for the SFR specifically indicates that it is simultaneously satisfied by completing a test Assurance Activity for a different SFR.
- The Assurance Activity for the SFR does not specify any actions related to ATE activities (e.g. FPT_APW_EXT.1).

Note that some SFRs do not have Assurance Activities associated with them at the element level (e.g. FPT_TST_EXT.1.1). In such cases, testing for the SFR is considered to be satisfied by completion of all Assurance Activities at the component level.

The following lists for each ATE Assurance Activity, the test objective, test instructions, test steps, and test results. Note that unless otherwise specified, the test configuration is to be in the evaluated configuration as defined by the AGD. For example, some tests require the TOE to be brought out of the evaluated configuration to temporarily disable cryptography to prove that the context of transmitted data is accurate. As part of the cleanup for each test, the TOE is returned to the evaluated configuration.

4.3.1 Security Audit

Test Case Number	001
SFR	FAU_GEN.1
Test Objective	The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the table of audit events and administrative actions listed above. This should include all instances of an event: for instance, if there are several different I&A mechanisms for a system, the FIA_UIA_EXT.1 events must be generated for each mechanism. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. If HTTPS is implemented, the test demonstrating the establishment and termination of a TLS session can be combined with the test for an HTTPS session. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the guidance documentation, and that the fields in each audit record have the proper entries. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of auditable events to TOE components in the Security Target. For all events involving more than one TOE component when an audit event is triggered, the evaluator has to check that the event has been audited on both sides (e.g. failure of building up a secure communication channel between the two components). This is not limited to error cases but includes also events about successful actions like successful build up/tear down of a secure communication channel between TOE components. Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.
Test Instructions	Execute this test per the test steps.
Test Steps	**Startup and shut-down of audit functions:**

	1. Authenticate to the TOE via the CLI. 2. Execute the following command to reboot the TOE: "reboot" NOTE: All additional required audit events in this requirement are collected throughout the course of executing the other test cases.
Test Results	The TOE generated all audit records as expected. The format of the audit records contained all of the parts required as expected - Pass
Execution Method	Manual

Test Case Number	002
SFR	FAU_GEN.2
Test Objective	This activity should be accomplished in conjunction with the testing of FAU_GEN.1.1. For distributed TOEs the evaluator shall verify that where auditable events are instigated by another component, the component that records the event associates the event with the identity of the instigator. The evaluator shall perform at least one test on one component where another component instigates an auditable event. The evaluator shall verify that the event is recorded by the component as expected and the event is associated with the instigating component. It is assumed that an event instigated by another component can at least be generated for building up a secure channel between two TOE components. If for some reason (could be e.g. TSS or Guidance Documentation) the evaluator would come to the conclusion that the overall TOE does not generate any events instigated by other components, then this requirement shall be omitted.
Test Instructions	Execute this test per the test steps.
Test Steps	The first part of this test assurance activity is accomplished in conjunction with the testing of FAU_GEN.1.1. The second part of this test assurance activity is not applicable because the TOE is not a distributed TOE.
Test Results	Pass
Execution Method	Manual

Test Case Number	003
SFR	FAU_STG_EXT.1
Test Objective	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: a) Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE during several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing. The evaluator shall verify that the TOE is capable of transferring audit data to an external audit server automatically without administrator intervention.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Ensure the syslog plugin on the TOE is stopped.

	2. Begin capturing packets between the TOE and the remote audit server. 3. Start the syslog plugin on the TOE to cause the automatic transmission of audit data to the remote audit server. 4. Stop capturing packets between the TOE and the remote audit server. 5. Inspect the packet capture and verify that the audit data is encrypted in transit. 6. Verify that the audit data was received on the remote audit server. 7. Record the remote audit server software version and program name.
Test Results	The TOE successfully negotiated a TLS connection to the audit server. The evaluator examined the captured packets and verified that the data transmitted between the TOE and the remote audit server were encrypted. The evaluator also compared the local audit records on the TOE against those received by the audit server and confirmed that they matched. The remote audit server software and version is: rsyslogd 8.1901.0 (aka 2019.01) - Pass
Execution Method	Manual

Test Case Number	004
SFR	FAU_STG_EXT.1
Test Objective	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: b) Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behavior defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that 1) The audit data remains unchanged with every new auditable event that should be tracked but that the audit data is recorded again after the local storage for audit data is cleared (for the option 'drop new audit data' in FAU_STG_EXT.1.3). 2) The existing audit data is overwritten with every new auditable event that should be tracked according to the specified rule (for the option 'overwrite previous audit records' in FAU_STG_EXT.1.3) 3) The TOE behaves as specified (for the option 'other action' in FAU_STG_EXT.1.3).
Test Instructions	Execute this test per the test steps. Use Configuration 2
Test Steps	**Purge of Database log** 1. From the test machine, run the test script to fill the local audit storage database with several lines of data. 2. Let the script in Step 1 run until the TOE reports a log purge event. 3. Verify that the oldest log entry is deleted in a first-in-first-out (FIFO) manner and that an audit record for the purge event is generated. **OS Log Rollover**

	1. From the test machine, run a script to fill the audit storage OS log. 2. Let the script in Step 1 run until the OS logs start to populate. (auditlog.1, auditlog.2, etc.) 3. Once there are 5 log files, verify that the oldest log file is deleted and is replaced with the next oldest. (auditlog.4 is deleted and replaced with auditlog.3)
Test Results	Due to the default operational TOE having a locked embedded operating system, this test was performed on Configuration 2 which used a special build where the embedded operating system was unlocked. The TOE successfully purged the database log files and performed rollover for the OS logs as specified in the ST. - Pass
Execution Method	Manual

Test Case Number	005
SFR	FAU_STG_EXT.1
Test Objective	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: c) Test 3: If the TOE complies with FAU_STG_EXT.2/LocSpace the evaluator shall verify that the numbers provided by the TOE according to the selection for FAU_STG_EXT.2/LocSpace are correct when performing the tests for FAU_STG_EXT.1.3
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE does not claim FAU_STG_EXT.2/LocSpace.
Test Results	Pass
Execution Method	Manual

Test Case Number	006
SFR	FAU_STG_EXT.1
Test Objective	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: d) Test 4: For distributed TOEs, Test 1 defined above should be applicable to all TOE components that forward audit data to an external audit server. For the local storage according to FAU_STG_EXT.1.2 and FAU_STG_EXT.1.3 the Test 2 specified above shall be applied to all TOE components that store audit data locally. For all TOE components that store audit data locally and comply with FAU_STG_EXT.2/LocSpace Test 3 specified above shall be applied. The evaluator shall verify that the transfer of audit data to an external audit server is implemented.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE is not a distributed TOE.
Test Results	Pass
Execution Method	Manual

4.3.2 Cryptographic Support

Test cases for FCS_CKM.1 (partially), FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash, and FCS_RBG_EXT.1 are not included within this section. ATE Assurance Activities have been satisfied by successfully achieving CAVP certificates for these SFRs. Refer to the results of the CAVP certificates listed below:

SFR	OpenSSL Implementation CAVP #C1887 and #A1941	Bouncy Castle Implementation CAVP #C1888 and #A1959
FCS_CKM.1	RSA per FIPS 186-4 Key Generation	N/A
	FFC using Diffie-Hellman group 14, per RFC 3526 Section 3	N/A
FCS_CKM.2	RSA Key Establishment per RSAES-PKCS-v1_5	RSA Key Establishment per RSAES-PKCS-v1_5
	Diffie-Hellman group 14 Key Establishment RFC 3526 Section 3	N/A
FCS_COP.1/ DataEncryption	AES CTR: 128 and 256 bits AES CBC: 128 and 256 bits AES GCM: 128 and 256 bits	AES CBC: 128 and 256 bits AES GCM: 256 bits
FCS_COP.1/SigGen	RSA FIPS 186-4 Signature Services 2048 bits	RSA FIPS 186-4 Signature Services 2048 bits
FCS_COP.1/ Hash	SHS: SHA-1, SHA-256, SHA-384, and SHA-512	SHS: SHA-1, SHA-256, and SHA-384
FCS_COP.1/KeyedHash	HMAC-SHA-1, HMAC-SHA-256, HMAC-384, and HMAC-SHA-512	HMAC-SHA-1, HMAC-SHA-256, and HMAC-384
FCS_RBG_EXT.1	CTR DRBG	Hash DRBG

Test Case Number	093
SFR	FCS_CKM.1.1 – TD0580
Test Objective	Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1.
Test Instructions	Execute this test per the test steps.
Test Steps	Refer to FCS_CKM.2.1 – Test Case 007.
Test Results	Pass
Execution Method	Manual

Test Case Number	007
SFR	FCS_CKM.2.1 – TD0580
Test Objective	The evaluator shall verify the correctness of the TSF's implementation of safe-prime groups by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses safe-prime groups. This test must be performed for each safe-prime group that each protocol uses.
Test Instructions	Execute this test per the test steps.
Test Steps	This test is satisfied by the testing of FTP_TRP.1/Admin and FTP_ITC.1.
Test Results	Pass
Execution Method	Manual

Test Case Number	008
SFR	FCS_SSHS_EXT.1.2 – TD0631
Test Objective	Test objective: The purpose of these tests is to verify server supports each claimed client authentication method. Test 1: For each supported client public-key authentication algorithm, the evaluator shall configure a remote client to present a public key corresponding to that authentication method (e.g., 2048-bit RSA key when using ssh-rsa public key). The evaluator shall establish sufficient separate SSH connections with an appropriately configured remote non-TOE SSH client to demonstrate the use of all applicable public key algorithms. It is sufficient to observe the successful completion of the SSH Authentication Protocol to satisfy the intent of this test.
Test Instructions	Execute this test per the test steps.

Test Steps	1. On the test machine, configure the SSH client to authenticate using the ssh-rsa public key algorithm. 2. Begin capturing packets between the SSH client and the TOE. 3. Connect to the TOE using the SSH client and confirm that the connection was successful. 4. Stop capturing packets.
Test Results	The TOE successfully negotiated a connection using when a valid public key was supplied to the TOE. - Pass
Execution Method	Manual

Test Case Number	009
SFR	FCS_SSHS_EXT.1.2 – TD0631
Test Objective	Test 2: The evaluator shall choose one client public key authentication algorithm supported by the TOE. The evaluator shall generate a new client key pair for that supported algorithm without configuring the TOE to recognize the associated public key for authentication. The evaluator shall use an SSH client to attempt to connect to the TOE with the new key pair and demonstrate that authentication fails.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Generate a new SSH ssh-rsa keypair. 2. Using the private key from the keypair generated in Step 1, attempt to authenticate to the TOE via the CLI using SSH with a valid username. 3. Verify that the authentication attempt to the TOE fails.
Test Results	The TOE correctly did not negotiate a connection when an invalid public key was supplied to the TOE. - Pass
Execution Method	Manual

Test Case Number	094
SFR	FCS_SSHS_EXT.1.2 – TD0631
Test Objective	Test 3: [Conditional] If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept password-based authentication and demonstrate that user authentication succeeds when the correct password is provided by the connecting SSH client.
Test Instructions	Execute this test per the test steps.
Test Steps	1. On TOE, configure the SSH server to authenticate connections using password-based authentication. 2. Initiate a SSH connection to the TOE using valid SSH credentials. 3. Verify that the authentication attempt was successful.
Test Results	The TOE successfully negotiated a connection when valid SSH authentication credentials were supplied to the TOE. - Pass
Execution Method	Manual

Test Case Number	095
SFR	FCS_SSHS_EXT.1.2 – TD0631
Test Objective	Test 4: [Conditional] If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept password-based authentication and demonstrate that user authentication fails when the incorrect password is provided by the connecting SSH client.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Initiate a SSH connection to the TOE using an invalid password. 2. Verify that the authentication attempt was unsuccessful.
Test Results	The TOE correctly did not negotiate a connection when invalid SSH authentication

	credentials were supplied to the TOE. - Pass
Execution Method	Manual

Test Case Number	010
SFR	FCS_SSHS_EXT.1.3
Test Objective	The evaluator shall demonstrate that if the TOE receives a packet larger than that specified in this component, that packet is dropped.
Test Instructions	Execute this test per the test steps.
Test Steps	1. On the test machine, execute the command to send a large packet to the TOE. 2. Verify that the TOE drops any packet larger than the specified size.
Test Results	The TOE successfully dropped the large packet. - Pass
Execution Method	Manual

Test Case Number	011
SFR	FCS_SSHS_EXT.1.4
Test Objective	The evaluator must ensure that only claimed ciphers and cryptographic primitives are used to establish an SSH connection. To verify this, the evaluator shall start session establishment for an SSH connection from a remote client (referred to as 'remote endpoint' below). The evaluator shall capture the traffic exchanged between the TOE and the remote endpoint during protocol negotiation (e.g. using a packet capture tool or information provided by the endpoint, respectively). The evaluator shall verify from the captured traffic that the TOE offers all the ciphers defined in the TSS for the TOE for SSH sessions, but no additional ones compared to the definition in the TSS. The evaluator shall perform one successful negotiation of an SSH session to verify that the TOE behaves as expected. It is sufficient to observe the successful negotiation of the session to satisfy the intent of the test. If the evaluator detects that not all ciphers defined in the TSS for SSH are supported by the TOE and/or the TOE supports one or more additional ciphers not defined in the TSS for SSH, the test shall be regarded as failed.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the SSH client test machine and the TOE. 2. Authenticate to the TOE using SSH. 3. Stop capturing packets. 4. Verify an SSH connection was successfully established. 5. Examine the packet capture's "Server: Key Exchange Init" message and verify that no other encryption algorithms other than those claimed in the Security Target are listed in the "encryption_algorithms_server_to_client" string.
Test Results	The TOE only responded with and successfully negotiated a connection using the claimed "Server: Key Exchange Init" algorithms: aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com - Pass
Execution Method	Manual

Test Case Number	012
SFR	FCS_SSHS_EXT.1.5 – TD0631
Test Objective	Test objective: This test case is meant to validate that the TOE server will support host public keys of the claimed algorithm types. Test 1: The evaluator shall configure (only if required by the TOE) the TOE to use

	each of the claimed host public key algorithms. The evaluator will then use an SSH client to confirm that the client can authenticate the TOE server public key using the claimed algorithm. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the test machine and the TOE. 2. Authenticate to the TOE via SSH using a ssh client and only selecting ssh-rsa as the public key algorithm. 3. Stop capturing packets between the test machine and the TOE. 4. Verify that ssh-rsa public key algorithm was used to negotiate the SSH connection.
Test Results	The TOE successfully negotiated connections when the SSH client was configured to use the claimed ssh-rsa host public key algorithm – Pass
Execution Method	Manual

Test Case Number	013
SFR	FCS_SSHS_EXT.1.5 – TD0631
Test Objective	Test objective: This negative test case is meant to validate that the TOE server does not support host public key algorithms that are not claimed. Test 2: The evaluator shall configure a non-TOE SSH client to only allow it to authenticate an SSH server host public key algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection from the non-TOE SSH client to the TOE SSH server and observe that the connection is rejected.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Generate a new SSH ssh-dss keypair using a ssh client. 2. Attempt to authenticate to the TOE via the CLI using the SSH client using the key pair that was generated in Step 1. 3. Verify that the authentication attempt to the TOE fails.
Test Results	The TOE correctly did not negotiate a connection when the SSH client was configured to use the unclaimed ssh-dss host public key algorithm - Pass
Execution Method	Manual

Test Case Number	014
SFR	FCS_SSHS_EXT.1.5 Removed per TD0631
~~**Test Objective**~~	~~Test 3: The evaluator shall configure an SSH client to only allow a public key algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection from the SSH client to the TOE and observe that the connection is rejected.~~
~~**Test Instructions**~~	~~Execute this test per the test steps.~~
~~**Test Steps**~~	~~1. Begin capturing packets between the test machine and the TOE.~~ ~~2. Authenticate to the TOE via SSH using a ssh client with only ssh-dss selected as the host key algorithm.~~ ~~3. Stop capturing packets between the test machine and the TOE.~~ ~~4. Verify that the TOE rejects the SSH connection.~~ ~~5. Examine packet capture and verify that the ssh-dss public key algorithm was offered by the test machine (client) in the "server_host_key_algorithms" string.~~
~~**Test Results**~~	~~Pass~~
~~**Execution Method**~~	~~Manual~~

Test Case Number	015
SFR	FCS_SSHS_EXT.1.6
Test Objective	Test 1: [conditional, if an HMAC or AEAD_AES_*_GCM algorithm is selected in the ST] The evaluator shall establish an SSH connection using each of the algorithms, except "implicit", specified by the requirement. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test. Note: To ensure the observed algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the test machine and the TOE. 2. Authenticate to the TOE via SSH using the command. 3. Stop capturing packets between the test machine and the TOE. 4. Verify that the encryption algorithm specified in Step 2 was used to negotiate the SSH connection. 5. Repeat Steps 1-4, except use the MAC algorithm: hmac-sha2-256 6. Repeat Steps 1-4, except use the MAC algorithm: hmac-sha2-512
Test Results	The TOE successfully negotiated a connection when the client was configured to use the claimed SSH HMAC algorithms: hmac-sha1, hmac-sha2-256, hmac-sha2-512) - Pass
Execution Method	Manual

Test Case Number	016
SFR	FCS_SSHS_EXT.1.6
Test Objective	Test 2: [conditional, if an HMAC or AEAD_AES_*_GCM algorithm is selected in the ST] The evaluator shall configure an SSH client to only allow a MAC algorithm that is not included in the ST selection. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails. Note: To ensure the proposed MAC algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the test machine and the TOE. 2. Authenticate to the TOE via SSH using the command. 3. Stop capturing packets between the test machine and the TOE. 4. Verify that the TOE rejects the SSH connection.
Test Results	The TOE correctly did not negotiate connections when the client was configured to use unclaimed MAC algorithms. - Pass
Execution Method	Manual

Test Case Number	017
SFR	FCS_SSHS_EXT.1.7
Test Objective	Test 1: The evaluator shall configure an SSH client to only allow the diffie-hellman-group1-sha1 key exchange. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails.
Test Instructions	Execute this test per the test steps.

Test Steps	1. Begin capturing packets between the test machine and the TOE. 2. Authenticate to the TOE via SSH using the command. 3. Stop capturing packets between the test machine and the TOE. 4. Verify that the TOE rejects the SSH connection.
Test Results	The TOE correctly did not negotiate connections when the client was configured to use the unclaimed diffie-hellman-group1-sha1 key exchange. - Pass
Execution Method	Manual

Test Case Number	018
SFR	FCS_SSHS_EXT.1.7
Test Objective	Test 2: For each allowed key exchange method, the evaluator shall configure an SSH client to only allow that method for key exchange, attempt to connect from the client to the TOE, and observe that the attempt succeeds.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the test machine and the TOE. 2. Authenticate to the TOE via SSH using the command. 3. Stop capturing packets between the test machine and the TOE. 4. Verify that the TOE accepts the SSH connection.
Test Results	The TOE successfuly negotiate connections when the client was configured to use the claimed diffie-hellman-group14-sha1 key exchange. - Pass
Execution Method	Manual

Test Case Number	019
SFR	FCS_SSHS_EXT.1.8
Test Objective	The evaluator needs to perform testing that rekeying is performed according to the description in the TSS. The evaluator shall test both, the time-based threshold and the traffic-based threshold. For testing of the time-based threshold, the evaluator shall use an SSH client to connect to the TOE and keep the session open until the threshold is reached. The evaluator shall verify that the SSH session has been active longer than the threshold value and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at the maximum allowed value of one hour of session time, but the value used for testing shall not exceed one hour. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. For testing of the traffic-based threshold the evaluator shall use the TOE to connect to an SSH client and shall transmit data to and/or receive data from the TOE within the active SSH session until the threshold for data protected by either encryption key is reached. It is acceptable if the rekey occurs before the threshold is reached (e.g. because the traffic is counted according to one of the alternatives given in the Application Note for FCS_SSHS_EXT.1.8). The evaluator shall verify that more data has been transmitted within the SSH session than the threshold allows and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at

	the maximum allowed value of one gigabyte of transferred traffic, but the value used for testing shall not exceed one gigabyte. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, the evaluator needs to verify that the threshold(s) can be configured as described in the guidance documentation and the evaluator needs to test that modification of the thresholds is restricted to Security Administrators (as required by FMT_MOF.1/Functions). In cases where data transfer threshold could not be reached due to hardware limitations it is acceptable to omit testing of this (SSH rekeying based on data transfer threshold) threshold if both the following conditions are met: a) An argument is present in the TSS section describing this hardware-based limitation and b) All hardware components that are the basis of such argument are definitively identified in the ST. For example, if specific Ethernet Controller or WiFi radio chip is the root cause of such limitation, these chips must be identified.
Test Instructions	Execute this test per the test steps.
Test Steps	**Time-based Rekey:** 1. Authenticate to the TOE via the Console using the administrator account. 2. Disable the inactivity timeout: a. Navigate to “Tools” > “Options” > “CounterACT User Profiles” > “Password and Sessions”. b. Select the “Session” tab. c. Check the box, “Terminate inactive sessions after” and specify a value of 90 minutes. d. Select “Apply”. 3. Log out of the TOE. 4. Establish a new SSH session. 5. Wait 1 hour and verify that the TOE generates an audit record for the SSH rekey performed by the TOE. **Traffic-based Rekey:** 1. Configure the TOE to perform a SSH rekey after no greater than 1GB of data is exchanged: `ssh -t server -o RekeyLimit "500M 1h"` 2. Transfer a 500 MB file to the TOE via SSH (i.e. using SCP). 3. Verify that the TOE generates an audit record for the SSH rekey performed by the TOE.
Test Results	The TOE successfully performed a time based rekey at no greater than 1 hour. The

	TOE also successfully performed a traffic based rekey at no greater than 1 GB of exchanged data. - Pass
Execution Method	Manual

Test Case Number	020
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Test Instructions	Execute this test per the test steps.
Test Steps	1. Configure a remote server such that only the following ciphersuite is supported: TLS_RSA_WITH_AES_256_CBC_SHA 2. Begin capturing packets between the remote server and the TOE. 3. Initiate a TLS connection from the TOE to the remote server. 4. Stop capturing packets. 5. Examine the packet capture and observe that the ciphersuite in Step 1 was used as part of the TLS connection negotiation. 6. Repeat Steps 1-5, except in Step 1 use "TLS_RSA_WITH_AES_128_CBC_SHA256". 7. Repeat Steps 1-5, except in Step 1 use "TLS_RSA_WITH_AES_256_CBC_SHA256". 8. Repeat Steps 1-5, except in Step 1 use "TLS_RSA_WITH_AES_256_GCM_SHA384".
Test Results	The TOE successfully negotiated a connection with each of the claimed ciphers. - Pass
Execution Method	Manual

Test Case Number	021
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 2: The evaluator shall attempt to establish the connection using a server with a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field and verify that a connection is established. The evaluator will then verify that the client rejects an otherwise valid server certificate that lacks the Server Authentication purpose in the extendedKeyUsage field, and a connection is not established. Ideally, the two certificates should be identical except for the extendedKeyUsage field.
Test Instructions	Execute this test per the test steps.
Test Steps	1. On the remote server, load the certificate containing the Server Authentication purpose. 2. Begin capturing packets between the TOE and the remote server. 3. Cause the TOE to establish a TLS connection to the remote server. 4. Stop capturing packets between the TOE and the remote server. 5. Inspect the packet capture and verify that the TOE successfully established

	a connection to the remote server. 6. On the remote server, load the certificate without the Server Authentication purpose. 7. Repeat Steps 2-4. 8. Inspect the packet capture and verify that the TOE failed to establish a connection to the remote server.
Test Results	The TOE successfully negotiated a connection to the remote server when the server presented a server certificate with the Server Authentication purpose. The TOE correctly did not negotiate a connection to the remote server when the server presented a certificate lacking the Server Authentication purpose. - Pass
Execution Method	Manual

Test Case Number	022
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 3: The evaluator shall send a server certificate in the TLS connection that does not match the server-selected ciphersuite (for example, send an ECDSA certificate while using the TLS_RSA_WITH_AES_128_CBC_SHA ciphersuite). The evaluator shall verify that the TOE disconnects after receiving the server's Certificate handshake message.
Test Instructions	Execute this test per the test steps.
Test Steps	1. On a remote server, load the certificate generated in setup. 2. Run the tool on the MITM test system by executing the command. 3. Begin capturing packets between the TOE and the remote server. 4. Establish a TLS connection between the TOE and the remote server. 5. Stop capturing packets. 6. Inspect the packet capture to verify that a TLS connection could not be established, and that the TOE client disconnected after receiving the server's Certificate handshake message.
Test Results	The TOE correctly disconnects from the remote server when the server certificate did not match the server-selected ciphersuite - Pass
Execution Method	Manual

Test Case Number	023
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 4: The evaluator shall perform the following 'negative tests': a) The evaluator shall configure the server to select the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the client denies the connection. b) Modify the server's selected ciphersuite in the Server Hello handshake message to be a ciphersuite not presented in the Client Hello handshake message. The evaluator shall verify that the client rejects the connection after receiving the Server Hello. c) [conditional]: If the TOE presents the Supported Elliptic Curves/Supported Groups Extension the evaluator shall configure the server to perform an ECDHE or DHE key exchange in the TLS connection using a non-supported curve/group (for example P-192) and shall verify that the TOE disconnects after receiving the server's Key Exchange handshake message.

Test Instructions	Execute this test per the test steps.
Test Steps	a) 1. Run the tool on the MITM test system by executing the command. 2. Begin capturing packets between the TOE and the remote server. 3. Establish a TLS connection between the TOE and the remote server. 4. Stop capturing packets. 5. Verify that the TLS connection could not be established, and the client refused the server's ciphersuite selection. b) 1. Repeat Steps 1-5 in part (a), except using the command for part (b). c) The TOE does not present the Supported Elliptic Curves / Supported Groups Extension. Therefore, this test is not applicable.
Test Results	The TOE correctly refused the connections in both instances. - Pass
Execution Method	Manual

Test Case Number	024
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 5: The evaluator performs the following modifications to the traffic: a) Change the TLS version selected by the server in the Server Hello to a non-supported TLS version and verify that the client rejects the connection. b) [conditional]: If using DHE or ECDH, modify the signature block in the Server's Key Exchange handshake message, and verify that the handshake does not finished successfully, and no application data flows. This test does not apply to cipher suites using RSA key exchange. If a TOE only supports RSA key exchange in conjunction with TLS, then this test shall be omitted.
Test Instructions	Execute this test per the test steps.
Test Steps	a) 1. Begin capturing packets between the TOE and the TLS server. 2. Run the tool on the MITM test system by executing the command. 3. Initiate a connection from the TOE to the server. 4. Stop capturing packets. 5. Confirm the TOE rejects the connection. b) The TOE only supports RSA key exchange in conjunction with TLS. Therefore, this test is not applicable.
Test Results	The TOE correctly refused the connection when an non-claimed TLS version was received. - Pass
Execution Method	Manual

Test Case Number	092
SFR	FCS_TLSC_EXT.1.1
Test Objective	Test 6: The evaluator performs the following 'scrambled message tests': a) Modify a byte in the Server Finished handshake message and verify that the handshake does not finish successfully and no application data flows. b) Send a garbled message from the server after the server has issued the

	ChangeCipherSpec message and verify that the handshake does not finish successfully and no application data flows. c) Modify at least one byte in the server's nonce in the Server Hello handshake message and verify that the client rejects the Server Key Exchange handshake message (if using a DHE or ECDHE ciphersuite) or that the server denies the client's Finished handshake message.
Test Instructions	Execute this test per the test steps.
Test Steps	a) 1. Begin capturing packets between the TOE and the TLS server. 2. Run the tool on the MITM test system by executing the command. 3. Initiate a connection from the TOE to the server. 4. Stop capturing packets. 5. Confirm the TOE rejects the connection. b) 1. Repeat Steps 1-5 in part (a), except using the command for part (b). c) 1. Repeat Steps 1-5 in part (a), except using the command for part (c).
Test Results	The TOE correctly does not complete the handshake and no application data flowed for any of the scenarios. - Pass
Execution Method	Manual

Test Case Number	025
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses,

	suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: a) Test 1 [conditional]: The evaluator shall present a server certificate that contains a CN that does not match the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each identifier type (e.g. IPv4, IPv6, FQDN) supported in the CN. When testing IPv4 or IPv6 addresses, the evaluator shall modify a single decimal or hexadecimal digit in the CN. Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass Test 1.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install a certificate on the remote server that does not contain the Subject Alternative Name (SAN) but contains a Common Name (CN) that does not match the reference identifier. 2. Begin capturing packets between the TOE and the server. 3. Connect the TOE to the server using TLS. 4. Stop capturing packets. 5. Verify that the connection fails.
Test Results	The TOE correctly rejects the certificate and does not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	026
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit

	numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: b) Test 2 [conditional]: The evaluator shall present a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each supported SAN type (e.g. IPv4, IPv6, FQDN, URI). When testing IPv4 or IPv6 addresses, the evaluator shall modify a single decimal or hexadecimal digit in the SAN.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install a certificate on the remote server that contains a CN that matches the reference identifier, contains the SAN extension but does not contain an identifier in the SAN that matches the reference identifier. 2. Begin capturing packets between the TOE and the server. 3. Connect the TOE to the server. 4. Stop capturing packets between the TOE and the server. 5. Verify the connection fails. 6. Repeat this test for supported SAN type.
Test Results	The TOE correctly rejects the certificates and does not negotiate either of the connection attempts. - Pass
Execution Method	Manual

Test Case Number	027
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the

	evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: c) Test 3 [conditional]: If the TOE does not mandate the presence of the SAN extension, the evaluator shall present a server certificate that contains a CN that matches the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection succeeds. The evaluator shall repeat this test for each identifier type (e.g. IPv4, IPv6, FQDN) supported in the CN. If the TOE does mandate the presence of the SAN extension, this Test shall be omitted.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install a certificate on the server that contains a CN that matches the reference identifier but does not contain the SAN extension 2. Begin capturing packets between the TOE and the server. 3. Connect the TOE to the server. 4. Stop capturing packets. 5. Verify the connection succeeds.
Test Results	The TOE correctly accepted the certificate and successfully negotiated the connection. - Pass
Execution Method	Manual

Test Case Number	028
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation

	when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: d) Test 4 [conditional]: The evaluator shall present a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. The evaluator shall verify that the connection succeeds. The evaluator shall repeat this test for each supported SAN type (e.g. IPv4, IPv6, FQDN, SRV).
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install a certificate on the remote server with a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. 2. Begin capturing packets between the TOE and the server. 3. Connect the TOE to the server. 4. Stop capturing packets. 5. Verify the connection succeeds.
Test Results	The TOE correctly accepted the certificate and successfully negotiated the connection. - Pass
Execution Method	Manual

Test Case Number	029
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the

	evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: e) Test 5 [conditional]: The evaluator shall perform the following wildcard tests with each supported type of reference identifier that includes a DNS name (i.e. CN-ID with DNS, DNS-ID, SRV-ID, URIID): 1) [conditional]: The evaluator shall present a server certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.example.com) and verify that the connection fails. 2) [conditional]: The evaluator shall present a server certificate containing a wildcard in the left-most label (e.g. *.example.com). The evaluator shall configure the reference identifier with a single left-most label (e.g. foo.example.com) and verify that the connection succeeds, if wildcards are supported, or fails if wildcards are not supported. The evaluator shall configure the reference identifier without a left-most label as in the certificate (e.g. example.com) and verify that the connection fails. The evaluator shall configure the reference identifier with two left-most labels (e.g. bar.foo.example.com) and verify that the connection fails. (Remark: Support for wildcards was always intended to be optional. It is sufficient to state that the TOE does not support wildcards and observe rejected connection attempts to satisfy corresponding assurance activities.)
Test Instructions	Execute this test per the test steps.
Test Steps	1. Install a certificate on the server containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.catl.local) and specify the reference identifier of the host to be foo.<remote-peer>.catl.local. 2. Begin capturing packets between the TOE and the server. 3. Connect the TOE to the server (e.g. foo.<remote-peer>.catl.local). 4. Stop capturing packets between the TOE and the server with Wireshark. 5. Verify the connection fails. 6. Install a certificate on the server containing a wildcard in the left-most label (e.g. *.catl.local), and specify the reference identifier of the host to be with a single left-most label (e.g. <remote-peer>.catl.local). 7. Using Wireshark, begin capturing packets between the TOE and the server. 8. Connect the TOE to the server. 9. Stop capturing packets between the TOE and the server. 10. Verify the connection succeeds. 11. Repeat Steps 6-9, except in Step 6, configure the reference identifier of the host to catl.local.

	12. Verify that the connection fails. 13. Repeat Steps 6-9, except in Step 6, configure the reference identifier of the host to foo.<remote-peer>.catl.local. 14. Verify that the connection fails. 15. Repeat Steps 1-14 for each supported reference identifier type that includes a DNS name.
Test Results	The TOE correctly accepted the certificate containing a wildcard in the left-most label (e.g. *.catl.local), and the reference identifier of the host was specified in the following format: (e.g. <remote-peer>.catl.local). The TOE correctly rejected every other combination of wildcard tested. - Pass
Execution Method	Manual

Test Case Number	030
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: f) Test 6 [conditional]: If IP addresses are supported, the evaluator shall present a server certificate that contains a CN that matches the reference identifier, except one of the groups has been replaced with an asterisk (*) (e.g. CN=192.168.1.* when connecting to 192.168.1.20, CN=2001:0DB8:0000:0000:0008:0800:200C:* when connecting to

	2001:0DB8:0000:0000:0008:0800:200C:417A). The certificate shall not contain the SAN extension. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each supported IP address version (e.g. IPv4, IPv6). Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass Test 6.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE does not support IP address reference identifiers.
Test Results	Pass
Execution Method	Manual

Test Case Number	031
SFR	FCS_TLSC_EXT.1.2
Test Objective	Note that the following tests are marked conditional and are applicable under the following conditions: a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable. or b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable or c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable. Note that for some tests additional conditions apply. IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986. • IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested. The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: g) Test 7 [conditional]: If the secure channel is used for FPT_ITT, and RFC 5280 is selected, the evaluator shall perform the following tests. Note, when multiple attribute types are selected in the SFR (e.g. when multiple attribute types are combined to form the unique identifier), the evaluator modifies each attribute type

	in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier): 1) The evaluator shall present a server certificate that does not contain an identifier in the Subject (DN) attribute type(s) that matches the reference identifier. The evaluator shall verify that the connection fails. 2) The evaluator shall present a server certificate that contains a valid identifier as an attribute type other than the expected attribute type (e.g. if the TOE is configured to expect id-atserialNumber=correct_identifier, the certificate could instead include id-at-name=correct_identifier), and does not contain the SAN extension. The evaluator shall verify that the connection fails. Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass this test. 3) The evaluator shall present a server certificate that contains a Subject attribute type that matches the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection succeeds. 4) The evaluator shall confirm that all use of wildcards results in connection failure regardless of whether the wildcards are used in the left or right side of the presented identifier. (Remark: Use of wildcards is not addressed within RFC 5280.)
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The ST does not claim FPT_ITT.
Test Results	Pass
Execution Method	Manual

Test Case Number	032
SFR	FCS_TLSC_EXT.1.3
Test Objective	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows: Test 1: Using the administrative guidance, the evaluator shall load a CA certificate or certificates needed to validate the presented certificate used to authenticate an external entity and demonstrate that the function succeeds, and a trusted channel can be established.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the server and the TOE. 2. Initiate a connection from the TOE to the server. 3. Stop capturing packets between the server and the TOE. 4. Verify the connection succeeds.
Test Results	The TOE correctly accepted the certificate and negotiated the connection. - Pass
Execution Method	Manual

Test Case Number	033
SFR	FCS_TLSC_EXT.1.3
Test Objective	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows:

	Test 2: The evaluator shall then change the presented certificate(s) so that validation fails and show that the certificate is not automatically accepted. The evaluator shall repeat this test to cover the selected types of failure defined in the SFR (i.e. the selected ones from failed matching of the reference identifier, failed validation of the certificate path, failed validation of the expiration date, failed determination of the revocation status). The evaluator performs the action indicated in the SFR selection observing the TSF resulting in the expected state for the trusted channel (e.g. trusted channel was established) covering the types of failure for which an override mechanism is defined.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the server and the TOE. 2. Initiate a connection from the TOE to the server. 3. Stop capturing packets between the server and the TOE. 4. Verify the connection fails.
Test Results	The TOE correctly rejected the certificate and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	034
SFR	FCS_TLSC_EXT.1.3
Test Objective	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows: Test 3 [conditional]: The purpose of this test to verify that only selected certificate validation failures could be administratively overridden. If any override mechanism is defined for failed certificate validation, the evaluator shall configure a new presented certificate that does not contain a valid entry in one of the mandatory fields or parameters (e.g. inappropriate value in extendedKeyUsage field) but is otherwise valid and signed by a trusted CA. The evaluator shall confirm that the certificate validation fails (i.e. certificate is rejected), and there is no administrative override available to accept such certificate.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – This conditional test does not apply as the ST states the TSF shall not implement any administrator override mechanism.
Test Results	Pass
Execution Method	Manual

Test Case Number	035
SFR	FCS_TLSC_EXT.1.4
Test Objective	Test 1 [conditional]: If the TOE presents the Supported Elliptic Curves/Supported Groups Extension, the evaluator shall configure the server to perform ECDHE or DHE (as applicable) key exchange using each of the TOE's supported curves and/or groups. The evaluator shall verify that the TOE successfully connects to the server.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE does not present the Supported Elliptic Curves/Supported Groups extension.
Test Results	Pass
Execution Method	Manual

Test Case Number	036
SFR	FCS_TLSS_EXT.1.1
Test Objective	Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Test Instructions	Execute this test per the test steps.
Test Steps	1. The following ciphersuites are configured for use by the TOE Console application: TLS_RSA_WITH_AES_256_CBC_SHA TLS_RSA_WITH_AES_128_CBC_SHA256 TLS_RSA_WITH_AES_256_CBC_SHA256 TLS_RSA_WITH_AES_256_GCM_SHA384 2. Configure the Java Runtime Environment java.security file to use TLS_RSA_WITH_AES_256_CBC_SHA as described in the setup procedures. 3. Using Wireshark, begin capturing packets between the TOE and the test machine. 4. Connect to the TOE via the TOE Console application. 5. Stop capturing packets with Wireshark. 6. Verify the connection succeeded. 7. Repeat Steps 2-6, except iterate through to the next ciphersuite in Step 1.
Test Results	The TOE successfully negotiated a connection with each of the claimed ciphers. - Pass
Execution Method	Manual

Test Case Number	037
SFR	FCS_TLSS_EXT.1.1
Test Objective	Test 2: The evaluator shall send a Client Hello to the server with a list of ciphersuites that does not contain any of the ciphersuites in the server's ST and verify that the server denies the connection. Additionally, the evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the server denies the connection.
Test Instructions	Execute this test per the test steps.
Test Steps	(a) Unsupported ciphersuites: 1. Begin capturing packets between the TOE and the Console. 2. Configure the TLS client to use the specified list of ciphersuites. 3. Initiate a connection between the TOE from the Console. 4. Stop capturing packets. 5. Verify that the TLS connection could not be established. (b) TLS_NULL_WITH_NULL_NULL: 1. Begin capturing packets between the TOE and the TLS client.

	2. Run the tool to send the TLS_NULL_WITH_NULL_NULL ciphersuite in the Client Hello. 3. Initiate a connection between the TOE from the Console. 4. Stop capturing packets. 5. Verify that the TLS connection could not be established and the server refused to negotiate a ciphersuite.
Test Results	The TOE correctly does not negotiate the connections in both instances. - Pass
Execution Method	Manual

Test Case Number	038
SFR	FCS_TLSS_EXT.1.1
Test Objective	Test 3: The evaluator shall perform the following modifications to the traffic: a) Modify a byte in the Client Finished handshake message, and verify that the server rejects the connection and does not send any application data. b) (Test Intent: The intent of this test is to ensure that the server's TLS implementation immediately makes use of the key exchange and authentication algorithms to: a) Correctly encrypt (D)TLS Finished message and b) Encrypt every (D)TLS message after session keys are negotiated.) The evaluator shall use one of the claimed ciphersuites to complete a successful handshake and observe transmission of properly encrypted application data. The evaluator shall verify that no Alert with alert level Fatal (2) messages were sent. The evaluator shall verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. The evaluator shall examine the Finished message (encrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 11 22 33 44 55...) and confirm that it does not contain unencrypted data (unencrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 14 00 00 0c...), by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages. There is a chance that an encrypted Finished message contains a hexadecimal value of '14' at the position where a plaintext Finished message would contain the message type code '14'. If the observed Finished message contains a hexadecimal value of '14' at the position where the plaintext Finished message would contain the message type code, the test shall be repeated three times in total. In case the value of '14' can be observed in all three tests it can be assumed that the Finished message has indeed been sent in plaintext and the test has to be regarded as 'failed'. Otherwise it has to be assumed that the observation of the value '14' has been due to chance and that the Finished message has indeed been sent encrypted. In that latter case the test shall be regarded as 'passed'.
Test Instructions	Execute this test per the test steps.
Test Steps	a) 1. Begin capturing packets between the TOE and the TLS client. 2. Run the tool on the MITM test system by executing the command. 3. Initiate a connection from the TLS client to the TOE. 4. Stop capturing packets. 5. Confirm the TLS connection failed to establish. b)

	1. Open Wireshark and begin capturing packets between the TOE and the TLS client. 2. Initiate a connection from the TLS client to the TOE. 3. Stop capturing packets. 4. Inspect the packet capture for each of the following: a. Verify the Finished message (Encrypted Handshake) is sent immediately after the server's ChangeCipherSpec message. b. Examine the Finished message and confirm it does not contain unencrypted data (by verifying that the first byte of the Finished message does not equal hexadecimal 14.
Test Results	The TOE correctly did not negotiate the connections, the Finished message was sent immediately after the server's ChangCipherSpec message, and there was no unencrypted data as part of the Finished message. - Pass
Execution Method	Manual

Test Case Number	039
SFR	FCS_TLSS_EXT.1.2
Test Objective	The evaluator shall send a Client Hello requesting a connection for all mandatory and selected protocol versions in the SFR (e.g. by enumeration of protocol versions in a test client) and verify that the server denies the connection for each attempt.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and client. 2. Execute the commands on the test machine to initiate a connection to the TOE using the disallowed protocols versions. 3. Stop capturing packets and verify that the connection(s) failed for the mandatory and selected protocol versions in the SFR.
Test Results	The TOE correctly did not negotiate a channel using any disallowed protocols versions. The TOE only supports TLSv1.2. - Pass
Execution Method	Manual

Test Case Number	040
SFR	FCS_TLSS_EXT.1.3
Test Objective	Test 1: [conditional] If ECDHE ciphersuites are supported: a) The evaluator shall repeat this test for each supported elliptic curve. The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single supported elliptic curve specified in the Elliptic Curves Extension. The Evaluator shall verify (though a packet capture or instrumented client) that the TOE selects the same curve in the Server Key Exchange message and successfully establishes the connection. b) The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single unsupported elliptic curve (e.g. secp192r1 (0x13)) specified in RFC4492, chap. 5.1.1. The evaluator shall verify that the TOE does not send a Server Hello message and the connection is not successfully established.
Test Instructions	Execute this test per the test steps.
Test Steps	The TOE does not support ECDHE ciphersuites. Therefore, this test is not applicable.
Test Results	Pass
Execution Method	Manual

Test Case Number	041
SFR	FCS_TLSS_EXT.1.3
Test Objective	Test 2: [conditional] If DHE ciphersuites are supported, the evaluator shall repeat the following test for each supported parameter size. If any configuration is necessary, the evaluator shall configure the TOE to use a supported Diffie-Hellman parameter size. The evaluator shall attempt a connection using a supported DHE ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a Server Key Exchange Message where p Length is consistent with the message are the ones configured Diffie-Hellman parameter size(s).
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE does not support DHE ciphersuites.
Test Results	Pass
Execution Method	Manual

Test Case Number	042
SFR	FCS_TLSS_EXT.1.3
Test Objective	Test 3: [conditional] If RSA key establishment ciphersuites are supported, the evaluator shall repeat this test for each RSA key establishment key size. If any configuration is necessary, the evaluator shall configure the TOE to perform RSA key establishment using a supported key size (e.g. by loading a certificate with the appropriate key size). The evaluator shall attempt a connection using a supported RSA key establishment ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a certificate whose modulus is consistent with the configured RSA key size.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the test machine. 2. Connect to the TOE via the TOE Console application. 3. Stop capturing packets. 4. Examine the packet capture to verify that the TOE sends a certificate whose modulus is consistent with the configured RSA key size.
Test Results	The TOE successfully negotiated the connection when presented a server certificate whose modulus is consistent with the configured RSA key size (2048 bit). - Pass
Execution Method	Manual

Test Case Number	043
SFR	FCS_TLSS_EXT.1.4 – TD0569
Test Objective	Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption). Remark: If multiple contexts are supported for session resumption, the session ID or session ticket may be obtained in one context for resumption in another context. It is possible that one or more contexts may only permit the construction of sessions to be reused in other contexts but not actually permit resumption themselves. For contexts which do not permit resumption, the evaluator is required to verify this behaviour subject to the description provided in the TSS. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.

	Test 1 [conditional]: If the TOE does not support session resumption based on session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2) or session tickets according to RFC5077, the evaluator shall perform the following test: a) The client sends a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket. b) The client verifies the server does not send a NewSessionTicket handshake message (at any point in the handshake). c) The client verifies the Server Hello message contains a zero-length session identifier or passes the following steps: Note: The following steps are only performed if the ServerHello message contains a non-zero length SessionID. d) The client completes the TLS handshake and captures the SessionID from the ServerHello. e) The client sends a ClientHello containing the SessionID captured in step d). This can be done by keeping the TLS session in step d) open or start a new TLS session using the SessionID captured in step d). f) The client verifies the TOE (1) implicitly rejects the SessionID by sending a ServerHello containing a different SessionID and by performing a full handshake (as shown in Figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Begin capturing packets between the TOE and the test machine. 2. Initiate a connection to the TOE by sending a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket. 3. Stop capturing packets between the TOE and the test machine. 4. Confirm that the TOE does not send a NewSessionTicket handshake message (at any point in the handshake). 5. Confirm that the Server Hello message contains a zero-length session identifier; otherwise perform the following steps: a. Capture the SessionID from the Server Hello. b. Send a new Client Hello containing the captured Session ID. c. Verify that the TOE rejects the SessionID by sending a Server Hello with a different SessionID and by performing a full handshake.
Test Results	The TOE correctly rejected the reused SessionID and sent a Server Hello with a different SessionID. - Pass
Execution Method	Manual

Test Case Number	044
SFR	FCS_TLSS_EXT.1.4 – TD0569
Test Objective	Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).

	Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session. Test 2 [conditional]: If the TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2), the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): a) The evaluator shall conduct a successful handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then initiate a new TLS connection and send the previously captured session ID to show that the TOE resumed the previous session by responding with ServerHello containing the same SessionID immediately followed by ChangeCipherSpec and Finished messages (as shown in Figure 2 of RFC 4346 or RFC 5246). b) The evaluator shall initiate a handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then, within the same handshake, generate or force an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message thereby disrupting the handshake. The evaluator shall then initiate a new Client Hello using the previously captured session ID, and verify that the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID and performing a full handshake (as shown in figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The Security Target does not specify that the TOE supports session resumption using session IDs; therefore, this conditional test does not apply.
Test Results	Pass
Execution Method	Manual

Test Case Number	045
SFR	FCS_TLSS_EXT.1.4 – TD0556 & TD0569
Test Objective	Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption). Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session. Test 3 [conditional]: If the TOE supports session tickets according to RFC5077, the

	evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): a) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the session ticket in the ClientHello. The evaluator shall confirm that the TOE responds with an abbreviated handshake described in section 3.1 of RFC 5077 and illustrated with an example in figure 2. Of particular note: if the server successfully verifies the client's ticket, then it may renew the ticket by including a NewSessionTicket handshake message after the ServerHello in the abbreviated handshake (which is shown in figure 2). This is not required, however as further clarified in section 3.3 of RFC 5077. b) The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator will then modify the session ticket and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake (as shown in figure 3 or 4 of RFC 5077), or (2) terminates the connection in some way that prevents the flow of application data.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The Security Target does not specify that the TOE supports session resumption using session tickets; therefore, this conditional test does not apply.
Test Results	Pass
Execution Method	Manual

4.3.3 Identification and Authentication

Test Case Number	046
SFR	FIA_AFL.1
Test Objective	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application): a) Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE (and, if the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall also use the operational guidance to configure the time period after which access is re-enabled). The evaluator shall test that once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful.
Test Instructions	Execute this test per the test steps.
Test Steps	**[Remote Console interface]** **Configuration steps:** 1. Authenticate to the TOE via the Console using the administrator account. 2. Set the number of successive unsuccessful authentication attempts: a. Navigate to “Tools” > “Options” > “CounterACT User Profiles” > “Password and Sessions”. b. Select the “Password” tab. c. Check the box, “Lock user account after” and set the inputs such that they read: “3” for failed logins for “5” minutes. d. Select “Apply” and “OK” in the confirmation window. 3. Log out of the TOE.

Auto-unlock:

1. Attempt to authenticate to the TOE via the Console using an invalid password 3 times successively.
2. Attempt to authenticate to the TOE via the Console using valid credentials.
3. Verify the 4th authentication attempt fails due to account lockout.
4. Wait until near the 5-minute mark since account lockout, but not exceeding 5 minutes.
5. Attempt to authenticate to the TOE via the Console using valid credentials.
6. Verify the 5th authentication attempt fails due to account lockout.
7. Wait until just after 5 minutes since the lockout has elapsed and attempt to authenticate to the TOE via the Console using valid credentials.
8. Verify that this authentication attempt succeeds.

Manual unlock:

1. Attempt to authenticate to the TOE via the Console using an invalid password 3 times.
2. Attempt to authenticate to the TOE via the Console using valid credentials.
3. Verify the 4th authentication attempt fails due to account lockout.
4. Authenticate to the TOE via the CLI.
5. Execute the following command to unlock the locked account:

```
fstool unlock_console_user <account>
```

6. Attempt to login using the originally locked administrator account and verify valid credentials result in successful authentication.

[Remote CLI interface]

Configuration steps:

1. Authenticate to the TOE via the Console using the administrator account.
2. Set the number of successive unsuccessful authentication attempts:
    a. Navigate to “Tools” > “Options” > “CounterACT User Profiles” > “Password and Sessions”.
    b. Select the “Password” tab.
    c. Check the box, “Lock user account after” and set the inputs such that they read: “3” for failed logins for “5” minutes. [NOTE: The time value specified for the lockout period is only applicable to the “Console” interface and has no effect for this test against the “remote CLI” interface. The lockout period for the “remote CLI” interface is configured in the “CLI configuration steps” section.]
    d. Select “Apply” and “OK” in the confirmation window.
3. Authenticate to the TOE via the SSH CLI using the administrator account.
4. Issue the followings commands to set the remote CLI interface lockout period to 4 minutes:

	`` fstool set_property os.lockout.fail 240 fstool service restart ``` 5. Wait until the service is running. **Auto-unlock:** 1. Attempt to authenticate to the TOE via SSH using an invalid password 3 times successively. 2. Attempt to authenticate to the TOE via SSH using valid credentials. 3. Verify the 4th authentication attempt fails due to account lockout. 4. Wait until near the 4-minute mark since account lockout, but not exceeding 4 minutes. 5. Attempt to authenticate to the TOE via SSH using valid credentials. 6. Verify the 5th authentication attempt fails due to account lockout. 7. Wait until just after the 4-minute mark since the account was locked out and attempt to authenticate to the TOE via the SSH using valid credentials. 8. Verify that this authentication attempt succeeds. **Manual unlock:** 1. Attempt to authenticate to the TOE via SSH using an invalid password 3 times successively. 2. Attempt to authenticate to the TOE via SSH using a valid password. 3. Verify the 4th authentication attempt fails due to account lockout. 4. Authenticate to the TOE via the CLI using a different administrator account. 5. Enter the following commands to validate account is locked as indicated by the locked-out account appearing 3 times in the presented table: ``` user faillock list ``` 6. Enter the following command to unlock account: ``` user faillock reset <locked username> ``` 7. Log out of the TOE. 8. Attempt to login using the previously locked administrator account using valid credentials. 9. Verify that the authentication attempt was successful.
Test Results	Security Administrator accounts that attempt to authenticate via the remote interfaces (remote CLI, remote Console) were locked out for the configured time period or until a manual unlock action occurs when consecutive authentication failure attempts reach the configured limit for that interface. - Pass
Execution Method	Manual

Test Case Number	047
SFR	FIA_AFL.1

Test Objective	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application): b) Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows. If the administrator action selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall confirm by testing that following the operational guidance and performing each action specified in the ST to re-enable the remote administrator's access results in successful access (when using valid credentials for that administrator). If the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall wait for just less than the time period configured in Test 1 and show that an authorisation attempt using valid credentials does not result in successful access. The evaluator shall then wait until just after the time period configured in Test 1 and show that an authorisation attempt using valid credentials results in successful access.
Test Instructions	Execute this test per the test steps.
Test Steps	This assurance activity is performed in conjunction with FIA_AFL.1 - Test 1 (Test Case 046).
Test Results	Pass
Execution Method	Manual

Test Case Number	050
SFR	FIA_PMG_EXT.1
Test Objective	The evaluator shall perform the following tests. a) Test 1: The evaluator shall compose passwords that meet the requirements in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that all characters, and a minimum length listed in the requirement are supported and justify the subset of those characters chosen for testing.
Test Instructions	Execute this test per the test steps.
Test Steps	**GUI Console:** 1. Authenticate to the TOE. 2. Select "Change Password" from the "Tools" menu. 3. Specify the following for the new password: abcdefghijklm123!@# 4. Select "OK" 5. Log out of the TOE. 6. Authenticate to the TOE using the password created in Step 3. 7. Repeat Steps 1-6 except change the password to: nopqrstuvwxyz456$%^ 8. Repeat steps 1-6 except change the password to: ABCDEFGHIJKLM789&*(

	9. Repeat steps 1-6 except change the password to: NOPQRSTUVWXYZ0) **CLI:** 10. Authenticate to the TOE via SSH. 11. Enter the command "password" to change the password for the "cliadmin" 12. Specify the following for the new password: thequickfoxjumps123!@# 13. Log out of the TOE. 14. Authenticate to the TOE via ssh CLI using the password created in Step 12. 15. Repeat Steps 12-16 except change the password to: overthelazybrowndog456$%^ 16. Repeat steps 12-16 except change the password to: THEQUICKFOXJUMPS7890&*() 17. Repeat steps 12-16 except change the password to: OVERTHELAZYBROWNDOG1!
Test Results	Attempts to change the password to values compliant with the password length requirement of at least 15 characters and containing all of the claimed characters were successful. - Pass
Execution Method	Manual

Test Case Number	051
SFR	FIA_PMG_EXT.1
Test Objective	The evaluator shall perform the following tests. b) Test 2: The evaluator shall compose passwords that do not meet the requirements in some way. For each password, the evaluator shall verify that the TOE does not support the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that the TOE enforces the allowed characters and the minimum length listed in the requirement and justify the subset of those characters chosen for testing.
Test Instructions	Execute this test per the test steps.
Test Steps	**GUI Console:** 1. Authenticate to the TOE. 2. Select "Change Password" from the "Tools" menu. 3. Attempt to change the password to "abcdefghijklmn". 4. The password change should fail because the password is fewer than 15 characters. 5. Repeat Steps 2-4, except in Step 3, attempt to change the password to "abcdef".

	CLI: 6. Authenticate to the TOE via SSH. 7. Enter the command “password” to change the password for the “cliadmin” 8. Attempt to change the password to “abcdefghijklmn”. 9. The password should be rejected because it is fewer than 15 characters. 10. Repeat Steps 6-9, except in Step 8, attempt to change the password to “abcdef”.
Test Results	Attempts to change the password to values less than 15 characters in length were unsuccessful. - Pass
Execution Method	Manual

Test Case Number	052
SFR	FIA_UAU.7
Test Objective	The evaluator shall perform the following test for each method of local login allowed: a) Test 1: The evaluator shall locally authenticate to the TOE. While making this attempt, the evaluator shall verify that at most obscured feedback is provided while entering the authentication information.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Login to the TOE via the local CLI using an administrator username and valid password. 2. Ensure the password is obscured (not shown on the screen) and that no feedback is provided.
Test Results	The TOE did not echo back the characters as the password was entered. - Pass
Execution Method	Manual

Test Case Number	053
SFR	FIA_UIA_EXT.1
Test Objective	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: a) Test 1: The evaluator shall use the guidance documentation to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct I&A information results in the ability to access the system, while providing incorrect information results in denial of access.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Attempt to authenticate to the TOE via SSH. 2. Specify a valid local username and valid password. 3. Verify that the authentication was successful. 4. Log out of the TOE. 5. Attempt to authenticate to the TOE via SSH. 6. Specify a valid local username and an invalid password. 7. Verify that the authentication attempt failed. 8. Attempt to authenticate to the TOE via SSH. 9. Specify an invalid local username and a valid password. 10. Verify that the authentication attempt failed.

	11. Attempt to authenticate to the TOE via SSH. 12. Specify an invalid local username and an invalid password. 13. Verify that the authentication attempt failed. 14. Repeat Steps 1-13, except authenticate via the local console. 15. Repeat Steps 1-13, except authenticate via the Console. 16. Repeat Steps 1-13, except authenticate via the Console and use Active Directory credentials. 17. Generate a public/private key-pair and ensure the public key is installed on the TOE and associated with a user account. 18. Attempt to login to the TOE using the corresponding private key via SSH public key authentication. 19. Verify that the authentication is successful. 20. Log out of the TOE. 21. Attempt to login to the TOE using a valid username and an invalid private key via SSH public key authentication. 22. Verify that the authentication attempt failed. 23. Attempt to login to the TOE using an invalid username and valid private key via SSH public key authentication. 24. Verify that the authentication attempt failed 25. Attempt to login to the TOE using an invalid username and invalid username and valid private key via SSH public key authentication. 26. Verify that the authentication attempt failed
Test Results	The TOE behaved correctly for all credential combinations for all interface/credential store combinations. The TOE also produced the correct audit records with the correct details - Pass
Execution Method	Manual

Test Case Number	054
SFR	FIA_UIA_EXT.1
Test Objective	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: b) Test 2: The evaluator shall configure the services allowed (if any) according to the guidance documentation, and then determine the services available to an external remote entity. The evaluator shall determine that the list of services available is limited to those specified in the requirement.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Attempt to authenticate to the TOE via SSH. 2. Specify “fstool ssh” as the username and password. 3. Verify that access is appropriately denied and that the only service available prior to successful authentication is the FTA_TAB.1 warning banner. 4. Attempt to authenticate to the TOE via SSH using public key authentication. 5. Specify “fstool ssh” as the username. 6. Verify that access is appropriately denied and that the only service available prior to successful authentication is the FTA_TAB.1 warning

	banner. 7. Attempt to authenticate to the TOE via the Console using local and Active Directory credentials. 8. Verify that the only service available prior to authentication is the FTA_TAB.1 warning banner.
Test Results	Attempt to login with invalid credentials were unsuccessful for both SSH and GUI Console. The only item available prior to authentication on the system was the warning banner. - Pass
Execution Method	Manual

Test Case Number	055
SFR	FIA_UIA_EXT.1
Test Objective	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: c) Test 3: For local access, the evaluator shall determine what services are available to a local administrator prior to logging in, and make sure this list is consistent with the requirement.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Attempt to authenticate to the TOE via the local console. 2. Specify “fstool ssh” as the username and password. 3. Verify that access is appropriately denied and that the only service available prior to authentication is the FTA_TAB.1 warning banner.
Test Results	The warning banner was successfully configured and it was displayed for local CLI. For local CLI access, only local credential store is used. - Pass
Execution Method	Manual

Test Case Number	056
SFR	FIA_UIA_EXT.1
Test Objective	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: d) Test 4: For distributed TOEs where not all TOE components support the authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2, the evaluator shall test that the components authenticate Security Administrators as described in the TSS.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE is not a distributed TOE.
Test Results	Pass
Execution Method	Manual

Test Case Number	057
SFR	FIA_UIA_EXT.2
Test Objective	Evaluation Activities for this requirement are covered under those for FIA_UIA_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA_UIA_EXT.1.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – Per the assurance activity, evaluation activities for this requirement are

	covered under those for FIA_UIA_EXT.1
Test Results	Pass
Execution Method	Manual

Test Case Number	058
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: a) Test 1a: The evaluator shall present the TOE with a valid chain of certificates (terminating in a trusted CA certificate) as needed to validate the leaf certificate to be used in the function and shall use this chain to demonstrate that the function succeeds. Test 1a shall be designed in a way that the chain can be 'broken' in Test 1b by either being able to remove the trust anchor from the TOEs trust store, or by setting up the trust store in a way that at least one intermediate CA certificate needs to be provided, together with the leaf certificate from outside the TOE, to complete the chain (e.g. by storing only the root CA certificate in the trust store). Test 1b: The evaluator shall then 'break' the chain used in Test 1a by either removing the trust anchor in the TOE's trust store used to terminate the chain, or by removing one of the intermediate CA certificates (provided together with the leaf certificate in Test 1a) to complete the chain. The evaluator shall show that an attempt to validate this broken chain fails.
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Create and install a server certificate which chains to the root CA, intermediate01, and intermediate02 certificates on the remote server. 2. Begin capturing packets between the server and the TOE. 3. Initiate a connection from the TOE to the server. 4. Stop capturing packets between the server and the TOE. 5. Verify connection was successful. 6. Remove the root CA certificate from the TOE’s certificate authority trust store. 7. Repeat Steps 2-4. 8. Verify connection was not successful due to missing CA certificate.
Test Results	The TOE successfully negotiated the connection when the root CA was present. The TOE correctly did not negotiate the connection when the root CA was removed. - Pass
Execution Method	Manual

Test Case Number	059
SFR	FIA_X509_EXT.1.1/Rev

Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: b) Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Begin capturing packets between the server and the TOE. 2. Initiate a connection from the TOE to the server. 3. Stop capturing packets between the server and the TOE. 4. Verify connection was not successful due to missing expired certificate.
Test Results	The TOE correctly found that the presented certificate was expired and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	060
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: c) Test 3: The evaluator shall test that the TOE can properly handle revoked certificates-–conditional on whether CRL or OCSP is selected; if both are selected, then a test shall be performed for each method. The evaluator shall test revocation of the peer certificate and revocation of the peer intermediate CA certificate i.e. the intermediate CA certificate should be revoked by the root CA. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails. Revocation checking is only applied to certificates that are not designated as trust anchors. Therefore, the revoked certificate(s) used for testing shall not be a trust anchor.
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** **OCSP** 1. Load a valid server certificate onto the server.

	2. Start a packet capture between the TOE and the remote server and the OCSP server. 3. Create a connection between the TOE and the server. 4. Stop the packet capture and verify the certificates were verified by the OCSP server. 5. Verify the connection succeeds. 6. Load a revoked certificate onto the server. 7. Start a packet capture between the TOE and the remote server and the OCSP server. 8. Create a connection between the TOE and the Server. 9. Stop the packet capture and verify the certificate was revoked by the OCSP server. 10. Verify the connection fails due to the revoked peer certificate. 11. Load a valid server certificate onto the server. 12. Revoke the intermediate 01 CA certificate in the presented chain. 13. Start a packet capture between the TOE and the remote server and the OCSP server 14. Create a connection between the TOE and the server. 15. Stop the packet capture and verify the certificate was revoked by the OCSP server. 16. Verify the connection fails due to the revoked intermediate 01 CA certificate.
Test Results	The TOE correctly found that the presented certificate was revoked and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	061
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: d) Test 4: If OCSP is selected, the evaluator shall configure the OCSP server or use a man-in-the-middle tool to present a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** **OCSP** 1. Load a certificate on the OCSP server without the OCSP signing purpose.

	2. Begin capturing packets between the TOE, the remote server, and the OCSP responder. 3. Cause the TOE to establish a connection to the remote server. 4. Stop capturing packets. 5. Verify the connection fails because the OCSP response could not be validated.
Test Results	The TOE correctly found that the presented certificate did not contain the OCSP signing purpose and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	062
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: e) Test 5: The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Begin capturing packets between the TOE and the environmental entity. 2. Run the tool on the MITM test system by executing the command. 3. Cause the TOE to initiate a connection to the environmental entity. 4. Stop capturing packets between the TOE and the environmental entity. 5. Verify the connection fails because the certificate will fail to parse correctly.
Test Results	The TOE correctly found that the presented certificate did not parse correctly and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	063
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:

	f) Test 6: The evaluator shall modify any byte in the certificate signatureValue field (see RFC5280 Sec. 4.1.1.3), which is normally the last field in the certificate, and demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Begin capturing packets between the TOE and the environmental entity. 2. Run the tool on the MITM test system by executing the command. 3. Cause the TOE to initiate a connection to the environmental entity. 4. Stop capturing packets between the TOE and the environmental entity. 5. The connection will fail because the certificate will fail to parse correctly.
Test Results	The TOE correctly found that the presented certificate did not parse correctly and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	064
SFR	FIA_X509_EXT.1.1/Rev
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: g) Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The hash of the certificate will not validate.)
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Begin capturing packets between the TOE and the environmental entity. 2. Run the tool on the MITM test system by executing the command. 3. Cause the TOE to initiate a connection to the environmental entity. 4. Stop capturing packets between the TOE and the environmental entity. 5. The connection will fail because the certificate will fail to parse correctly.
Test Results	The TOE correctly found that the presented certificate did not parse correctly and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	065
SFR	FIA_X509_EXT.1.1/Rev – TD0527
Test Objective	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. It is not necessary

	to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: The following tests are run when a minimum certificate path length of three certificates is implemented. Test 8: (Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen). The evaluator shall conduct the following tests: Test 8a: (Conditional on TOE ability to process CA certificates presented in certificate message) The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a valid chain of EC certificates (terminating in a trusted CA certificate), where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain. Test 8b: (Conditional on TOE ability to process CA certificates presented in certificate message) The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a chain of EC certificates (terminating in a trusted CA certificate), where the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key information field, and is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid. Test 8c: The evaluator shall establish a subordinate CA certificate, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is accepted into the TOE's trust store. The evaluator shall then establish a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is rejected, and not added to the TOE's trust store.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – Support for EC certificates is not indicated in FCS_COP.1/SigGen.
Test Results	Pass
Execution Method	Manual

Test Case Number	066
SFR	FIA_X509_EXT.1/Rev
Test Objective	The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that

	require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests is to verify that the TOE accepts a certificate as a CA certificate only if it has been marked as a CA certificate by using basicConstraints with the CA flag set to True (and implicitly tests that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: a self-signed root CA certificate, an intermediate CA certificate and a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). a) Test 1: The evaluator shall ensure that at least one of the CAs in the chain does not contain the basicConstraints extension. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: (i) as part of the validation of the leaf certificate belonging to this chain; (ii) when attempting to add a CA certificate without the basicConstraints extension to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains). The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection is distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to repeat the tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Configure the server to present an otherwise valid intermediate02 CA certificate with one that does not contain the basicConstraints extension to the TOE. 2. Attempt to establish a connection to the remote server from the TOE. 3. Verify the connection attempt fails.
Test Results	The TOE correctly found that the presented certificate did not have the basicConstraints extension and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	067
SFR	FIA_X509_EXT.1/Rev
Test Objective	The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then

	the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests is to verify that the TOE accepts a certificate as a CA certificate only if it has been marked as a CA certificate by using basicConstraints with the CA flag set to True (and implicitly tests that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: a self-signed root CA certificate, an intermediate CA certificate and a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). b) Test 2: The evaluator shall ensure that at least one of the CA certificates in the chain has a basicConstraints extension in which the CA flag is set to FALSE. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: (i) as part of the validation of the leaf certificate belonging to this chain; (ii) when attempting to add a CA certificate with the CA flag set to FALSE to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains). The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection is distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to repeat the tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).
Test Instructions	Execute this test per the test steps.
Test Steps	**TOE acting as a TLS Client connecting to a Server (syslog, Active Directory)** 1. Configure the server to present an otherwise valid intermediate02 CA certificate with one that has the CA flag set to FALSE in the basicConstraints extension to the TOE. 2. Attempt to establish a connection to the remote server from the TOE. 3. Verify the connection attempt fails.
Test Results	The TOE correctly found that the presented certificate had the basicConstraints extension set to false and did not negotiate the connection. - Pass
Execution Method	Manual

Test Case Number	068
SFR	FIA_X509_EXT.2
Test Objective	The evaluator shall perform the following test for each trusted channel: The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the guidance documentation to determine that all supported administrator-configurable options behave in their documented manner.
Test Instructions	Execute this test per the test steps.

Test Steps	**OCSP** 1. Ensure that the OCSP responder is running. 2. Begin capturing packets between the TOE and the remote server, and between the TOE and the OCSP responder. 3. Perform some activity on the TOE that causes the TOE to check the validation of the certificate. 4. Stop capturing packets. 5. Verify that the TOE accepts the certificate. 6. Disconnect the OCSP responder. 7. Repeat Steps 1-4. 8. Verify that the TOE denies the certificate.
Test Results	The TOE correctly did not negotiate the connection when the presented certificate's revocation status could not be verified. - Pass
Execution Method	Manual

Test Case Number	069
SFR	FIA_X509_EXT.3
Test Objective	The evaluator shall perform the following tests: a) Test 1: The evaluator shall use the guidance documentation to cause the TOE to generate a Certification Request. The evaluator shall capture the generated message and ensure that it conforms to the format specified. The evaluator shall confirm that the Certification Request provides the public key and other required information, including any necessary user-input information.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE Console as the administrator. 2. Navigate to "Tools" > "Options" > "Certificates" > "System Certificates" 3. On the right of the screen click "Generate CSR" Common Name <TOE-hostname.catl.local> Organization – BAH Organizational Unit – CCTL Locality – Laurel State – Maryland Country Code – US Email Address – cctl@catl.local Key Length – RSA-2048 Signature Algorithm – SHA-256 Validity - 3 years 4. Click "Next" 5. When the CSR is generated, scroll down to ensure the public key and common name are present. 6. Copy the CSR to the test machine and execute the command on a test machine to validate the CSR.
Test Results	The TOE successfully generated a CSR with the supplied inputs. The test machine command successfully validated the CSR generated by the TOE by confirming that the inputs in the CSR matched the expected values. - Pass
Execution Method	Manual

Test Case Number	070
SFR	FIA_X509_EXT.3
Test Objective	The evaluator shall perform the following tests: Test 2: The evaluator shall demonstrate that validating a response message to a Certification Request without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the certificate response message and demonstrate that the function succeeds.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Perform FIA_X509_EXT.3 – Test Case 069. 2. Sign the CSR generated in the previous step. 3. Append a CA certificate that does not chain to the signed certificate file from Step 2. 4. On the "Manage System Certificate" window, complete the wizard to install the signed CSR from Step 2 and invalid CA certificate from Step 3. 5. Verify that the TOE fails to validate the certificate because the certificate path could not be verified. 6. Repeat Steps 1 - 2. 7. Append all required CAs up to the root CA that form the complete issuer chain to the signed certificate file from Step 2. 8. Repeat Step 5. 9. Verify that the certificate import was successful.
Test Results	An invalid certificate response message (signed certificate with invalid CA) fails to validate. A valid certificate response message (signed certificate with valid CA) successfully validates. - Pass
Execution Method	Manual

4.3.4 Security Management

Test Case Number	071
SFR	FMT_MOF.1/ManualUpdate
Test Objective	The evaluator shall try to perform the update using a legitimate update image without prior authentication as Security Administrator (either by authentication as a user with no administrator privileges or without user authentication at all – depending on the configuration of the TOE). The attempt to update the TOE shall fail. The evaluator shall try to perform the update with prior authentication as Security Administrator using a legitimate update image. This attempt should be successful. This test case should be covered by the tests for FPT_TUD_EXT.1 already.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console using a non-security administrator account. 2. Attempt to update the TOE using a legitimate update image: a. If "Enterprise Manager": i. Navigate to "Tools" > "Options" > "CounterACT Devices". b. If "Appliance":

	i. Navigate to “Tools” > “Options” > “Appliance”. 3. Verify that the options to perform the update are not available. 4. For the successful attempt to update, please see FPT_TUD_EXT.1 Test Case 078.
Test Results	The evaluator opted to use the "authentication as a user with no administrator privileges" to perform this test. The evidence shows that a non-priviledged user does not have the ability to perform these actions as the buttons are not active. - Pass
Execution Method	Manual

Test Case Number	072
SFR	FMT_MTD.1/CoreData
Test Objective	No separate testing for FMT_MTD.1/CoreData is required unless one of the management functions has not already been exercised under any other SFR.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console using a non-security administrator account. 2. Attempt to update the TOE using a legitimate update image: Navigate to “Tools” > “Options” > “Certificates” > “Trusted Certificates”. 3. Verify that the options “Add”, “Edit”, or “Remove” are not available.
Test Results	Pass
Execution Method	Manual

Test Case Number	073
SFR	FMT_SMF.1
Test Objective	The evaluator tests management functions as part of testing the SFRs identified in section 2.4.4. No separate testing for FMT_SMF.1 is required unless one of the management functions in FMT_SMF.1.1 has not already been exercised under any other SFR.
Test Instructions	Execute this test per the test steps.
Test Steps	**This test is satisfied by testing performing throughout the other test assurance activities.** Ability to administer the TOE locally and remotely - Tested: FTP_TRP.1/Admin - Test Case 090 Ability to configure the access banner - Tested: FTA_TAB.1 - Test Case 085 Ability to configure the session inactivity time before session termination or locking - Tested: FTA_SSL.3 - Test Case 082 Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates - Tested: FPT_TUD_EXT.1 - Test Cases 078 and 079 Ability to configure the authentication failure parameters for FIA_AFL.1 - Tested: FIA_AFL.1 - Test Case 046 Ability to configure thresholds for SSH rekeying - Tested: FCS_SSHS_EXT.1.8 - Test Case 019 Ability to modify the behaviour of the transmission of audit data to an external IT entity - Tested: FAU_STG_EXT.1 - Test Case 003 Ability to re-enable an Administrator account - Tested: FIA_AFL.1 - Test Case 046 Ability to set the time which is used for time-stamps - Tested: FPT_STM_EXT.1 - Test Case 075 Ability to manage the trusted public keys database – Tested: FCS_SSH_EXT.1.2 Test Case 007 and 008. Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors - Tested: FIA_X509_EXT.1.1/Rev - Test Case 058
Test Results	Pass
Execution Method	Manual

Test Case Number	074
SFR	FMT_SMR.2
Test Objective	In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test

	involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this cPP be tested; for instance, if the TOE can be administered through a local hardware interface; SSH; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.
Test Instructions	Execute this test per the test steps.
Test Steps	This test is satisfied by testing performing throughout the other test assurance activities.
Test Results	Pass
Execution Method	Manual

4.3.5 Protection of the TSF

Test Case Number	075
SFR	FPT_STM_EXT.1
Test Objective	The evaluator shall perform the following tests: a) Test 1: If the TOE supports direct setting of the time by the Security Administrator then the evaluator uses the guidance documentation to set the time. The evaluator shall then use an available interface to observe that the time was set correctly. If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the CLI. 2. Execute the following commands to set the date and time: `date -s "YYYY-MM-DD HH:MM:SS"` `date --hwclock` 3. Repeat Step 1. 4. Execute the following command to verify that the date and time was set to that specified in Step 2: `date`
Test Results	The TOE's clock was successfully configured to the specified value. - Pass
Execution Method	Manual

Test Case Number	076
SFR	FPT_STM_EXT.1
Test Objective	The evaluator shall perform the following tests: b) Test 2: If the TOE supports the use of an NTP server; the evaluator shall use the guidance documentation to configure the NTP client on the TOE and set up a communication path with the NTP server. The evaluator will observe that the NTP server has set the time to what is expected. If the TOE supports multiple protocols for establishing a connection with the NTP server, the evaluator shall perform this

	test using each supported protocol claimed in the guidance documentation. If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
Test Instructions	Execute this test per the test steps.
Test Steps	The TOE does not claim use of NTP server; therefore, this test does not apply.
Test Results	Pass
Execution Method	Manual

Test Case Number	077
SFR	FPT_TST_EXT.1
Test Objective	It is expected that at least the following tests are performed: a) Verification of the integrity of the firmware and executable software of the TOE b) Verification of the correct operation of the cryptographic functions necessary to fulfil any of the SFRs. Although formal compliance is not mandated, the self-tests performed should aim for a level of confidence comparable to: a) [FIPS 140-2], chap. 4.9.1, Software/firmware integrity test for the verification of the integrity of the firmware and executable software. Note that the testing is not restricted to the cryptographic functions of the TOE. b) [FIPS 140-2], chap. 4.9.1, Cryptographic algorithm test for the verification of the correct operation of cryptographic functions. Alternatively, national requirements of any CCRA member state for the security evaluation of cryptographic functions should be considered as appropriate. The evaluator shall either verify that the self-tests described above are carried out during initial start-up or that the developer has justified any deviation from this. For distributed TOEs the evaluator shall perform testing of self-tests on all TOE components according to the description in the TSS about which self-test are performed by which component.
Test Instructions	Execute this test per the test steps.
Test Steps	**Self-tests during startup:** 1. Power off the TOE. 2. Power on the TOE. 3. Authenticate to the TOE via the CLI. 4. Execute the following command: `service status` 5. Verify that the TOE reports self-tests are being performed after executing Step 4.

	Manual initiation of self-tests: 6. Authenticate to the TOE via the CLI. 7. Execute the following command to: `selftest` 8. Enter cliadmin password.
Test Results	The TOE successfully performed power-on self-tests (POST) in addition to manually initiated self-tests.- Pass
Execution Method	Manual

Test Case Number	078
SFR	FPT_TUD_EXT.1
Test Objective	The evaluator shall perform the following tests: a) Test 1: The evaluator performs the version verification activity to determine the current version of the product. If a trusted update can be installed on the TOE with a delayed activation, the evaluator shall also query the most recently installed version (for this test the TOE shall be in a state where these two versions match). The evaluator obtains a legitimate update using procedures described in the guidance documentation and verifies that it is successfully installed on the TOE. For some TOEs loading the update onto the TOE and activation of the update are separate steps ('activation' could be performed e.g. by a distinct activation step or by rebooting the device). In that case the evaluator verifies after loading the update onto the TOE but before activation of the update that the current version of the product did not change but the most recently installed version has changed to the new product version. After the update, the evaluator performs the version verification activity again to verify the version correctly corresponds to that of the update and that current version of the product and most recently installed version match again. If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console. 2. Obtain the current version of the TOE: a. At the top menu bar click the "Help" drop down menu then click "About Forescout". 3. Manually obtain and install a legitimate update onto the TOE by performing the following steps: a. If "Enterprise Manager": i. Navigate to "Tools" > "Options" > "CounterACT Devices". b. If "Appliance":

	i. Navigate to “Tools” > “Options” > “Appliance”. c. Click “Upgrade”. d. Navigate to the directory where the update is at. Specify the update file and click “Install”. e. Proceed through the dialog boxes to install the update. 4. After the installation of the update, check the current version of the TOE and the most recently installed version of the TOE software and verify that they correspond to that of the update.
Test Results	The TOE successfully updated to the newer software version after the update was applied. The TOE's version verification activity confirmed the version increased as compared to the version reported prior to the update.- Pass
Execution Method	Manual

Test Case Number	079
SFR	FPT_TUD_EXT.1
Test Objective	The evaluator shall perform the following tests: b) Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 1) A modified version (e.g. using a hex editor) of a legitimately signed update 2) An image that has not been signed 3) An image signed with an invalid signature (e.g. by using a different key as expected for creating the signature or by manual modification of a legitimate signature) 4) If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped.

	The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console. 2. Obtain the current version of the TOE as well as the most recently installed version of the software on the TOE. 3. Manually obtain and install an illegitimate update onto the TOE by performing the following steps: a. Navigate to "Tools" > "CounterACT Devices" b. Click "Upgrade". c. Navigate to the directory where the update is at. Specify the update file and click "Install". 4. Verify that the TOE rejects the illegitimate update. 5. Verify that the TOE current version and TOE most recently installed version, reflect the same version information as prior to the update attempt. 6. Repeat Steps 1-5, except in Step 3, use an image that has not been signed. 7. Repeat Steps 1-5, except in Step 3, use an image signed with an invalid signature.
Test Results	The TOE correctly failed to update when invalid updates (modified binary via hex edit, missing signature, modified signature) were presented to the TOE. The TOE's version prior to the update attempts remained the same after the update attempts. - Pass
Execution Method	Manual

Test Case Number	080
SFR	FPT_TUD_EXT.1
Test Objective	The evaluator shall perform the following tests: c) Test 3 [conditional]: If the TOE itself verifies a hash value over an image against a published hash value (i.e. reference value) that has been imported to the TOE from outside such that the TOE itself authorizes the installation of an image to update the TOE, the following test shall be performed (otherwise the test shall be omitted. If the published hash is provided to the TOE by the Security Administrator and the verification of the hash value over the update file(s) against the published hash is performed by the TOE, then the evaluator shall perform the following tests. The evaluator first confirms that no update is pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. 1) The evaluator obtains or produces an illegitimate update such that the hash of the update does not match the published hash. The evaluator provides the published hash value to the TOE and calculates the hash of the update either on the TOE itself (if that functionality is provided by the TOE), or else outside the TOE. The evaluator confirms that the hash values are different, and attempts to install the update on the TOE,

	verifying that this fails because of the difference in hash values (and that the failure is logged). Depending on the implementation of the TOE, the TOE might not allow the Security Administrator to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE 2) The evaluator uses a legitimate update and tries to perform verification of the hash value without providing the published hash value to the TOE. The evaluator confirms that this attempt fails. Depending on the implementation of the TOE it might not be possible to attempt the verification of the hash value without providing a hash value to the TOE, e.g. if the hash value needs to be handed over to the TOE as a parameter in a command line message and the syntax check of the command prevents the execution of the command without providing a hash value. In that case the mechanism that prevents the execution of this check shall be tested accordingly, e.g. that the syntax check rejects the command without providing a hash value, and the rejection of the attempt is regarded as sufficient verification of the correct behaviour of the TOE in failing to verify the hash. The evaluator then attempts to install the update on the TOE (in spite of the unsuccessful hash verification) and confirms that this fails. Depending on the implementation of the TOE, the TOE might not allow to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE 3) If the TOE allows delayed activation of updates, the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs. Depending on the point in time when the attempted update is rejected, the most recently installed version might or might not be updated. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Instructions	Execute this test per the test steps.
Test Steps	N/A – The TOE does not use a published hash mechanism to validate trusted updates.
Test Results	Pass
Execution Method	Manual

4.3.6 Protection of the TSF

Test Case Number	081
SFR	FTA_SSL_EXT.1
Test Objective	The evaluator shall perform the following test: a) Test 1: The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a local interactive session with the TOE. The evaluator then observes that the session is either locked or terminated after the configured time period. If locking was selected from the component, the evaluator then ensures that re-authentication is needed when trying to unlock the session.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console using the "admin" account. 2. Set the inactivity timeout value: a. Navigate to "Tools" > "Options" > "CounterACT User Profiles" > "Password and Sessions". b. Select the "Session" tab. c. Check the box, "Terminate inactive sessions after," and set the value to 3 minutes. d. Select "Apply". 3. Log out of the TOE. 4. Authenticate to the TOE via the local console. 5. Wait for the configured number of minutes to elapse and verify that the session is terminated. 6. Repeat Steps 1-5, except specify the inactivity time to 4 minutes. 7. Repeat Steps 1-5, except specify the inactivity.
Test Results	For each configured inactivity timeout value, the TOE successfully terminates the session and required a new login to gain access back to the TOE. - Pass
Execution Method	Manual

Test Case Number	082
SFR	FTA_SSL.3
Test Objective	For each method of remote administration, the evaluator shall perform the following test: a) Test 1: The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a remote interactive session with the TOE. The evaluator then observes that the session is terminated after the configured time period.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console using the administrator account. 2. Set the inactivity timeout value: a. Navigate to "Tools" > "Options" > "CounterACT User Profiles" > "Password and Sessions". b. Select the "Session" tab. c. Check the box, "Terminate inactive sessions after," and set the value to 3 minutes.

	d. Select "Apply". 3. Log out of the TOE. 4. Authenticate to the TOE via the Console. 5. Perform an activity then wait for 3 minutes to elapse and verify that the session is terminated. 6. Authenticate to the TOE via the CLI using SSH. 7. Wait for 3 minutes to elapse and verify that the session is terminated. 8. Repeat Steps 1-7, except specify the inactivity period to 4 minutes. 9. Repeat Steps 1-7, except specify the inactivity period to 5 minutes.
Test Results	For each configured inactivity timeout value, the TOE successfully terminated the session. - Pass
Execution Method	Manual

Test Case Number	083
SFR	FTA_SSL.4
Test Objective	For each method of remote administration, the evaluator shall perform the following tests: a) Test 1: The evaluator initiates an interactive local session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the local console. 2. Type the "exit" command to terminate the console session. 3. Observe that the console session has been terminated.
Test Results	The local administrator was successful in manually terminating the local CLI connection. - Pass
Execution Method	Manual

Test Case Number	084
SFR	FTA_SSL.4
Test Objective	For each method of remote administration, the evaluator shall perform the following tests: b) Test 2: The evaluator initiates an interactive remote session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Instructions	Execute this test per the test steps.
Test Steps	**CLI (SSH)** 1. Authenticate to the TOE via SSH. 2. Type the "exit" command to terminate the SSH session. 3. Observe that the SSH session has been terminated. **Console Graphical User Interface (GUI)** 1. Authenticate to the TOE via the Console. 2. Log out of the Console by clicking "File" > "Log-Out". 3. Confirm the exit by clicking "Yes" on the dialog box.

	4. Observe that the Console session has been terminated.
Test Results	The remote administrator was successful in manually terminating the remote SSH CLI and GUI (Console) connection. - Pass
Execution Method	Manual

Test Case Number	085
SFR	FTA_TAB.1
Test Objective	The evaluator shall also perform the following test: a) Test 1: The evaluator follows the guidance documentation to configure a notice and consent warning message. The evaluator shall then, for each method of access specified in the TSS, establish a session with the TOE. The evaluator shall verify that the notice and consent warning message is displayed in each instance.
Test Instructions	Execute this test per the test steps.
Test Steps	1. Authenticate to the TOE via the Console using the "admin" account. 2. Set the login banner: a. Navigate to "Tools" > "Options" > "CounterACT User Profiles" > "Password and Sessions". b. Select the "Login" tab. c. Check the box, "Before login, prompt user to accept these Terms and Conditions," and type a message in the adjacent text box. d. Select "Apply" 3. Log out of the TOE. 4. Authenticate to the TOE via the Console. 5. Verify that the configured warning banner is displayed prior to authentication. 6. Authenticate to the TOE via SSH. 7. Verify that the configured warning banner is displayed prior to authentication. 8. Log out of the TOE. 9. Authenticate to the TOE via the local console. 10. Verify that the configured warning banner is displayed prior to authentication.
Test Results	The configured warning banner is displayed on all of the claimed interfaces used for authentication to the TOE (local console, SSH CLI, remote GUI (Console)). - Pass
Execution Method	Manual

4.3.7 Trusted Path/Channels

Test Case Number	086
SFR	FTP_ITC.1
Test Objective	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests:

	a) Test 1: The evaluators shall ensure that communications using each protocol with each authorized IT entity is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful. Further assurance activities are associated with the specific protocols.
Test Instructions	Execute this test per the test steps.
Test Steps	**Remote Audit Server** 1. Begin capturing packets between the TOE and the remote audit log server. 2. Initiate communication from the TOE to the remote audit log server. 3. Perform some activity on the TOE that causes audit log data to be transmitted to the remote audit server. 4. Stop capturing packets between the TOE and the remote audit log server. 5. Verify that the communications between the TOE and the remote audit log server are not sent in plaintext. **Active Directory** 1. Begin capturing packets between the TOE and the Active Directory server. 2. Initiate communication from the TOE to the Active Directory server. 3. Perform some activity on the TOE that causes the TOE to communicate with the Active Directory server. 4. Stop capturing packets between the TOE and the Active Directory server. 5. Verify that the communications between the TOE and the Active Directory server are not sent in plaintext.
Test Results	The TOE successfully negotiated a secure channel to the audit server and AD using TLS. Communications were not sent in plaintext to either server. - Pass
Execution Method	Manual

Test Case Number	087
SFR	FTP_ITC.1
Test Objective	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: b) Test 2: For each protocol that the TOE can initiate as defined in the requirement, the evaluator shall follow the guidance documentation to ensure that in fact the communication channel can be initiated from the TOE. Further assurance activities are associated with the specific protocols.
Test Instructions	Execute this test per the test steps.
Test Steps	Testing for this SFR is met by the testing performed in FTP_ITC.1 – Test 1 (Test Case 086).
Test Results	Pass
Execution Method	Manual

Test Case Number	088
SFR	FTP_ITC.1
Test Objective	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: c) Test 3: The evaluator shall ensure, for each communication channel with an authorized IT entity, the channel data is not sent in plaintext. Further assurance activities are associated with the specific protocols.
Test Instructions	Execute this test per the test steps.
Test Steps	Testing for this SFR is met by the testing performed in FTP_ITC.1 – Test 1 (Test Case 086).
Test Results	Pass
Execution Method	Manual

Test Case Number	089
SFR	FTP_ITC.1
Test Objective	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: d) Test 4: Objective: The objective of this test is to ensure that the TOE reacts appropriately to any connection outage or interruption of the route to the external IT entities. The evaluator shall, for each instance where the TOE acts as a client utilizing a secure communication mechanism with a distinct IT entity, physically interrupt the connection of that IT entity for the following durations: i) a duration that exceeds the TOE's application layer timeout setting, ii) a duration shorter than the application layer timeout but of sufficient length to interrupt the network link layer. The evaluator shall ensure that, when the physical connectivity is restored, communications are appropriately protected and no TSF data is sent in plaintext. In the case where the TOE is able to detect when the cable is removed from the device, another physical network device (e.g. a core switch) shall be used to interrupt the connection between the TOE and the distinct IT entity. The interruption shall not be performed at the virtual node (e.g. virtual switch) and must be physical in nature. Further assurance activities are associated with the specific protocols.

Test Instructions	Execute this test per the test steps.
Test Steps	**Remote Audit Server** 1. Begin capturing packets between the TOE and the remote audit log server. 2. Initiate communication from the TOE to the remote audit log server. 3. Perform some activity on the TOE that causes audit log data to be transmitted to the remote audit server. 4. Physically disconnect the connection between the TOE and the remote audit log server. 5. Restore the connection between the TOE and the remote audit log server immediately after a period that exceeds the TOE's application layer timeout setting. 6. Stop capturing packets between the TOE and the remote audit log server. 7. Verify that the communications between the TOE and the remote audit log server are not sent in plaintext. 8. Repeat Steps 1-7, except in Step 5, perform the reconnect such that the disconnect duration is shorter than the application layer timeout but of sufficient length to interrupt the network link layer. **Active Directory** 1. Begin capturing packets between the TOE and the Active Directory server. 2. Initiate communication from the TOE to the Active Directory server. 3. Perform some activity on the TOE that causes the TOE to communicate with the Active Directory server. 4. Physically disconnect the connection between the TOE and the Active Directory server. 5. Restore the connection between the TOE and the Active Directory server immediately after a period that exceeds the TOE's application layer timeout setting. 6. Stop capturing packets between the TOE and the Active Directory server. 7. Verify that the communications between the TOE and the Active Directory server are not sent in plaintext. 8. Repeat Steps 1-7, except in Step 5, perform the reconnect such that the disconnect duration is shorter than the application layer timeout but of sufficient length to interrupt the network link layer.
Test Results	When physical connectivity is restored, communications are appropriately protected and no TSF data is sent in plaintext. NOTE: Due to the ephemeral nature of the Active Directory (AD) connection, a new handshake is performed in both of the tested time durations. Communications are appropriately protected and no TSF data is sent in plaintext in these instances as well. - Pass
Execution Method	Manual

Test Case Number	090
SFR	FTP_TRP.1/Admin
Test Objective	The evaluator shall perform the following tests:

	a) Test 1: The evaluators shall ensure that communications using each specified (in the guidance documentation) remote administration method is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful. Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.
Test Instructions	Execute this test per the test steps.
Test Steps	**CLI (SSH)** 1. Begin capturing packets between the TOE and the test machine. 2. Authenticate to the TOE via SSH. 3. Type the "exit" command to terminate the SSH session. 4. Observe that the SSH session has been terminated. 5. Stop capturing packets between the TOE and the test machine. 6. Examine the packet capture and verify that the data transmitted between the test machine and the TOE is protected using SSH and not sent in plaintext. **Console (TLS)** 1. Begin capturing packets between the TOE and the test machine. 2. Authenticate to the TOE via the Console. 3. Log out of the Console by clicking "File" > "Log-Out". 4. Confirm the exit by clicking "Yes" on the dialog box. 5. Stop capturing packets between the TOE and the test machine. 6. Examine the packet capture and verify that the data transmitted between the test machine and the TOE is protected using TLS and not sent in plaintext. Audit records for failed remote CLI (SSH) connection establishment are produced as part of FCS_SSHS_EXT.1 test assurance activities. Audit records for failed remote Console (TLS) connection establishment are produced as part of FCS_TLSS_EXT.1 test assurance activities.
Test Results	The TOE successfully negotiated a TLS secure channel when connected to the GUI (Console) and a SSH channel for remote SSH CLI access. Communications were not sent in plaintext to either server. - Pass
Execution Method	Manual

Test Case Number	091
SFR	FTP_TRP.1/Admin
Test Objective	The evaluator shall perform the following tests: b) Test 2: The evaluator shall ensure, for each communication channel, the channel data is not sent in plaintext. Further assurance activities are associated with the specific protocols.

	For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.
Test Instructions	Execute this test per the test steps.
Test Steps	Testing for this SFR is met by the testing performed in FTP_TRP.1/Admin – Test 1 (Test Case 090).
Test Results	Pass
Execution Method	Manual

5 Evaluation Activities for SARs

This section addresses assurance activities that are defined in the *collaborative Protection Profile for Network Devices Version 2.2e* [NDcPP] that correspond with Security Assurance Requirements.

ADV_FSP.1-1 & **ADV_FSP.1-2** – *"The evaluator shall examine the interface documentation to ensure it describes the purpose and method of use for each TSFI that is identified as being security relevant."*

Section 1.4 of the Security Target provides a labeled figure. Table 5 in the Security Target describes the purpose and method of use for each security relevant TSFI by enumerating all security relevant interfaces, including those used for remote administration of the TOE and where the TOE communicates with an external IT entity in the operational environment. It was found to describe the following security relevant interfaces:

- E1: Local user to TOE – This is the local console interface in which the user must connect to the TOE's CLI directly over the USB and VGA ports in order to perform management functions.
- E2: Remote user to TOE – This is the remote management interface in which the user must connect to the TOE's CLI using SSH.
- E3: Console to TOE – This interface is used for remote TOE management. All communications are protected using TLS.
- E4: TOE to Server Switch – Interface used by the TOE to obtain DHCP, DNS, SMTP and SNMP information. It is strictly considered an external interface that would be used for the TOEs intended operational capabilities that are outside the scope of the PP. As such, it is non-SFR interfering.
- E5: TOE to Certificate Authority Server (CA) – This interface is used for verifying the revocation status of certificates that the TOE uses.
- E6: TOE to User Directory AD Server – This interface is used for remote authentication of a user of the TOE via an Active Directory server. All communications are protected using TLS.
- E7: TOE to Audit Server – This interface is used by the TOE to audit data to the audit server. All communications are protected using TLS.
- E8: Console to Update Server – This interface is used to manually obtain Forescout software updates from a non-TOE device. The TOE itself does not directly connect to the update server in the evaluated configuration. It is strictly considered an external interface but is provided as part of showing the complete test environment used. As such, it is non-SFR interfering.

ADV_FSP.1-3 – *"The evaluator shall check the interface documentation to ensure it identifies and describes the parameters for each TSFI that is identified as being security relevant."*

The AGD was developed with the intent to provide the specific guidance for managing TOE functionality or a pointer to the necessary documentation as defined by the Intended Audience statement in Section 2. Thus, the evaluation team has determined that only the commands located within the AGD and the specific pointers to other documents are considered to be security relevant for this evaluation. Through the completion of the independent functional testing, the evaluation team was able to test each SFR by executing the commands in each SFR's relevant test case(s). The evaluation team has determined that since the AGD document contains and/or provides the necessary pointer for all security relevant commands that were executed by the evaluation team in performing the independent testing, that the subset of the commands defined or referenced to in the AGD are all of the security relevant commands necessary to enforce the SFRs specified in the NDcPP.

ADV_FSP.1-5 – *"The evaluator shall examine the interface documentation to develop a mapping of the interfaces to SFRs."*

The descriptions provided for each SFR in the TSS Section of the Security Target adequately describe a mapping of any relevant interfaces to that SFR. For example, Sections 8.1.2 (Audit Server) and 8.2.10 (TLS Client), and 8.7.1 (Trusted Channel) in the Security Target describe SFRs that map to interface E7

where the TOE communicates with an external IT entity. Specifically, the interface between the TOE and the Audit Server is a trusted channel interface that maps to FTP_ITC.1. The interface between a remote administrator and the TOE's CLI interface is the trusted path that maps to FTP_TRP.1/Admin, interface E2. Additionally, Section 8.6.1 describes the local console interface inactivity termination behavior for local TOE administration which maps to interface E1. There are two interfaces (E4 and E8) that are non-SFR interfering/supporting as they either are strictly for operational capabilities of the product that are outside the scope of this PP.

AGD_OPE.1 – *"The evaluator shall ensure the Operational guidance documentation is distributed to Security Administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that Security Administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration."*

"The evaluator shall ensure that the Operational guidance is provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target."

"The evaluator shall ensure that the Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE."

"The evaluator shall ensure the Operational guidance makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs."

"In addition, the evaluator shall ensure that the following requirements are also met.

> *a) The guidance documentation shall contain instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.*
>
> *b) The documentation must describe the process for verifying updates to the TOE by verifying a digital signature. The evaluator shall verify that this process includes the following steps:*
>
> > *5) Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).*
> >
> > *6) Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes instructions that describe at least one method of validating the hash/digital signature.*
>
> *c) The TOE will likely contain security functionality that does not fall in the scope of evaluation under this cPP. The guidance documentation shall make it clear to an administrator which security functionality is covered by the Evaluation Activities."*

Section 6.1 of the AGD indicates that when the TOE is in FIPS mode that it uses the cryptography described in the ST for all claimed cryptographic operations. Section 2 of the AGD also has the following warning: "Any functionality that is not described here or in the Forescout Security Target was not evaluated and should be exercised at the user's risk."

Section 6.3 and 7.8 of the AGD describes obtaining software images of TOE, including updates, from Forescout's website. Section 6.3 and 7.8 also describes the update process and procedures. Sections 6.3 and 7.8 of the AGD consistently describe the verification process of a software image. A digital signature mechanism is used to verify the update files prior to the installation of the software. The AGD also describes how the user can determine if the installation was successful or failed.

Section 2 of the AGD states "This document is intended for administrators responsible for installing, configuring, and/or operating Forescout. Guidance provided in this document allows the reader to deploy the product in an environment that is consistent with the configuration that was evaluated as part of the product's Common Criteria (CC) testing process. It also provides the reader with instructions on how to exercise the security functions that were claimed as part of the CC evaluation. The reader is also expected to be familiar with the general operation of the Forescout product. This supplemental guidance includes references to Forescout's standard documentation set for the product and does not explicitly reproduce materials located there. The reader is also expected to be familiar with the Forescout Security Target and the general CC terminology that is referenced in it. This document references the Security Functional Requirements (SFRs) that are defined in the Security Target document and provides instructions for how to perform the security functions that are defined by these SFRs. The Forescout product as a whole provides a great deal of security functionality but only those functions that were in the scope of the claimed PP are discussed here. Any functionality that is not described here or in the Forescout Security Target was not evaluated and should be exercised at the user's risk." Since the AGD references external documents, it is understood that only the sections of those external documents specifically referenced by the AGD are an extension to the statement above. All other portions of the externally referenced documents are considered outside the scope of the evaluation.

AGD_PRE.1 – *"The evaluator shall examine the Preparative procedures to ensure they include a description of how the Security Administrator verifies that the operational environment can fulfil its role to support the security functionality (including the requirements of the Security Objectives for the Operational Environment specified in the Security Target)."*

"The evaluator shall examine the Preparative procedures to ensure they are provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target."

"The evaluator shall examine the preparative procedures to ensure they include instructions to successfully install the TSF in each Operational Environment."

"The evaluator shall examine the preparative procedures to ensure they include instructions to manage the security of the TSF as a product and as a component of the larger operational environment."

"In addition, the evaluator shall ensure that the following requirements are also met.

The preparative procedures must

a) include instructions to provide a protected administrative capability; and

b) identify TOE passwords that have default values associated with them and instructions shall be provided for how these can be changed."

Section 5.3 of the AGD has all assumptions for the Operational Environment that are defined within Section 4.3 of the ST. Each assumption a description of the actions that the TOE administrators must perform to ensure that all of the security assumptions for the Operational Environment are met. These assumptions map to the security objectives for the Operational Environment that are defined in the NDcPP. The actions provided are written in an easily understood style and clear regarding the requirements levied on the Operational Environment.

Section 5.1 of the AGD describes all models of the TOE claimed in Section 2.4 of the ST as well as states that testing was only conducted against these models, running the Forescout software version 8.3. The AGD and reference pointers to other Forescout documentation do not describe any methods of configuring and/or administration between the models. For these reasons, the evaluation team finds that the TOE platforms are adequately addressed by the AGD documentation.

Section 5.2 of the AGD describes the non-TOE products within the Operational Environment which describes their use and requirements. The non-TOE products include the Management Workstation, Update Server, Certificate Authority (CA) Server/Online Certificate Status Protocol (OCSP) Responder, Active Directory Server, Syslog Server, and Network Infrastructure. The Management Workstation is used by administrators to connect to the TOE remotely via an SSH client or the Forescout Console client. The update server is a general-purpose computer that is used to download updates from Forescout's web site and then also functions in conjunction with the Management Workstation to establish a connection to the TOE via the Forescout Console client. The Certificate Authority (CA) Server/Online Certificate Status Protocol (OCSP) Responder provides certificates that can be installed on the TOE as well as operational environment components the TOE establishes connections with over TLS. When the TOE establishes TLS connections to the remote syslog server or the remote Active Directory server, the revocation status of the certificates presented by those servers is verified by the OCSP Responder. The Active Directory instance is a remote authentication store and is described in Section 6.8 of the AGD. The TOE connects to the remote Syslog Server to send Syslog messages for remote storage over TLS. Section 6.7 describes the configuration of the Syslog client on the TOE for secure communication with a remote Syslog server. Network Infrastructure includes components such as routers, switches, and DNS services the TOE may utilize in order to communicate with entities in the Operation Environment. For these reasons, the evaluation team finds that the preparative procedures for each product the TOE supports in the Operational Environment is adequately described.

Section 6 of the AGD provides the initial installation of the TOE. As reviewed in the prior paragraph, several sections of the AGD describe the instructions to install and configure the TOE and the Operational Environment products to communicate. Additionally, through the review of the Operational Guidance Assurance Activities for each SFR, the evaluation team determined that the AGD provides instructions on managing the TOE itself as related to the SFRs, managing the TOE and preparing the Operational Environment products to communicate, and include descriptions to protect the administration of the TOE through TOE functions. Finally, Section 5.3 of the AGD has all assumptions for the Operational Environment that are defined within Section 4.3 of the ST. Each assumption includes a description of the actions that the TOE administrators must perform to ensure that all of the security assumptions for the Operational Environment are met. These assumptions map to the security objectives for the Operational Environment that are defined in the NDcPP. The actions provided are written in an easily understood style and clear regarding the requirements levied on the Operational Environment. For these reasons, the evaluation team has determined that the AGD contains the necessary instructions and information to securely operate the TOE within the Operational Environment that has been described in the ST.

ALC_CMC.1 – *"When evaluating that the TOE has been provided and is labelled with a unique reference, the evaluator performs the work units as presented in the CEM."*

The evaluation team verified that the Security Target (ST), TOE, and Supplemental Administrative Guidance (AGD) were labeled consistently to correctly identify the hardware and software versions in the CC evaluation. The ST clearly specifies the TOE Reference as being the "The TOE is the Forescout family of products, which includes the following appliance models:
CT-R, CT-100, CT-1000, CT-2000, CT-4000, CT-10000, CEM-5, CEM-10, CEM-25, CEM-50, CEM-100, CEM-150, CEM-200, 4130, 5110, 5120, 5140, and 5160.

Each appliance runs software version 8.3." In addition, the ST specifies the TOE models that are specifically covered in Tables 6-9. The evaluation team verified that the version number of the software matched the version in the TOE Reference.

The TOE hardware was identified by physical examination of the network appliance.

ALC_CMS.1 – *"When evaluating the developer's coverage of the TOE in their CM system, the evaluator performs the work units as presented in the CEM."*

The evaluation team verified that the Security Target (ST), TOE, and Supplemental Administrative Guidance (AGD) were labeled consistently to correctly identify the hardware and software versions in the

CC evaluation. The ST clearly specifies the TOE Reference as being the "The TOE is the Forescout family of products, which includes the following appliance models:
CT-R, CT-100, CT-1000, CT-2000, CT-4000, CT-10000, CEM-5, CEM-10, CEM-25, CEM-50, CEM-100, CEM-150, and CEM-200, 5110, 5120, 5140, 5160
Each appliance runs software version 8.3." In addition, the ST specifies the TOE models that are specifically covered in Tables 6-9. The evaluation team verified that the version number of the software matched the version in the TOE Reference.

The TOE hardware was identified by physical examination of the network appliance.

AVA_VAN.1 – TD0547 – *"The evaluator shall examine the documentation outlined below provided by the developer to confirm that it contains all required information. This documentation is in addition to the documentation already required to be supplied in response to the EAs listed previously."*

"The evaluator formulates hypotheses in accordance with process defined in Appendix A. The evaluator documents the flaw hypotheses generated for the TOE in the report in accordance with the guidelines in Appendix A.3. The evaluator shall perform vulnerability analysis in accordance with Appendix A.2. The results of the analysis shall be documented in the report according to Appendix A.3."

The evaluation team created a set of vulnerability tests to attempt to subvert the security of the TOE. These tests were created based upon the evaluation team's review of the vulnerability analysis evidence and independent research. The evaluation team conducted searches for public vulnerabilities related to the TOE. A few notable resources consulted include securityfocus.com, the cve.mitre.org, and the nvd.nist.gov.

The following keywords were used individually and as part of various permutations and combinations to search for vulnerabilities identified in the public domain:

Keyword	Description
Forescout	This is a generic term for searching for known vulnerabilities produced by the company as a whole.
CounterACT	This is a generic term for searching for known vulnerabilities for the specific product. NOTE: The TOE is no longer referred to as CounterACT, however, we still included this in our search because the product name change was recent.
CentOS 7.5	This is a term for searching for known vulnerabilities for the underlying OS. A specific version was not included in the search because this version may be within a range of vulnerable operating system versions and not listed separately. Bearing in mind that this is a locked operating system that has been enhanced by the vendor who is not using the full functionality of the OS.
Model/nomenclature CEM CT-R, CT-100, CT-1000, CT-2000, CT-4000, CT-10000, CEM-5, CEM-10, CEM-25, CEM-50, CEM-100, CEM-150, CEM-200, 4130, 5110, 5120, 5140, and 5160.	Specific models search / nomenclature search
Generic Terminology	
Central Enterprise Manager	Generic term
Libraries	

Keyword	Description
OpenSSL (1.0.2k build 23)	This is a term for searching for known vulnerabilities for the underlying cryptographic software utilized by the TOE. A specific version was not included in the search because this version may be within a range of vulnerable operating system versions and not listed separately. However, the version was used to further filter findings.
OpenSSH 7.4p1-22	This is a term for searching for known vulnerabilities for the OpenSSH server utilized by the TOE. A specific version was not included in the search because this version may be within a range of vulnerable operating system versions and not listed separately. However, the version was used to further filter findings.
BC-FJA 1.0.2 (Bouncy Castle)	A specific version was not included in the search because this version may be within a range of vulnerable Bouncy Castle versions and not listed separately. However, the version was used to further filter findings.
OpenJDK 1.8.0_282 (8u282 alternative)	A specific version was not included in the search because this version may be within a range of vulnerable Java versions and not listed separately. However, the version was used to further filter findings.
PostgreSQL (Postgres) 13.1	A specific version was not included in the search because this version may be within a range of vulnerable Java versions and not listed separately. However, the version was used to further filter findings.
NMAP 7.91 and 5.21	A specific version was not included in the search because this version may be within a range of vulnerable Java versions and not listed separately. However, the versions were used to further filter findings. Both versions were part of the filtered search.
Hardware	
Intel Celeron J1900 (Bay Trail)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Intel Xeon E5 2609 v3 (Haswell)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Intel Xeon E5 2620 v3 (Haswell)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Intel Xeon E5 2640 v3 (Haswell)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Intel Xeon E5 2650 v3 (Haswell)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Xeon Silver 4110 (Skylake)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Xeon Silver 4114 (Skylake)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Xeon Gold 5118 (Skylake)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Xeon Gold 6132 (Skylake)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites
Gen 8 Intel® Core™ i5-8500T (Coffee Lake)	Terms used in the advanced searches for NVD and CVE details websites. Generic terms used in various combinations for additional websites

The TOE handles a large number of network protocols. However, any vulnerability that may be present in the TOE's implementation of these protocols will also show up in a product-specific search (e.g. "A vulnerability in the CT-1000 Series implementation of SSH may allow a remote attacker to…"). Therefore, searches for the specific protocols were not performed.

Upon the completion of the vulnerability analysis research, the team had identified generic vulnerabilities upon which to build a test suite. These tests were created specifically with the intent of exploiting these vulnerabilities within the TOE or its configuration. The team tested the following areas:

- Port Scanning
  Remote access to the TOE should be limited to the standard TOE interfaces and procedures. This test enumerates network port and service information to determine if any ports were open and running services outside of the TOE standard configuration.
- SSH Timing Attack (User Enumeration)
  This attack attempts to enumerate validate usernames for the SSH interface, by exploiting a vulnerability in OpenSSH as described in CVE-2018-15473.
- Force SSHv1
  This attack determines if the client will accept both SSHv1 and SSHv2 connections when the TOE claims to only support SSHv2.

The results of the tests were as follows:

- Port Scanning – The open ports and running services were consistent with the evaluated configuration of the TSF and its environmental dependencies (i.e. every service found to be running was necessary in some way)
- SSH Timing Attack (User Enumeration) – The TSF's SSH server does not behave any differently when receiving valid versus invalid username data.
- Force SSHv1 – the TOE successfully blocked attempts at forcing SSHv1

The TOE successfully prevented any attempts of subverting its security.

Verdict: Through the completion of the vulnerability testing and public search (updated 7/19/2022), the evaluation team determined that the TOE was not vulnerable to any of the defined attacks or had unsatisfied publicly known vulnerabilities. There were no issues discovered that could affect the security posture of a deployed system.
The evaluation team has completed vulnerability testing of this component, resulting in a verdict of PASS.

6 Conclusions

The TOE was evaluated against the ST and has been found by this evaluation team to be conformant with the ST. The overall verdict for this evaluation is: Pass.

7 Glossary of Terms

Acronym	Definition
AD	Active Directory
CC	Common Criteria
CLI	Command-line Interface
CPU	Central Processing Unit
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
NIAP	National Information Assurance Partnership
NTP	Network Time Protocol
OS	Operating System
PP	Protection Profile
SAR	Security Assurance Requirement
SCP	Secure Copy Protocol
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSL	Secure Sockets Layer
SSH	Secure Shell
ST	Security Target
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function

Table 6-1: Acronyms

Term	Definition
Administrator, System Administrator, Security Administrator	The class of TOE administrators that are tasked with managing the TOE's functional and security configuration. Embodies those administrators that have access to the CLI and Console.
Connection	One to One simple flows between a network port and a tool port.
Console or Console application	The Forescout Console is a GUI application used for creating NAC, firewall and IPS policies, generating reports, viewing and managing detection information, and managing Forescout Appliances.
Endpoint	A Network Host discovered by the Forescout platform, for example desktop, laptop, server, etc.
Enterprise Manager	A Forescout platform configured to manage multiple Appliances distributed across the network.
Local CLI	When the TOE's command line interface (CLI) is accessed locally with a physical connection to the TOE via the keyboard/video ports or a serial port and a terminal emulator that is compatible with serial communications is referred to as the local

	console.
Plugins	Functionality enhancement modules that can be incorporated into the Forescout platform. Plugins enable deeper inspection as well as broader control over network endpoints. Bundled plugins are pre-packaged with the Forescout platform. Other plugins may be available from Forescout or from a third party.
Network Port	Where data arrives into the TOE. The ports which receive copied network data for the TOE.
Remote console	When the TOE's CLI is accessed remotely using SSH is referred to as the remote console.
Administrator, System Administrator, Security Administrator	The class of TOE administrators that are tasked with managing the TOE's functional and security configuration. Embodies those administrators that have access to the CLI and Console.
Connection	One to One simple flows between a network port and a tool port.
Console or Console application	The Forescout Console is a GUI application used for creating NAC, firewall and IPS policies, generating reports, viewing and managing detection information, and managing Forescout Appliances.
Endpoint	A Network Host discovered by the Forescout platform, for example desktop, laptop, server, etc.

Table 6-2: Terminology