Commercial Solutions for Classified (CSfC) Selections for Transport Layer Security (TLS) Protected Servers

Overview

Transport Layer Security (TLS) Protected Server products (as defined in the Mobile Access (MA) Capability Package (CP)) used in Commercial Solutions for Classified (CSfC) solutions shall be validated by National Information Assurance Partnership (NIAP)/Common Criteria Evaluation and Validation Scheme (CCEVS) or Common Criteria Recognition Arrangement (CCRA) partnering schemes as complying with the current requirements of NIAP's Protection Profile for Network Devices Version 2.2e and this validated compliance shall include the selectable requirements contained in this document.

TLS can be used/implemented in many different ways, threats and technology continuously progress, and TLS continues to evolve, which may cause the below selections to change or become obsolete. The objective of the below selections is to provide information to support the use of the Commercial National Security Algorithm Suite (CNSA Suite) and support the use of TLS Protected Servers in CSfC Solutions.

Please provide questions, comments on usability, applicability, and/or shortcomings to the CSfC Program (csfc@nsa.gov).

Notes

Note 1: The following selections apply to CSfC TLS Protected Server functionality. If needed, functionality and/or configurations outside the scope of a CSfC TLS protected server that conflict with the CSfC selections could be NIAP validated without using a separate iteration of the Security Functional Requirement (SFR) (this is a change to previous guidance in Note 1). The Security Target (ST) author should document a specific CSfC TLS Protected Server configuration in the product's Administrative Guide with a note that the configuration should be considered the NIAP-certified evaluated configuration for CSfC TLS Protected Server Use Cases. The CSfC TLS Protected Server configuration should be used to validate compliance with CSfC selections.

Note 2: See TD0591 for clarification on Physical Network Devices (pND) and Virtual Network Devices (vND).

Note 3: The below SFRs/Selections contain some mandatory SFRs without Selections or modifications. The exclusion of other mandatory SFRs in the below Selections does not indicate that mandatory NDcPP SFRs aren't required (i.e., Compliance with the NDcPP requirements as prescribed by the PP and outlined in the Overview Section above is required). Some mandatory SFRs are included in the below Selections to highlight some SFRs relevant to CSfC TLS protected servers.

Document Conventions

The conventions used in descriptions of the document are as follows:

- Assignment completed within a selection in the PP: the completed assignment text is indicated with *italicized and underlined text* (i.e., CSfC mandatory completed assignments/selections unless otherwise indicated by the text "at least one of the following underlined selections")
- Assignment partially completed in the PP: indicated with *italicized text*
- Refinement text is indicated with ~~strikethroughs~~
- Additional clarifying text or CSfC specific language is indicated with `light blue Courier New Text`
- Links to sources, additional information, and email addresses are indicated with blue underlined text.

Protection Profile for Network Devices Version 2.2e Selections

FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with `at least one of the following` specified cryptographic key generation algorithm: [**Selection:**

- *RSA schemes using cryptographic key sizes of 2048-bit `and 3072-bits` or greater that meet the following FIPS PUB 186-4, "Digital Signature Standard (DSS), Appendix B.3";*
- *ECC schemes using NIST curves [Selection: ~~P-256~~, P-384, ~~P-521~~] that meet the following: FIPS PUB 186-4, "Digital Signature Standard(DSS)", Appendix B.4*];
- ~~*FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1*~~
- *FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and* [***Selection***: *RFC 3526, RFC 7919*].

] ~~and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*]~~.

FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with `at least one of the following` specified cryptographic key establishment methods: [**Selection:**

- ~~*RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";*~~
- ~~*Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"*~~
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"* (`See TD0581`);
- ~~*Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";*~~
- *FFC Schemes using "safe-prime" groups that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and* [***Selection***: *groups listed in RFC 3526, groups listed in RFC 7919*] (`See TD0580`).

] ~~that meets the following: [assignment: *list of standards*]~~.

FCS_COP.1.1/DataEncryption
The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [**Selection:** *CBC, CTR, GCM*] mode and cryptographic key sizes [Selection: ~~*128 bits*~~, ~~*192 bits*~~, *256 bits*] that meet the following: AES as specified in ISO 18033-3, [**Selection:** *CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772*].

FCS_COP.1.1/SigGen
The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with `at least one of the following` a specified cryptographic algorithm: [**Selection**:

- *RSA Digital Signature Algorithm using cryptographic key sizes (modulus) of [assignment*: *3072 bits or greater];*
- *Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [assignment*: *384 bits or greater]]*

that meet `at least one of the following that corresponds to the previous selection`*:* [**Selection:**

- *For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3;*
- *For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [*selection: ~~*P-256*~~, *P-384*, ~~*P-521*~~*]; ISO/IEC 14888-3, Section 6.4*]

].

FCS_COP.1.1/Hash
The TSF shall perform *cryptographic hashing* services in accordance with ~~a~~ specified cryptographic algorithm [**Selection**: ~~*SHA-1*~~, *SHA-256, SHA-384*, *SHA-512*] ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ and message digest sizes [**Selection**: ~~*160*~~, *256*, *384*, *512*] bits that meet the following: ISO/IEC 10118-3:2004.

FCS_COP.1.1/KeyedHash
The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [**Selection**: ~~*HMAC-SHA-1, HMAC-SHA-256*~~, *HMAC-SHA-384*, *HMAC-SHA-512*] and cryptographic key sizes [**Assignment**: *key size(s) in bits ≥ the message digest size(s)*] and message digest sizes [**Selection**: ~~*160, 256*~~, *384*, *512*] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using `at least one of the following` [**Selection:** *Hash_DRBG (`SHA-384`, `SHA-512`), HMAC_DRBG (`SHA-384`, `SHA-512`), CTR_DRBG (`AES-256`)*].

`Application Note:` `The objective of the CSfC specific language for DRBG algorithms is to ensure compatibility with the CSfC CPs by selecting compliant algorithms that provide the required security strength.`

FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [**Selection**: [*Assignment*: *number of software-based sources*] software-based noise source, [*assignment*: *at least one (1)] platform-based noise source*]] with a minimum of [Selection: ~~*128 bit*~~, ~~*192 bits*~~, *256 bits*]

of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using `at least one of the following` [**Selection**: *X.509 certificate*, *digital signature*, *published hash*] prior to installing those updates.

FPT_STM_EXT.1.1
The TSF shall be able to provide reliable time stamps for its own use.

FPT_STM_EXT.1.2
The TSF shall [**Selection:** *allow the Security Administrator to set the time*, *synchronize time with an NTP server*].

FTP_ITC.1.1
The TSF shall be capable of using [**Selection:** *IPsec*, *SSH*, *TLS*, ~~*DTLS*~~, *HTTPS*] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [**Selection:** *authentication server*, [***Assignment*:** *TLS Software Applications: HTTPS/TLS Clients on authorized End User Devices (EUDs)*, *other capabilities*], *no other capabilities*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

Application Note: `TLS Servers in CSfC Solutions must support HTTPS and/or TLS to/from the TLS Software Application. SSH, TLS, and/or IPsec are all acceptable selections for audit server connections in CSfC Solutions.`

FTP_TRP.1.1/Admin
The TSF shall be capable of using `at least one of the following` [**Selection:** ~~*DTLS*~~, *IPsec*, *SSH*, *TLS*, *HTTPS*] to provide a communication path between itself and authorized remote Administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and provides detection of modification of the channel data.

FTP_TRP.1.2/Admin
The TSF shall permit remote Administrators to initiate communication via the trusted path.

FTP_TRP.1.3/Admin
The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

FIA_X509_EXT.1.1/ITT
If applicable due to a distributed TOE, the TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of two certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using at least one of the following [**Selection:** *the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3* and *Certificate Revocation List (CRL) as specified in ~~RFC 5759~~* RFC 8603 *Section ~~5~~*7]
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - 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.

FAU_STG.1.1
The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2
The TSF shall be able to prevent unauthorized modifications to the stored audit records in the audit trail.

FPT_ITT.1.1
If applicable due to a distributed TOE, the TSF shall protect TSF data from disclosure and detect its modification when it is transmitted between separate parts of the TOE through the use of [**Selection:** *IPsec, SSH, TLS, ~~DTLS~~, HTTPS*].

FCS_SSHC_EXT.1.1
If the TOE has an SSH client, the TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [**Selection**: *4256, ~~4344~~, 5647, 5656, 6187, ~~6668~~, 8268, 8308 section 3.1, 8332*].

FCS_SSHC_EXT.1.2
If the TOE has an SSH client, the TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [**Selection**: *password-based, no other method*].

FCS_SSHC_EXT.1.3
If the TOE has an SSH client, the TSF shall ensure that, as described in RFC 4253, packets greater than [**Assignment**: *number of bytes*] bytes in an SSH transport connection are dropped.

FCS_SSHC_EXT.1.4
If the TOE has an SSH client, the TSF shall ensure that the SSH transport implementation uses at least one of the following encryption algorithms and rejects all other encryption algorithms: [**Selection***: ~~aes128-cbc~~, ~~aes256-cbc~~, ~~aes128-ctr~~, ~~aes256-ctr~~, ~~AEAD_AES_128_GCM~~, AEAD_AES_256_GCM, ~~aes128-gcm@openssh.com~~, aes256-gcm@openssh.com*].

FCS_SSHC_EXT.1.5

If the TOE has an SSH client, the TSF shall ensure that the SSH public-key based authentication implementation uses at least one of the following underlined selections [**Selection**: ~~*ssh-rsa*~~, ~~*rsa-sha2-256*~~, *rsa-sha2-512*, ~~*ecdsa-sha2-nistp256*~~, ~~*x509v3-ssh-rsa*~~, *ecdsa-sha2-nistp384*, ~~*ecdsa-sha2-nistp521*~~, ~~*x509v3-ecdsa-sha2-nistp256*~~, *x509v3-ecdsa-sha2-nistp384*, *x509v3-ecdsa-sha2-nistp521*, ~~*x509v3-rsa2048-sha256*~~] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHC_EXT.1.6

If the TOE has an SSH client, the TSF shall ensure that the SSH transport implementation uses at least one of the following [**Selection**: ~~*hmac-sha1*~~, ~~*hmac-sha1-96*~~, ~~*hmac-sha2-256*~~, ~~*hmac-sha2-512*~~, ~~*AEAD_AES_128_GCM*~~, *AEAD_AES_256_GCM*, *implicit*] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHC_EXT.1.7

If the TOE has an SSH client, the TSF shall ensure that [**Selection**: ~~*diffie-hellman-group14-sha1*~~, *diffie-hellman-group15-sha512*, ~~*ecdh-sha2-nistp256*~~] and [**Selection**: ~~*diffie-hellman-group14-sha256*~~, *diffie-hellman-group16-sha512*, ~~*diffie-hellman-group17-sha512*~~, ~~*diffie-hellman-group18-sha512*~~, *ecdh-sha2-nistp384*, ~~*ecdh-sha2-nistp521*~~, ~~*no other methods*~~] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_EXT.1.8

If the TOE has an SSH client, the TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

FCS_SSHC_EXT.1.9

If the TOE has an SSH client, the TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and [**Selection**: *a list of trusted certification authorities*, *no other methods*] as described in RFC 4251 section 4.1.

FCS_SSHS_EXT.1.1

If the TOE has an SSH Server, the TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [**Selection**: *4256*, ~~*4344*~~, *5647*, *5656*, *6187*, ~~*6668*~~, *8268*, *8308 section 3.1*, *8332*].

FCS_SSHS_EXT.1.2

If the TOE has an SSH Server, the TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [**Selection**: *password-based*, *no other method*].

FCS_SSHS_EXT.1.3

If the TOE has an SSH Server, the TSF shall ensure that, as described in RFC 4253, packets greater than [**Assignment**: *number of bytes*] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4

If the TOE has an SSH Server, the TSF shall ensure that the SSH transport implementation uses at least one of the following encryption algorithms and rejects all other encryption algorithms: [**Selection**: ~~*aes128-cbc*~~, ~~*aes256-cbc*~~, ~~*aes128-ctr*~~, ~~*aes256-ctr*~~, ~~*AEAD_AES_128_GCM*~~, *AEAD_AES_256_GCM*, ~~*aes128-gcm@openssh.com*~~, *aes256-gcm@openssh.com*].

FCS_SSHS_EXT.1.5

If the TOE has an SSH Server, the TSF shall ensure that the SSH public-key based authentication implementation uses at least one of the following underlined selections [**Selection**: ~~*ssh-rsa*~~, ~~*rsa-sha2-256*~~, *rsa-sha2-512*, ~~*ecdsa-sha2-nistp256*~~, ~~*x509v3-ssh-rsa*~~, *ecdsa-sha2-nistp384*, ~~*ecdsa-sha2-nistp521*~~, ~~*x509v3-ecdsa-sha2-nistp256*~~, *x509v3-ecdsa-sha2-nistp384*, ~~*x509v3-ecdsa-sha2-nistp521*~~, ~~*x509v3-rsa2048-sha256*~~] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6

If the TOE has an SSH Server, the TSF shall ensure that the SSH transport implementation uses at least one of the following [**Selection**: ~~*hmac-sha1*~~, ~~*hmac-sha1-96*~~, ~~*hmac-sha2-256*~~, ~~*hmac-sha2-512*~~, ~~*AEAD_AES_128_GCM*~~, *AEAD_AES_256_GCM*, *implicit*] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7

If the TOE has an SSH Server, the TSF shall ensure that [**Selection**: ~~*diffie-hellman-group14-sha1*~~, *diffie-hellman-group15-sha512*, ~~*ecdh-sha2-nistp256*~~] and [**Selection**: ~~*diffie-hellman-group14-sha256*~~, *diffie-hellman-group16-sha512*, ~~*diffie-hellman-group17-sha512*~~, ~~*diffie-hellman-group18-sha512*~~, *ecdh-sha2-nistp384*, ~~*ecdh-sha2-nistp521*~~, ~~*no-other-methods*~~] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8

If the TOE has an SSH Server, the TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

FCS_TLSS_EXT.2.1

The TSF shall support TLS communication with mutual authentication of TLS clients using X.509v3 certificates.

FCS_TLSS_EXT.2.2

When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also perform at least one of the following [**Selection**:

- *Not implement any administrator override mechanism*
- *require administrator authorization to establish the connection if the TSF fails to [selection: ~~match the reference identifier~~, ~~validate certificate path~~, ~~validate expiration date~~, determine the revocation status] of the presented client certificate*

].

FCS_TLSS_EXT.2.3
The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifiers according to RFC 6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.

FAU_STG_EXT.5.1
`If applicable due to a distributed TOE,` each TOE component which does not store security audit data locally shall be able to buffer security audit data locally until it has been transferred to another TOE component that stores or forwards it. All transfer of audit records between TOE components shall use a protected channel according to [**Selection:** *FPT_ITT.1*, *FTP_ITC.1*].

FCS_HTTPS_EXT.1.1
`If the TOE uses HTTPS,` the TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2
`If the TOE uses HTTPS,` the TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3
`If the TOE uses HTTPS` and if a peer certificate is presented, the TSF shall `perform at least one of the following` [**Selection**: ~~*not require client authentication*~~, *not establish the connection*, ~~*request authorization to establish the connection*~~, *[assignment: allow the Administrator to choose whether to establish the connection if the TSF fails to determine the revocation status]*] if the peer certificate is deemed invalid.

FCS_TLSS_EXT.1.1
The TSF shall implement [Selection: ~~*TLS 1.1 (RFC 4346)*~~, *TLS 1.2 (RFC 5246)*] and reject all other TLS and SSL versions. The TLS implementation will support `at least one of the following` ciphersuites: [**Selection:**

- ~~*TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*~~
- ~~*TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*~~
- ~~*TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*~~
- ~~*TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*~~
- ~~*TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*~~
- ~~*TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*~~
- ~~*TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*~~
- ~~*TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*~~
- ~~*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_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246*~~
- ~~*TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246*~~
- ~~*TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*~~
- ~~*TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*~~
- ~~*TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*~~
- *TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- ~~*TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*~~
- ~~*TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*~~
- ~~*TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*~~

- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- ~~*TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*~~
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- ~~*TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*~~
- ~~*TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*~~

].

FCS_TLSS_EXT.1.2
The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [Selection: *TLS 1.1*, ~~*TLS 1.2*~~, ~~*none*~~].

FCS_TLSS_EXT.1.3
The TSF shall perform key establishment for TLS using at least one of the following underlined selections [**Selection**: ~~*RSA with key size* [selection: *2048 bits, 3072 bits, 4096 bits*]~~, ~~*Diffie-Hellman parameters with size* [selection: *2048 bits, 3072 bits, 4096 bits, 6144 bits, 8192 bits*]~~, *Diffie-Hellman groups* [**Selection**: ~~*ffdhe2048*~~, *ffdhe3072*, *ffdhe4096*, ~~*ffdhe6144*~~, ~~*ffdhe8192*~~, *no other groups*], *ECDHE curves* [**Selection**: ~~*secp256r1*~~, *secp384r1*, ~~*secp521r1*~~*] and no other curves*]].

FCS_TLSS_EXT.1.4
The TSF shall support [Selection: *no session resumption or session tickets*, ~~*session resumption based on session IDs according to RFC 4346 (TLS1.1) or RFC 5246 (TLS1.2), session resumption based on session tickets according to RFC 5077*~~].

FIA_X509_EXT.1.1/Rev
The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using at least one of the following [**Selection:** *the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*~~,~~ and *Certificate Revocation List (CRL) as specified in* ~~*RFC 5759*~~ RFC 8603 *Section* ~~*5*~~7].
- The TSF shall validate the extendedKeyUsage field 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.1.2/Rev
The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2.1
The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for all of the following applicable underlined selections [**Selection:** ~~*DTLS*~~, *HTTPS*, *IPsec*, *SSH*, *TLS*] and [**Selection:** *code signing for system software updates* [***Assignment***: *other uses*], *no additional uses*].

FIA_X509_EXT.2.2
When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall perform at least one of the following [**Selection**: *allow the Administrator to choose whether to accept the certificate in these cases,* ~~*accept the certificate,*~~ *not accept the certificate*].

FIA_X509_EXT.3.1
The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [**Selection:** *device-specific information, Common Name, Organization, Organizational Unit, Country*].

FIA_X509_EXT.3.2
The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

FPT_TUD_EXT.2.1
If X.509 certificate are used to authenticate firmware/software updates, the TSF shall check the validity of the code signing certificate before installing each update.

FPT_TUD_EXT.2.2
If X.509 certificate are used to authenticate firmware/software updates and if revocation information is not available for a certificate in the trust chain that is not a trusted certificate designated as a trust anchor, the TSF shall [**Selection**: *not install the update, allow the Administrator to choose whether to accept the certificate in these cases*].

FPT_TUD_EXT.2.3
If X.509 certificate are used to authenticate firmware/software updates and if the certificate is deemed invalid because the certificate has expired, the TSF shall [**Selection**: *allow the Administrator to choose whether to install the update in these cases, not accept the certificate*].

FPT_TUD_EXT.2.4
If X.509 certificate are used to authenticate firmware/software updates and if the certificate is deemed invalid for reasons other than expiration or revocation information being unavailable, the TSF shall not install the update.

FAU_STG_EXT.1.1
The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1