FDE Interpretation # 201808

Status: ☒*Active* □ *Inactive*

Date: 11-20-2018

Type of Document: ☒*Technical Decision* □ *Technical Recommendation*

Approved by: ☒*FDE iTC Interpretations Team* □ *FDE iTC*

Affected Document(s): FDE AA cPP v2.0, FDE EE cPP v2.0

Affected Section(s): FCS_PCC_EXT.1.1, FCS_RBG_EXT.1.2, FCS_SMC_EXT.1.1

Superseded Interpretation(s):

Issue:

CNSA requires use of SHA-384, it was missed as a selectable in FCS_PCC_EXT.1.1 and FCS_SMC_EXT.1.1 (ECD section only). It should also be added in the app note for FCS_RBG_EXT.1.2.

Resolution:

The FIT acknowledges the issues described in the 'Issue' section above. **Bolding** indicates change.

In the FDE AA cPP v2.0:
The requirement FCS_PCC_EXT.1.1 shall be modified as follows:
A password used by the TSF to generate a password authorization factor shall enable up to [assignment: positive integer of 64 or more] characters in the set of {upper case characters, lower case characters, numbers, and [assignment: other supported special characters]} and shall perform Password-based Key Derivation Functions in accordance with a specified cryptographic algorithm HMAC-[selection: SHA-256, **SHA-384**, SHA-512], with [assignment: positive integer of 1000 or more] iterations, and output cryptographic key sizes [selection: 128 bits, 256 bits] that meet the following: [NIST SP 800-132].

The ECD section requirement FCS_PCC_EXT.1.1 shall be modified as follows:
A password used by the TSF to generate a password authorization factor shall enable up to [assignment: positive integer of 64 or more] characters in the set of {upper case characters, lower case characters, numbers, and [assignment: other supported special characters]} and shall perform Password-based Key Derivation Functions in accordance with a specified cryptographic algorithm HMAC-[selection: SHA-256, **SHA-384,** SHA-512], with [assignment: positive integer of 1000 or more] iterations, and output cryptographic key sizes [selection: 128 bits, 256 bits] that meet the following: [assignment: PBKDF recommendation or specification].

The application note for FCS_RBG_EXT.1.2 shall be modified as follows:
ISO/IEC 18031:2011 contains different methods of generating random numbers; each of these, in turn, depends on underlying cryptographic primitives (hash functions/ciphers). The ST author will select the function used and include the specific underlying cryptographic primitives used in the requirement. While any of the identified hash functions (SHA-256, **SHA-384,** SHA-512) are allowed for Hash_DRBG or HMAC_DRBG, only AES based implementations for CTR_DRBG are allowed. Table C.2 in ISO/IEC 18031:2011 provides an identification of Security strengths, Entropy and Seed length requirements for the AES-128 and 256 Block Cipher.

The CTR_DRBG in ISO/IEC 18031:2011 requires using derivation function, whereas NIST SP 800-90A does not. Either model is acceptable. In the first selection in FCS_RBG_EXT.1.1, the ST author **chooses** the standard to which the TSF is compliant.

In the first selection in FCS_RBG_EXT.1.2 the ST author fills in how many entropy sources are used for each type of entropy source they employ. It should be noted that a combination of hardware and software based noise sources is acceptable.

It should be noted that the entropy source is considered to be a part of the DRBG and if the DRBG is included in the TOE, the developer is required to provide the entropy description outlined in Appendix D. The documentation *and tests* required in the Evaluation Activity for this element necessarily cover each source indicated in FCS_RBG_EXT.1.2. Individual contributions to the entropy pool may be combined to provide the minimum amount of entropy as long as the Entropy Documentation demonstrates that entropy from each of these individual sources is generated independently.

The ECD section application note for FCS_RBG_EXT.1.2 shall be modified as follows:
Application Note: ISO/IEC 18031:2011contains three different methods of generating random numbers; each of these, in turn, depends on underlying cryptographic primitives (hash functions/ciphers). The ST author will select the function used, and include the specific underlying cryptographic primitives used in the requirement. While any of the identified hash functions (SHA-256, **SHA-384,** SHA-512) are allowed for Hash_DRBG or HMAC_DRBG, only AES6 based implementations for CTR_DRBG are allowed.

The ECD section requirement of FCS_SMC_EXT.1.1 shall be modified as follows:
FCS_SMC_EXT.1.1 The TSF shall combine submasks using the following method [selection: exclusive OR (XOR), SHA-256, **SHA-384,** SHA-512] to generate an [assignment: types of keys].

In the FDE EE cPP v2.0:
The application note for FCS_RBG_EXT.1.2 shall be modified as follows:
ISO/IEC 18031:2011 contains different methods of generating random numbers; each of these, in turn, depends on underlying cryptographic primitives (hash functions/ciphers). The ST author will select the function used and include the specific underlying cryptographic primitives used in the requirement. While any of the identified hash functions (SHA-256, **SHA-384**, SHA-512) are allowed for Hash_DRBG or HMAC_DRBG, only AES based implementations for CTR_DRBG are allowed. Table C.2 in ISO/IEC 18031:2011 provides an identification of Security strengths, Entropy and Seed length requirements for the AES-128 and 256 Block Cipher.

The CTR_DRBG in ISO/IEC 18031:2011 requires using derivation function, whereas NIST SP 800-90A does not. Either model is acceptable. In the first selection in FCS_RBG_EXT.1.1, the ST author **chooses** the standard to which the TSF is compliant.

In the first selection in FCS_RBG_EXT.1.2 the ST author fills in how many entropy sources are used for each type of entropy source they employ. It should be noted that a combination of hardware and software based noise sources is acceptable.

It should be noted that the entropy source is considered to be a part of the DRBG and if the DRBG is included in the TOE, the developer is required to provide the entropy description outlined in Appendix D. The documentation *and tests* required in the Evaluation Activity for this element necessarily cover each source indicated in FCS_RBG_EXT.1.2. Individual contributions to the entropy pool may be combined to provide the minimum amount of entropy as long as the Entropy Documentation demonstrates that entropy from each of these individual sources is generated independently.

The requirement FCS_SMC_EXT.1.1 shall be modified as follows:
FCS_SMC_EXT.1.1 The TSF shall combine submasks using the following method [selection: exclusive OR (XOR), SHA-256, **SHA-384,** SHA-512] to generate an [intermediary key or DEK].

The ECD section application note for FCS_RBG_EXT.1.2 shall be modified as follows:
ISO/IEC 18031:2011 contains three different methods of generating random numbers; each of these, in turn, depends on underlying cryptographic primitives collaborative Protection Profile for Full Drive Encryption - Encryption Engine Version 2.0 (hash functions/ciphers). The ST author will select the function used, and include the specific underlying cryptographic primitives used in the requirement. While any of the identified hash functions (SHA-256, **SHA-384,** SHA-512) are allowed for Hash_DRBG or HMAC_DRBG, only AES based implementations for CTR_DRBG are allowed.

The ECD section requirement of FCS_SMC_EXT.1.1 shall be modified as follows:
FCS_SMC_EXT.1.1 The TSF shall combine submasks using the following method [selection: exclusive OR (XOR), SHA-256, **SHA-384,** SHA-512] to generate an [assignment: types of keys].

Rationale:
SHA-384 was intended to be an acceptable selection for all SHA selections.

Further Action:

None.

Action by FDE iTC:

None.