|  |
| --- |
| Proposed Resolution for Some Security CIDs |
| Date: 2015-01-14 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| René Struik | Struik Security Consultancy | Toronto ON | +1 (415) 690-7363+1 (647) 867-5658Skype: rstruik | rstruik.ext@gmail.com |

Abstract

This document discusses some security-related LB#204 comments related to the draft TGai specification.

This relates to the following comments on section 11.11.2.4.1, 11.11.2.4.2 (comments related to AEAD scheme):

6795, 6794, 6793, 6792, 6790, 6789, 6788, 6670, 6786, 6690, 6687, 6686, 6685, 6684, 6679, 6678, 6677, 6676, 6787, 6785, 6805, 6797, 6624

Suggested resolutions are relative to Draft D3.1. {Obviously, comments reference Draft D3.0.}

**Summary sheet: Suggested resolution of a selection of comments from 13/1076r30:**

**DETAILS: Suggested resolution of a selection of comments from 14/1351-22**

**CID #6795:** 11.11.2.4.2, p. 122, l. 10: "If the AEAD cipher requires a unique counter, the current value of the AEAD counter from the PTKSA shall passed to the AEAD encryption algorithm." -- the AEAD cipher for FILS (GCM) requires a unique counter (and missing "be")

Proposed change: Change to "The unique counter required by the AEAD encryption algorithm shall be the current value of the AEAD counter from the PTKSA."

Suggested resolution: Accept.

**CID #6794:** 11.11.2.4.1, p. 120, l. 12: "If the AEAD cipher requires a unique counter, the current value of the AEAD counter from the PTKSA shall be passed to the AEAD encryption algorithm." -- the AEAD cipher for FILS (GCM) requires a unique counter

Proposed change: Change to "The unique counter required by the AEAD encryption algorithm shall be the current value of the AEAD counter from the PTKSA."

Suggested resolution: Accept. See also CID #6795.

**CID #6793:** 11.11.2.4.2, p. 122, l. 19: "If the AEAD cipher mode requires an AEAD counter," -- there's only one AEAD cipher mode, namely GCM, and it requires an AEAD counter

Proposed change: Delete the cited text

Suggested resolution: Accept. See also CID #6795.

**CID #6792:** 11.11.2.4.1, p. 120, l. 20: "If the AEAD cipher mode requires an AEAD counter," -- there's only one AEAD cipher mode, namely GCM, and it requires an AEAD counter

Proposed change: Delete the cited text

Suggested resolution: Accept. See also CID #6795.

**CID #6790:** 11.11.2.4.2, p. 122, l. 25: "the output plaintext replaces the ciphertext as portion of the frame that follows the FILS session element" -- hm, including the FCS?

Proposed change: Clarify (saying MMPDU rather than frame may help); also change "output" to "returned" to match the previous subclause

Suggested resolution: Reject.

Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6789:** 11.11.2.4.1, p. 120, l. 25: "the returned plaintext replaces the ciphertext as portion of the frame that follows the FILS session element" -- hm, including the FCS?

Proposed change: Clarify (saying MMPDU rather than frame may help)

Suggested resolution: Reject. See also CID #6790.

Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6788:** 11.11.2.4.2, p. 122, l. 11: "The ciphertext output by the AEAD algorithm becomes the data that follows the FILS session element in the encrypted and authenticated Association Response frame." Does this mean that the Association Response MMPDU is encrypted, then the AEAD cipher output is spliced into this at the octet position after the FILS session element? This sounds a bit grotesque. And what happens to the FCS at the end of the frame (= MPDU)?

Proposed change: Clarify

Suggested resolution: Reject. See also CID #6790.

The data to be encrypted and authenticated is the data that would be part of the unsecured frame at this stage of outgoing processing and frame formation, i.e., before adding a FCS check, etc. Not sure what is wrong here. Obviously, the FCS is the error detection check sum computed over the frame that would otherwise be ready for sending (i.e., is computed over the frame as final step). Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6670:** 11.11.2.4.1, p. 119, l. 52: Is the intent really to bitwise OR the nonces and MAC addresses?

Proposed change: Change " | " to " || " at the cited location (5 instances)

Suggested resolution: Accept.

The text refers to concatenation, but okay to remove any ambiguity here.

**CID #6786:** 11.11.2.4.2, p. 122, l. 9: "The plaintext passed to the AEAD encryption algorithm is the data that would follow the FILS session element in an unencrypted frame." -- huh? That data would be some other element (or perhaps the FCS, if there are no more elements). What is intended here?

Proposed change: Reword using specifics rather than hypotheticals

Suggested resolution: Reject. See also CID #6790.

The data to be encrypted and authenticated is the data that would be part of the unsecured frame at this stage of outgoing processing and frame formation, i.e., before adding a FCS check, etc. Not sure what is wrong here. Obviously, the FCS is the error detection check sum computed over the frame that would otherwise be ready for sending (i.e., is computed over the frame as final step). Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6690:** 11.11.2.4.1, p. 119, l. 7: "the capability"

Proposed change: Change to "the Capability Information field"

Suggested resolution: Accept. See also CID #6025.

Editorial note: please note that one should make a corresponding change in 11.11.2.4.2, p. 132, l. 29 (last dashed item).

**CID #6687:** 11.11.2.4.2, p. 121, l. 60: "The Association Response frame shall be secured with KEK using the AEAD algorithm as defined in 11.11.2.5" -- how, exactly, is KEK used to secure the frame?

Proposed change: Change to "[...] shall be encrypted using the AEAD algorithm as defined in 11.11.2.5 with the KEK as the key"

Suggested resolution: Accept.

It is indeed more appropriate to first mention the cryptographic mechanism and then the parameters with which this construct is instantiated.

**CID #6686:** 11.11.2.4.1, p. 119, l. 61: "The Association Request frame shall be secured with KEK using the AEAD algorithm as defined in 11.11.2.5" -- how, exactly, is KEK used to secure the frame?

Proposed change: Change to "[...] shall be encrypted using the AEAD algorithm as defined in 11.11.2.5 with the KEK as the key"

Suggested resolution: Accept. See also CID #6687.

It is indeed more appropriate to first mention the cryptographic mechanism and then the parameters with which this construct is instantiated.

**CID #6685:** 11.11.2.4.2, p. 122, l. 19: "If the AEAD cipher mode requires an AEAD counter, the STA implicitly uses the AP's initial AEAD counter of the value 128 followed by 12 octets of

Proposed change: Clarify

Suggested resolution: Accept.

Not sure whether a textual change is required. However, the idea here is that the STA and AP have to instantiate the AEAD cipher, which requires as one of its input parameters a nonce. Here, STA takes nonces starting at the all-zero string, whereas the AP uses nonces starting at the integer 2103, where both increment nonces (by one at a time) in their local state if used more than once. This way, nonces on either end (STA and AP) never clash, since the nonce space is partitioned according to the value of the leftmost bit hereof (set to 0 for STA; set to 1 for AP). Since the AP initially may never have seen the STA, the convention used here is that nonces start at minimum value according to rules indicated above and then simply increment according to local state information, upon reuse.

**CID #6684:** 11.11.2.4.1, p. 120, l. 20: "If the AEAD cipher mode requires an AEAD counter, the AP implicitly uses the STA's initial AEAD counter of all zeros to decrypt and verify the received frame." -- if you can just use an implicit counter why bother maintaining actual counters?

Proposed change: Clarify

Suggested resolution: Accept. See also CID #6685.

Not sure whether a textual change is required. However, the idea here is that the STA and AP have to instantiate the AEAD cipher, which requires as one of its input parameters a nonce. Here, STA takes nonces starting at the all-zero string, whereas the AP uses nonces starting at the integer 2103, where both increment nonces (by one at a time) in their local state if used more than once. This way, nonces on either end (STA and AP) never clash, since the nonce space is partitioned according to the value of the leftmost bit hereof (set to 0 for STA; set to 1 for AP). Since the AP initially may never have seen the STA, the convention used here is that nonces start at minimum value according to rules indicated above and then simply increment according to local state information, upon reuse.

**CID #6679:** 11.11.2.4.2, p. 123, l. 2: "The STA install GTK and set key RSC." -- what does this mean?

Proposed change: Change to "The STA shall install the GTK and set the key RSC."

Suggested resolution: Accept. See also CID #6328.

Editorial note: this seems to already have been incorporated in Draft D3.1.

**CID #6678:** 11.11.2.4.2, p. 122, l. 60: "the STA and AP shall irretrievably destroy the temporary keys" -- what are "the temporary keys"

Proposed change: List the keys which are irretrievably obliterated

Suggested resolution: Revised.

replace: "temporary keys" with the following:

"non-persistent secret keying material that is created by executing the key establishment with FILS shared key authentication scheme (11.11.2.2.1) or the key establishment with FILS public key authentication scheme (11.11.2.2.2)."

**CID #6677:** 11.11.2.4.1, p. 122, l. 16: Is "The STA decrypts and verifies the received Association Response frame with KEK." -- how, exactly, is KEK used to verify the frame?

Proposed change: Add some extra words, or a cross-reference

Suggested resolution: Revise. See also CID #6687.

Replace “[…] with KEK” by “[…] with the AEAD algorithm as defined in 11.11.2.5 with the KEK as the key".

It is indeed more appropriate to first mention the cryptographic mechanism and then the parameters with which this construct is instantiated.

**CID #6676:** 11.11.2.4.1, p. 120, l. 18: "The AP decrypts and verifies the received Association Request frame with KEK." -- how, exactly, is KEK used to verify the frame?

Proposed change: Add some extra words, or a cross-reference

Suggested resolution: Revise. See also CID #6677.

Replace “[…] with KEK” by “[…] with the AEAD algorithm as defined in 11.11.2.5 with the KEK as the key".

It is indeed more appropriate to first mention the cryptographic mechanism and then the parameters with which this construct is instantiated.

**CID #6787:** 11.11.2.4.1, p. 120, l. 13: "The ciphertext output by the AEAD algorithm becomes the data that follows the FILS session element in the encrypted and authenticated Association Request frame." Does this mean that the Association Request MMPDU is encrypted, then the AEAD cipher output is spliced into this at the octet position after the FILS session element? This sounds a bit grotesque. And what happens to the FCS at the end of the frame (= MPDU)?

Proposed change: Clarify

Suggested resolution: Reject. See also CID #6788.

The data to be encrypted and authenticated is the data that would be part of the unsecured frame at this stage of outgoing processing and frame formation, i.e., before adding a FCS check, etc. Not sure what is wrong here. Obviously, the FCS is the error detection check sum computed over the frame that would otherwise be ready for sending (i.e., is computed over the frame as final step). Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6785:** 11.11.2.4.1, p. 120, l. 10: "The plaintext passed to the AEAD encryption algorithm is the data that would follow the FILS session element in an unencrypted frame." -- huh? That data would be some other element (or perhaps the FCS, if there are no more elements). What is intended here?

Proposed change: Reword using specifics rather than hypotheticals

Suggested resolution: Reject. See also CID #6786.

The data to be encrypted and authenticated is the data that would be part of the unsecured frame at this stage of outgoing processing and frame formation, i.e., before adding a FCS check, etc. Not sure what is wrong here. Obviously, the FCS is the error detection check sum computed over the frame that would otherwise be ready for sending (i.e., is computed over the frame as final step). Unsecuring an incoming frame is contingent on the frame passing a check on the frame check sequence (FCS field), after which this FCS is further discarded. Similarly, securing an outgoing frame preceeds calculation of the FCS field. In either case, the FCS field is not part of the data to be authenticated and/or secured (and technically cannot be).

**CID #6805:** 11.11.2.4.2, p. 122, l. 2: "GTK rekeying shall be performed as described in 11.6.7 (Group Key Handshake)." -- what about PTK rekeying

Proposed change: Add some information about PTK rekeying under FILS

Suggested resolution: Reject.

FILS does not use the pairwise key hierarchy of 11.6.1.3 directly; if uses a key derivation function to produce the keys TK, KCK, KEK, without first deriving a PTK.

**CID #6797:** 11.6.1.3: The referenced subclause in the baseline says "Except when preauthentication is used, the pairwise key hierarchy utilizes PRF-384, PRF-512 or PRF-704 to derive session-specific keys from a PMK". FILS is now another exception

Proposed change: Add "or FILS authentication" before "is used"

Suggested resolution: Accept.

The corresponding sentence in the 802.11-2012 specification indeed seems to require updating, as proposed. Editorial note: need to check current numbering in the 802.11m (REV) effort.

**CID #6624:** here are 16 instances of "AEAD counter", but Aren't there two, one for sending stuff to the peer, and one for checking stuff received from the peer? Only two of the 16 instances are "peer's AEAD counter" and the rest are vague

Proposed change: Add some words to indicate which AEAD counter is being used in the 14 vague instances

Suggested resolution: Reject.

It is very hard to act on a comment that seems to be based on a global text string search. It would help if the commenter could give a specific instance where the context of the AEAD counter value is indeed unclear. Right now, it seems that references in the key confirmation section (11.11.2.4) are all unambiguous. Disposition of this comment could change if evidence regarding obvious ambiguities is presented to the group.