IEEE P802.11
Wireless LANs

|  |
| --- |
| FT protocol with FILS AKMs |
| Date: 2018-01-16 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Jouni Malinen | Qualcomm, Inc. |  |  | jouni@qca.qualcomm.com |
|  |  |  |  |  |

Abstract

Contribution 17-906r4 addressed number of issues related to FILS. However, the changes in that document for making FT protocol work with FILS AKMs, i.e., the case of using FILS authentication to derive FT key hierarchy during initial mobility domain association followed by use of FT protocol for subsecuent reassociation, were not complete. They did not describe how the AES-SIV output is encoded in the Reassociation Request/Response frame. An attempt to implement this failed and there did not result in any good and clean way of encoding the data without changing the FTE design significantly.

To address this issue, this contribution proposes an alternatively direction for fixing the issue: derive additional keys (KEK2 and KCK2) to allow the previously defined (from P802.11r) protection mechanism in FTE (AES key wrap for protection GTK and IGTK and CMAC to generate the MIC) without ending up with issues of using keys with multiple different cryptographic operations.

As part of preparing this contribution, a separate item in use of the new Suite B AKM for FT was noticed to not be complete (FTE MIC subfield was not made variable length). That issue is also fixed here since the same change is needed with the one of the FILS+FT AKMs.

[place document body text here]

See 13.8 (FT authentication sequence).

* Fast BSS Transition element (FTE)

*Modify 9.4.2.48 as shown (REVmd/D0.5 page 1089):*

The FTE includes information needed to perform the FT authentication sequence or FILS authentication(11ai) during a fast BSS transition in an RSN. This element is shown in FTE format.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Element ID | Length | MIC Control | MIC | ANonce | SNonce | Optional Parameter(s) |
| Octets: | 1 | 1 | 2 | variable | 32  | 32 | variable |
| * FTE format
 |

The Element ID and Length fields are defined in 9.4.2.1 (General).

The MIC Control field is two octets and is defined in MIC Control field.

|  |  |  |
| --- | --- | --- |
|  | B0                          B7 | B8                                     B15 |
|  | Reserved | Element Count |
| Bits: | 8 | 8 |
| * MIC Control field
 |

The Element Count subfield of the MIC Control field contains the number of elements that are included in the message integrity code (MIC) calculation(#114).

(#114)When the Element Count subfield has a value greater than 0, the MIC field contains a MIC that is calculated using the algorithm specified in 13.8.4 (FT authentication sequence: contents of third message) and 13.8.5 (FT authentication sequence: contents of fourth message). Otherwise, the MIC field contains the value of 0. The length of the MIC field depends on the negotiated AKM selector and is specific in Table 12-9 (Integrity and key-wrap algorithms).

The ANonce field contains a value chosen by the R1KH. It is encoded following the conventions in 9.2.2 (Conventions).

The SNonce field contains a value chosen by the S1KH. It is encoded following the conventions in 9.2.2 (Conventions).

The format of the Optional Parameter(s) field is shown in Optional Parameter(s) field.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Subelement ID | Length | Data |
| Octets: | 1 | 1 | variable |
| * Optional Parameter(s) field
 |

The Subelement ID field is defined in Subelement IDs:

|  |
| --- |
| * Subelement IDs
 |
| Value | Contents of Data field |
| 0 | Reserved |
| 1 | PMK-R1 key holder identifier (R1KH-ID) |
| 2 | GTK subelement |
| 3 | PMK-R0 key holder identifier (R0KH-ID) |
| 4 | IGTK |
| 5–255 | Reserved |

R1KH-ID indicates the identity of the R1KH, which is used by the S0KH and the R0KH for deriving the PMK-R1s. It is encoded following the conventions in 9.2.2 (Conventions).

The GTK subelement contains the group temporal key, which is encrypted (see procedures in 13.8.5 (FT authentication sequence: contents of fourth message)) and is defined in GTK subelement format.

*Replace “16-40” with “24-40” as the Wrapped Key field length in Figure 9-352.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key Info | Key Length | RSC | Wrapped Key |
| Octets: | 1 | 1 | 2 | 1 | 8 | 24–40  |
| * GTK subelement format(#114)
 |

The GTK subelement Key Info subfield is defined in GTK subelement’s Key Info subfield.

|  |  |  |
| --- | --- | --- |
|  | B0                          B1 | B2                                    B15 |
|  | Key ID | Reserved |
| Bits: | 2 | 14 |
| * GTK subelement’s Key Info subfield
 |

Key Length field is the length of the Key field in octets, not including any padding (see 13.8.5 (FT authentication sequence: contents of fourth message)).

RSC field contains the receive sequence counter (RSC) for the GTK being installed. Delivery of the RSC field value allows a STA to identify replayed MPDUs. If the RSC field value is less than 8 octets in length, the remaining octets are set to 0. The least significant octet of the transmit sequence counter (TSC) or packet number (PN) is in the first octet of the RSC field. See Table 12-6 (Key RSC field).

For WEP, the RSC value is reserved.

The Wrapped Key field contains the encrypted GTK as described in 13.8.5 (FT authentication sequence: contents of fourth message).

When sent by a non-AP STA, the R0KH-ID indicates the R0KH with which the S0KH negotiated the PMK‑R0 it is using for this transition. When sent by an AP, the R0KH-ID indicates the R0KH that the S0KH will be using to generate a PMK-R0 security association. It is encoded following the conventions from 9.2.2 (Conventions).

The IGTK field contains the Integrity GTK, used for protecting robust Management frames. The IGTK subelement format is shown in IGTK subelement format.

*Replace “16-40” with “24-40” as the Wrapped Key field length in Figure 9-354.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | IPN | Key Length | Wrapped Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | 24-40 |
| * IGTK subelement format(#114)
 |

The Key ID field indicates the value of the BIP key identifier.

The IPN field indicates the receive sequence counter for the IGTK being installed, to allow a STA to identify replayed protected group addressed robust Management frames.

The Key Length field is the length of IGTK in octets, not including any padding (see 13.8.5 (FT authentication sequence: contents of fourth message)).

The Wrapped Key field contains the wrapped IGTK being distributed. The length of the resulting AES-Keywrapped IGTK in the Wrapped Key field is Key Length + 8 octets.

* Keys and key distribution
* Key hierarchy
* FT key hierarchy
* PTK

*Modify 12.7.1.7.5 as shown (REVmd/D0.5 page 2442):*

The third-level key in the FT key hierarchy is the PTK. When FILS authentication is used to establish the FT key hierarchy, PTK for the initial mobility domain association is derived as part of the FILS authentication as defined in 12.12.2.5.3 (PTKSA Key derivation with FILS authentication). Otherwise, this(11ai) key is mutually derived by the S1KH and the R1KH used by the target AP, with the key length being a function of the negotiated cipher suite as defined by Table 12-5 (Cipher suite key lengths) in 12.7.2 (EAPOL-Key frames).

Using the KDF defined in 12.7.1.7.2 (Key derivation function (KDF)), the PTK derivation is as follows:

PTK = KDF-Hash-Length(PMK-R1, "FT-PTK", SNonce || ANonce || BSSID || STA-ADDR)

where

* KDF-Hash-Length is the KDF as defined in 12.7.1.7.2 (Key derivation function (KDF)) using the hash algorithm identified by the AKM suite selector (see Table 9-144 (AKM suite selectors)).
* PMK-R1 is the key that is shared between the S1KH and the R1KH.
* SNonce is a 256-bit random bit string contributed by the S1KH.
* ANonce is a 256-bit random bit string contributed by the R1KH.
* STA-ADDR is the non-AP STA’s MAC address.
* BSSID is the BSSID of the target AP.
* Length is the total number of bits to derive, i.e., number of bits of the PTK. The length is dependent on the negotiated cipher suites and AKM suites as defined by Table 12-5 (Cipher suite key lengths) in 12.7.2 (EAPOL-Key frames) and Integrity and key-wrap algorithms in EAPOL-Key frame construction and processing.

Each PTK has five component keys, KCK, KEK, a temporal key, KCK2, and KEK2 derived as follows:

The KCK shall be computed as the first KCK\_bits bits (bits 0 to KCK\_bits–1) of the PTK:

KCK = L(PTK, 0, KCK\_bits).

The KCK is used to provide data origin authenticity in EAPOL-Key frames, as defined in 12.7.2 (EAPOL-Key frames), and in the FT authentication sequence, as defined in 13.8 (FT authentication sequence).

The KEK shall be computed as the next KEK\_bits of the PTK:

KEK = L(PTK, KCK\_bits, KEK\_bits)

The KEK is used to provide data confidentiality for certain fields (KeyData) in EAPOL-Key frames, as defined in 12.7.2 (EAPOL-Key frames), and in the FT authentication sequence, as defined in 13.8 (FT authentication sequence).

The temporal key (TK) shall be computed as the next TK\_bits (see Table 12-5 (Cipher suite key lengths)) bits of the PTK:

TK = L(PTK, KCK\_bits+KEK\_bits, TK\_bits)

The KCK2 shall be computed as the next KCK2\_bits of the PTK:

KCK2 = L(PTK, KCK\_bits+KEK\_bits+TK\_bits, KCK2\_bits).

The KCK2 is used to provide data origin authenticity in the FT authentication sequence, as defined in 13.8 (FT authentication sequence).

The KEK2 shall be computed as the next KEK2\_bits of the PTK:

KEK2 = L(PTK, KCK\_bits+KEK\_bits+TK\_bits+KCK2\_bits, KEK2\_bits)

The KEK2 is used to provide data confidentiality for certain fields in the FT authentication sequence, as defined in 13.8 (FT authentication sequence).

For vendor-specific cipher suites, the length of the temporal key (and the value of Length) depend on the vendor-specific algorithm.

The temporal key is configured into the STA by the SME through the use of the MLME-SETKEYS.request primitive. The STA uses the temporal key with the pairwise cipher suite; interpretation of this value is specific to the cipher suite.

The PTK is referenced and named as follows:

PTKName = Truncate-128(SHA-256(PMKR1Name || “FT-PTKN” || SNonce || ANonce || BSSID || STA‑ADDR))

where

* "FT-PTKN" is treated as an ASCII string.

The PTKName is used to identify the PTK.

* EAPOL-Key frame construction and processing

*Modify 12.7.3 as shown (REVmd/D0.5 page 2452):*

EAPOL-Key frames are constructed and processed according to the AKM negotiated at association time. The negotiated AKM determines what algorithm is used to construct and verify a MIC, the size of the MIC, and the algorithm used to wrap and unwrap the Key Data field.

Table 12-9 (Integrity and key-wrap algorithms) indicates the particular algorithms to use when constructing and processing EAPOL-Key frames and FT authentication sequence. The AKM of “Deprecated” indicates an AKM of 00-0F-AC:1 or 00-0F-AC:2 when either TKIP or “Use group cipher suite” is the negotiated pairwise cipher. For all other AKMs the negotiated pairwise cipher suite does not influence the algorithms used to process EAPOL-Key frames. For the 00-0F-AC:16 and 00-0F-AC:17 AKMs (FILS with FT), different keys and algorithms are used in EAPOL-Key frames and FT authentication sequence. These different cases are indicated in the table in <EAPOL-Key> / <FT authentication> format.

|  |  |  |
| --- | --- | --- |
| * Integrity and key-wrap algorithms
 |  |  |
| AKM | Integrity algorithm | KCK\_bits | Size of MIC | Key-wrap algorithm | KEK\_bits | KCK2\_bits | KEK2\_bits |
| 00-0F-AC:1 | HMAC-SHA-1-128 | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:2  | HMAC-SHA-1-128 | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:3 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:4 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:5 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:6 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:8 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:9 | AES-128-CMAC | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:11 | HMAC-SHA-256 | 128 | 16 | NIST AES Key Wrap | 128 | 0 | 0 |
| 00-0F-AC:12 | HMAC-SHA-384 | 192 | 24 | NIST AES Key Wrap | 256 | 0 | 0 |
| 00-0F-AC:13 | HMAC-SHA-384 | 192 | 24 | NIST AES Key Wrap | 256 | 0 | 0 |
| 00-0F-AC:14 | AES-SIV-256 | 0 | 0 | AES-SIV-256 | 256 | 0 | 0 |
| 00-0F-AC:15 | AES-SIV-512 | 0 | 0 | AES-SIV-512 | 512 | 0 | 0 |
| 00-0F-AC:16 | AES-SIV-256 / AES-128-CMAC | 0 | 0 / 16 | AES-SIV-256 / NIST AES Key Wrap | 256 | 128 | 128 |
| 00-0F-AC:17 | AES-SIV-512 / HMAC-SHA-384 | 0 | 0 / 24 | AES-SIV-512 / NIST AES Key Wrap | 512 | 192 | 256 |

* Fast BSS transition

13.8 FT authentication sequence

* FT authentication sequence: contents of third message

*Modify 13.8.4 as shown (REVmd/D0.5 page 2555):*

The RSNE shall be present only if dot11RSNAActivated is true. If present, the RSNE shall be set as follows:

* Version field shall be set to 1.
* PMKID Count field shall be set to 1.
* PMKID field shall contain the PMKR1Name.
* All other fields shall be as specified in 9.4.2.25 (RSNE) and 12.6.3 (RSNA policy selection in an infrastructure BSS).

The MDE shall contain the MDID and FT Capability and Policy fields. This element shall be identical to the MDE contained in the first message of this sequence.

The FTE shall be present only if dot11RSNAActivated is true. If present, the FTE shall be set as follows:

* ANonce, SNonce, R0KH-ID, and R1KH-ID shall be set to the values contained in the second message of this sequence.
* The Element Count field of the MIC Control field shall be set to the number of elements protected in this frame (variable).
* When the negotiated AKM is 00-0F-AC:3, 00-0F-AC:4, or 00-0F-AC:9, the MIC shall be calculated using the KCK and the AES-128-CMAC algorithm. The output of the AES-128-CMAC shall be 128 bits.
* When the negotiated AKM is 00-0F-AC:13, the MIC shall be calculated using the KCK and the HMAC-SHA-384 algorithm. The output of the HMAC-SHA-384 shall be truncated to 192 bits.
* (#114)When the negotiated AKM is 00-0F-AC:16, the MIC shall be calculated using the KCK2 and the AES-128-CMAC algorithm. The output of the AES-128-CMAC shall be 128 bits.
* When the negotiated AKM is 00-0F-AC:17, the MIC shall be calculated using the KCK2 and the HMAC-SHA-384 algorithm. The output of the HMAC-SHA-384 shall be truncated to 192 bits.
* The MIC shall be calculated on the concatenation of the following data, in the order given here:
* FTO’s MAC address (6 octets)
* Target AP’s MAC address (6 octets)
* Transaction sequence number (1 octet), which shall be set to the value 5 if this is a Reassociation Request frame and, otherwise, set to the value 3
* RSNE
* MDE
* FTE, with the MIC field of the FTE set to 0
* Contents of the RIC-Request (if present)
* All other fields shall be set to 0.

If resources are being requested by the FTO, then a sequence of elements forming the RIC‑Request shall be included.

* FT authentication sequence: contents of fourth message

*Modify 13.8.5 as shown (REVmd/D0.5 page 2555):*

If the status code is SUCCESS, then the following rules apply.

The RSNE shall be present only if dot11RSNAActivated is true. If present, the RSNE shall be set as follows:

* Version field shall be set to 1.
* PMKID Count field shall be set to 1.
* PMKID field shall contain the PMKR1Name
* All other fields shall be identical to the contents of the RSNE advertised by the target AP in Beacon and Probe Response frames.

The MDE shall contain the MDID and FT Capability and Policy fields. This element shall be identical to the MDE contained in the second message of this sequence.

The FTE shall be present only if dot11RSNAActivated is true. If present, the FTE shall be set as follows:

* ANonce, SNonce, R0KH-ID, and R1KH-ID shall be set to the values contained in the second message of this sequence.
* The Element Count field of the MIC Control field shall be set to the number of elements protected in this frame (variable).
* When this message of the authentication sequence appears in a Reassociation Response frame, the Optional Parameter(s) field in the FTE may include the GTK and IGTK subelements. If a GTK or an IGTK are included, the Key field of the subelement shall be encrypted using KEK (when the negotiated AKM is 00-0F-AC:3, 00-0F-AC:4, 00-0F-AC:9, or 00-0F-AC:13) or KEK2 (when the negotiated AKM is 00-0F-AC:16 or 00-0F-AC:17) and the NIST AES key wrap algorithm. The Key field shall be padded before encrypting if the key length is less than 16 octets or if it is not a multiple of 8. The padding consists of appending a single octet 0xdd followed by zero or more 0x00 octets. When processing a received message, the receiver shall ignore this trailing padding. Addition of padding does not change the value of the Key Length field. Note that the length of the encrypted Key field can be determined from the length of the GTK or IGTK subelement.
* When the negotiated AKM is 00-0F-AC:3, 00-0F-AC:4, or 00-0F-AC:9, the MIC shall be calculated using the KCK and the AES-128-CMAC algorithm. The output of the AES-128-CMAC algorithm shall be 128 bits.
* When the negotiated AKM is 00-0F-AC:13, the MIC shall be calculated using the KCK and the HMAC-SHA-384 algorithm. The output of the HMAC-SHA-384 shall be truncated to 192 bits.
* (#114)When the negotiated AKM is 00-0F-AC:16, the MIC shall be calculated using the KCK2 and the AES-128-CMAC algorithm. The output of the AES-128-CMAC shall be 128 bits.
* When the negotiated AKM is 00-0F-AC:17, the MIC shall be calculated using the KCK2 and the HMAC-SHA-384 algorithm. The output of the HMAC-SHA-384 shall be truncated to 192 bits.
* The MIC shall be calculated on the concatenation of the following data, in the order given here:
* FTO’s MAC address (6 octets)
* Target AP’s MAC address (6 octets)
* Transaction sequence number (1 octet), which shall be set to the value 6 if this is a Reassociation Response frame or, otherwise, set to the value 4
* RSNE
* MDE
* FTE, with the MIC field of the FTE set to 0
* Contents of the RIC-Response (if present)
* All other fields shall be set to 0.

If this message is other than a Reassociation Response frame and dot11RSNAActivated is false, a TIE may appear. If this message is other than a Reassociation Response frame, includes a RIC-Response, and dot11RSNAActivated is false, then a timeout interval shall appear. If it appears, it shall be set as follows:

* Timeout Interval Type field shall be set to 1 (reassociation deadline)
* Timeout Interval Value field shall be set to the reassociation deadline time.

If resources were requested by the FTO, then a RIC-Response shall be included.