IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Proposed Modifications for FT with FILS Association | | | | |
| Date: 2014-11-06 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Soo Bum Lee  Jouni Malinen  George Cherian  Santosh Abraham | Qualcomm | 5775 Morehouse Dr., San Diego, CA 92121 | +1 858 651 6645 | gcherian@qti.qualcomm.com |
|  |  |  |  |  |

Abstract

This document proposes modifications to allow FT association with FILS.

4 General description

* Overview of the services
* Access control and data confidentiality services

*Change 4.5.4.8 as follows*

**4.5.4.8 Fast BSS transition**

The FT mechanism defines a means for a STA to set up security and QoS parameters prior to reassociation

to a new AP. This mechanism allows time-consuming operations to be removed from the time-critical

reassociation process within a mobility domain. When the FILS authentication is used during a handover across mobility domains, the overhead incurred during the FT initial mobility domain association in an RSN is further reduced.

* Frame formats

8.3.3.11 Authentication frame format

***Change Table 8-44 as follows:***

|  |  |  |  |
| --- | --- | --- | --- |
| * Presence of fields and[to match REVmc] elements in Authentication frames [CID 2287] | | | |
| Authentication algorithm | Authentication transaction sequence no. | Status code | Presence of fields 4-20 |
| FILS | 1 | Reserved [CID 3162] | The FILS Session element is present. [CID 2817]  [13/1514r1]The FILS Authentication Type is present.  The FILS Nonce is present.  The RSNE is present. [14/0341r5]  The FILS Wrapped Data element is present if FILS shared key authentication is used.   The Finite cyclic group is present if FILS Authentication Type field indicates PFS or if FILS public key authentication is used. [CID 4968 multiple places]   The Element is present if FILS Authentication Type field indicates PFS or if FILS public key authentication is used. [13/1354r2]   The PMKID list is present if STA is asserting cached PMKs.  The Mobility Domain element is present if FILS authentication is used for FT initial mobility domain association. [14/0052r2] |
| FILS | 2 | Status | The FILS Session element is present. [CID 2817]  The RSNE is present. [14/0341r5]  The [13/1514r1] FILS Authentication Type is present if Status is 0.  The FILS Nonce is present if Status is 0.  The FILS Wrapped Data element is present if Status is 0 and FILS shared key authentication is used.  The Finite cyclic group is present if FILS Authentication Type field indicates PFS or if FILS public key authentication is used.  The Element is present if FILS Authentication Type field indicates PFS or if FILS public key authentication is used.  [CID 3036, 13/1354r2]]  The PMKID list is present if AP agrees to perform PMK caching. [14/0052r2]  The Association Timeout Info element is present if AP expects time to transmit Association Response exceeds 1 TU.  The Mobility Domain element and Fast BSS Transition element are present if Status is 0 and FILS authentication is used for FT initial mobility domain association. [14/0791r1] |

* Management frame body components

**8.4.2 Information Elements**

**8.4.2.24 RSNE**

* AKM suites

***Insert two new rows in Table 8-139 as shown, adjusting numbers appropriately:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table 8-139--AKM suite selectors | | | | |
| OUI | Suite type |  | Meaning |  |
| Authentication type | Key management type | Key derivation type |
| 00-0F-AC | <ANA-3> | FT authentication over FILS with SHA-256 and GCM-128 | FT over FILS key management as defined in 11.6.1.7 | Defined in  11.6.1.7.2 |
| 00-0F-AC | <ANA-4> | FT authentication over FILS with SHA-384 and GCM-256 | FT over FILS key management as defined in 11.6.1.7 | Defined in  11.6.1.7.2 |

* Security
* Keys and key distribution

11.6.1 Key hierarchy

**11.6.1.7 FT key hierarchy**

***Change 11.6.1.7.1 as follows:*** [14/0341r5]

**11.6.1.7.1 Overview**

This subclause describes the FT key hierarchy and its supporting architecture. The FT key hierarchy is

designed to allow a STA to make fast BSS transitions between APs without the need to perform an SAE or

IEEE 802.1X authentication at every AP within the mobility domain.

The FT key hierarchy can be used with SAE, IEEE 802.1X authentication, PSK authentication, or FILS authentication.

A three-level key hierarchy provides key separation between the key holders. The FT key hierarchy for the Authenticator is shown in Figure 11-31 (FT key hierarchy at an Authenticator). An identical key hierarchy exists for the Supplicant, and identical functions are performed by the corresponding S0KH and S1KH.

The FT key hierarchy shown in Figure 11-27 consists of three levels whose keys are derived using the key

derivation function (KDF) described in 11.6.1.7.2 as follows:

a) PMK-R0 – the first-level key of the FT key hierarchy. This key is derived as a function of the master

session key (MSK) or PSK. It is stored by the PMK-R0 key holders, R0KH and S0KH.

b) PMK-R1 – the second-level key of the FT key hierarchy. This key is mutually derived by the S0KH

and R0KH.

c) PTK – the third-level key of the FT key hierarchy that defines the IEEE 802.11 and IEEE 802.1X

protection keys. The PTK is mutually derived by the PMK-R1 key holders, R1KH and S1KH.

As shown in Figure 11-27, the R0KH computes the PMK-R0 from the key obtained from SAE

authentication (for the purposes of FT this key is identified as the Master PMK, or MPMK), from the PSK,

from the MSK resulting (per IETF RFC 3748-2004 [B38] ) from a successful IEEE Std 802.1X authentication

between the AS and the Supplicant, or from the IKM (Intermediate Keying Material) (see 11.11.2.3.1) resulting from a successful FILS authentication. Upon a successful authentication, the R0KH shall delete any priorPMK-R0 security association for this mobility domain pertaining to this S0KH. The R0KH shall also delete all PMK-R1 security associations derived from that prior PMK-R0 security association. The PMK-R1s are generated by the R0KH and are assumed to be

delivered from the R0KH to the R1KHs within the same mobility domain. The PMK-R1s are used for PTK generation. Upon receiving a new PMK-R1 for an S0KH, an R1KH deletes the prior PMK-R1 security association and PTKSAs derived from the prior PMK-R1.

***Change Figure 11-31 to the following***



**Figure 11-31 – FT key hierarchy at an Authenticator**

***Change 11.6.1.7.2 as follows***

**11.6.1.7.2 Key derivation function (KDF)**

The KDF for the FT key hierarchy, and for AKMs 00-0F-AC:11, ~~and~~ 00-0F-AC:12, 00-0F-AC:<ANA-1>, 00-0F-AC:<ANA-2>, 00-0F-AC:<ANA-3>, and 00-0F-AC:<ANA-4> is a variant of the pseudorandom function (PRF) defined in 11.6.1.2 (PRF) and is defined as follows:

***Change 11.6.1.7.3 as follows***

**11.6.1.7.3 PMK-R0**

The first-level key in the FT key hierarchy, PMK-R0, is derived using the KDF defined in 11.6.1.7.2. The

PMK-R0 is the first level 256-bit keying material used to derive the next level keys (PMK-R1s):

R0-Key-Data = KDF-Hash-Z(XXKey, "FT-R0", SSIDlength || SSID || MDID || R0KHlength || R0KH-ID

|| S0KH-ID)

PMK-R0 = L(R0-Key-Data, 0, Q)

PMK-R0Name-Salt = L(R0-Key-Data, Q, 128)

where

— KDF-Hash-Z is the KDF as defined in 11.6.1.7.2 used to generate a key of length 384 bits.

— L(-) is defined in 11.6.1.

— If the AKM negotiated is 00-0F-AC:3, then Hash shall be SHA256, Z shall be 384, Q shall be 256, and XXKey shall be the second 256 bits of the MSK (which is derived from the IEEE Std 802.1X authentication), i.e., XXKey = L(MSK, 256, 256). If the AKM negotiated is 00- 0F-AC:4, then Hash shall be SHA256, Z shall be 384, Q shall be 256, and XXKey shall be the PSK. If the AKM negotiated is 00-0F-AC:9, then Hash shall be SHA256, Z shall be 384, Q shall be 256, and XXKey shall be the MPMK generated as the result of SAE authentication. If the AKM negotiated is 00-0F-AC:13, then Hash shall be SHA384, Z shall be 512, Q shall be 384, and XXKey shall be the first 384 bits of the MSK (which is derived from the IEEE 802.1X authentication), i.e., XXKey = L(MSK, 0, 384). If the AKM negotiated is 00-0F-AC:<ANA-3>, then Hash shall be SHA384, Z shall be 384, Q shall be 256, and XXKey shall be the FILS-FT described in 11.11.2.3.1. If the AKM negotiated is 00-0F-AC:<ANA-4>, then Hash shall be SHA384, Z shall be 512, Q shall be 384, and XXKey shall be the FILS-FT described in 11.11.2.3.1.

***Change 11.6.1.7.4 as follows***

**11.6.1.7.4 PMK-R1**

The second-level key in the FT key hierarchy, PMK-R1, is a 256-bit key used to derive the PTK. The PMK- R1 is derived using the KDF defined in 11.6.1.7.2 (Key derivation function (KDF)):

PMK-R1 = KDF-Hash-Z(PMK-R0, "FT-R1", R1KH-ID || S1KH-ID)

where

— KDF-Hash-Z is the KDF as defined in 11.6.1.7.2 (Key derivation function (KDF)).

— If the AKM negotiated is 00-0F-AC:3, 00-0F-AC:4, 00-0F-AC:9, or 00-0F-AC:<ANA-3>, then Hash shall be SHA256, and Z shall be 256. If the AKM negotiated is 00-0F-AC:13 or 00-0F-AC:<ANA-4>, then Hash shall be SHA384, and Z shall be 384.

***Change 11.6.1.7.5 as follows:*** [14/0341r5]

**11.6.1.7.5 PTK**

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 11.11.2.3.2. Otherwise, this 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 11-4 (Cipher suite key lengths) in 11.6.2 (EAPOL-Key frames).

***Change 11.6.2 as follows***

* EAPOL-Key frames
* **Key Information.** This field is 2 octets and specifies characteristics of the key. See Figure 11-33 (Key Information bit layout).
* Key MIC (bit 8): When AKM is not 00-0F-AC<ANA-1>, 00-0F-AC<ANA-2>, 00-0F-AC:<ANA-3>, or 00-0F-AC:<ANA-4>, this bit is set to 1 if a MIC is in this EAPOL-Key frame and is set to 0 if this message contains no MIC. When using an AEAD cipher this bit is set to 0. [REVmc] [CIDs 4321, 4320]
* **EAPOL-Key IV**. This field is 16 octets. It contains the IV used with the KEK. It shall contain 0 when an IV is not required. When AKM is not 00-0F-AC<ANA-1>, 00-0F-AC<ANA-2>, 00-0F-AC:<ANA-3>, or 00-0F-AC:<ANA-4>, ~~I~~it should be initialized by taking the current value of the global key counter (see 11.6.11 (RSNA Authenticator key management state machine)) and then incrementing the counter. Note that only the lower 16 octets of the counter value are used. When the AKM is 00-0F-AC:<ANA-1>, 00-0F-AC:<ANA-2>, 00-0F-AC:<ANA-3>, or 00-0F-AC:<ANA-4>, the current value of the AEAD counter from the PTKSA is copied to the left-most 13 octets of this field. [CIDs 4321, 4320 ]

**Key MIC.** When AKM is not 00-0F-AC<ANA-1>, 00-0F-AC<ANA-2>, 00-0F-AC:<ANA-3>, or 00-0F-AC:<ANA-4>, ~~T~~the EAPOL Key MICis a MIC of the EAPOL-Key frames, from and including the EAPOL protocol version field to and including the Key Data field, calculated with the Key MIC field set to 0. If the Encrypted Key Data subfield (of the Key Information field) is 1, the Key Data field is encrypted prior to computing the MIC. When using an AEAD cipher, the EAPOL Key MIC is not used. The length of this field depends on the negotiated AKM as defined in 11.6.3 (EAPOL-Key frame construction and processing). [14/0565r15 ([CIDs 4321, 4320 ]CID 4322)]

**11.6.3 EAPOL-Key frame construction and processing**

***Insert two new rows in Table 11-8 as shown, adjusting numbers appropriately:***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table 11-8—Integrity and Key Wrap Algorithms | | |  |  |  |
| AKM | Integrity Algorithm | KCK bits | Size of MIC | Key-wrap algorithm | KEK bits |
| 00-0F-AC:<ANA-3> | AES-GCM-128 | 256 | 0 | AES-GCM-128 | 128 |
| 00-0F-AC:<ANA-4> | AES-GCM-256 | 384 | 0 | AES-GCM-256 | 256 |

***Change 11.11.2.3 as follows***

**11.11.2.3 Key derivation with FILS authentication**

When not using PMKSA caching, a PMK is created using the Extract function of RFC 5869. When using PMKSA caching, a new PMKSA is not created. Instead, the PMKSA used for PMKSA caching remains and continues to be identified by the appropriate PMKID. Regardless of whether PMKSA caching is used or not, a PTKSA shall be generated with each FILS authentication exchange.

PTKSA creation uses the KDF from 11.6.1.7.2 (Key derivation function (KDF)) to derive the following keys from the PMK: -a key confirmation key (KCK); a key encryption key (KEK); and a temporal key (TK).

When the AKM used is 00-0F-AC:<ANA-1> or 00-0F-AC:<ANA-3> the hash algorithm used for the FILS-FT, PMKSA and PTKSA creation shall be SHA256 and when the AKM used is 00-0F-AC:<ANA-2> or 00-0F-AC:<ANA-4> the hash algorithm used for the FILS-FT, PMKSA and PTKSA creation shall be SHA384

***Change 11.11.2.3.1 as follows***

**11.11.2.3.1 PMKSA key derivation with FILS authentication**

The Extract function used to derive the PMK takes the two nonces as salt and the secret(s) from FILS Key establishment as input keying material. A PMKID used to identify the PMKSA is generated using the hash algorithm from the negotiated AKM on input data specific to the FILS Key Establishment step. The length of the PMK shall be either 256 bits or 382 bits depending on the negotiated AKM, and the length of the PMKID shall be 128 bits:

PMK = HMAC-Hash(SNonce || ANonce, IKM)

PMKID = Truncate-128(Hash(Context))

When the AKM negotiated is 00-0F-AC:<ANA-3> or 00-0F-AC:<ANA-4>, the FILS-FT is derived using the two nonces and the secret(s) from FILS Key establishment as input keying material. The FILS-FT is used to establish the FT key hierarchy as described in 11.6.1.7.

FILS-FT = KDF-Hash-Z(IKM, “FILS-FT”, SNonce || ANonce)

Where:

* KDF-Hash-Z is the KDF as defined in 11.6.1.7.2 (Key derivation function (KDF)). If the AKM negotiated is 00-0F-AC:<ANA-3>, then Z shall be 256. If the AKM negotiated is 00-0F-AC:<ANA-4>, then Z shall be 384. [14/0341r5]
* SNonce is the STA nonce and ANonce is the AP nonce
* IKM is one of:
* The rMSK is FILS Shared Key authentication was performed without PFS,
* A concatenation of the rMSK and the shared secret, ss, in that order if FILS Shared Key authentication was performed with PFS, or
* The shared secret, ss, if FILS Public Key authentication was performed.
* Context is either:
* The EAP Init/Re-Auth packet when FILS Shared Key authentication is being performed without PFS, or
* A concatenation of the STA’s Diffie-Hellman public value and the AP’s Diffie-Hellman public value, in that order, in all other cases.

***Change 11.11.2.3.2 as follows***

* PTK key derivation with FILS authentication [14/0692r3]

For PTKSA key generation, the inputs to the KDF are the PMK of the PMKSA, a constant label, and a concatenation of the STA MAC address, the AP’s BSSID, the STA’s nonce, and the AP’s nonce. When the AKM used is 00-0F-AC:<ANA-1> or 00-0F-AC:<ANA-3>, the length of KEK shall be 128 bits, and the length of the KCK 256 bits. When the AKM used is 00-0F-AC:<ANA-2> or 00-0F-AC:<ANA-4> the length of the KEK shall be 256 bits, and the length of KCK shall be 384 bits, The total amount of bits extracted from the KDF shall therefore be 384+TK or 640+TK bits depending on the AKM used, where TK\_bits is determined from Table 11-4 (Cipher suite key lengths). [14/0692r3][13/1354r2] [14/0341r5]

KCK || KEK || TK = KDF-X(,PMK “PTKSA Derivation”, SPA ||AA || ANonce) [14/0341r5][14/0692r3, CID 4323]

Where:

* X is 384+TK\_bits or 640+TK bits from Table 11-4 (Cipher suite key lengths) depending on the AKM used. [4393]
* PMK is the PMK from the PMKSA, either created from an initial FILS connection or from a cached PMKSA, when PMKSA caching is used. [14/0341r5]
* SPA is the STA’s MAC address and the AA is the AP’s BSSID.
* SNonce is the STA’s nonce and ANonce is the AP’s nonce. [14/0692r3][13/1354r2] [14/0341r5]

If the negotiated AKM is 00-0F-AC:<ANA-1>, 00-0F-AC:<ANA-2>, 00-0F-AC:<ANA-3>, or 00-0F-AC:<ANA-4>, FILS requires an additional element: a 13 octet AEAD counter to be part of the newly created PTKSA. The STA shall set the AEAD counter to 13 octets of zero and the AP shall set the first octet to the value 128 and the remaining octets to zero (i.e. the first bit of the AEAD counter is 1 and the rest of the bits in the counter are 0). To allow for proper processing, each side shall include the AEAD counter of the other as a peer's AEAD counter (see AEAD cipher mode for FILS ). [14/0341r5][CID 4394]

***Change 11.11.2.5 as follows***

* AEAD cipher mode for FILS [14/0341r5]

FILS authentication uses an AEAD cipher mode to protect (Re)Association and EAPOL-Key frames after FILS key establishment. The AEAD cipher mode is determined by the specific FILS AKM negotiated.

AES-GCM-128 is used if the AKM is 00-0F-AC:<ANA-1> or 00-0F-AC:<ANA-3> and AES-GCM-256 is used if the AKM is 00-0F-AC:<ANA-2> or 00-0F-AC:<ANA-4>.AES-GCM-X (in Table 8-113) is GCM with X-bit AES key. [14/1278r3] [CIDs 4962, 4073multiple places][CID 4847]

**12. Fast BSS transition**

**12.2 Overview**

***Change 12.2.2 as shown***

**12.2.2 Authenticator key holders**

The R0KH and R1KH are responsible for the derivation of keys in the FT key hierarchy. For fast BSS

transition, the functions of the IEEE 802.1X Authenticator are distributed among the R0KH and R1KHs.

The R0KH interacts with the IEEE 802.1X Authenticator to receive the MSK resulting from an EAP

Authentication. The R1KH interacts with the IEEE 802.1X Authenticator to open the Controlled Port. Both the R0KH and R1KH interactions with the IEEE 802.1X Authenticator occur within the SME.

The R0KH derives the PMK-R0 for use in the mobility domain utilizing the MSK (when the AKM

negotiated is 00-0F-AC:3), the PSK (when the AKM negotiated is 00-0F-AC:4) or the PMK (when the

AKM negotiated is 00-0F-AC:9) or the IKM (when the AKM negotiated is 00-0F-AC:<ANA-3> or 00-0F-AC:<ANA-4>). The R0KH shall be responsible for deriving a PMK-R1 for each R1KH within the mobility domain.

***Change 12.2.3 as shown***

**12.2.3 Supplicant key holders**

The S0KH and S1KH are responsible for the derivation of keys in the FT key hierarchy. The S0KH and

S1KH are entities that are assumed to physically reside in the Supplicant.

The S0KH interacts with the IEEE 802.1X functional block (see Figure 4-14 in 4.9) to receive the MSK

resulting from an EAP authentication or the IKM resulting from a FILS authentication. The S1KH interacts with 802.1X to open the Controlled Port. Both the S0KH and S1KH interactions with 802.1X occur within the SME of a STA.

The S0KH derives the PMK-R0 for use in the mobility domain utilizing the MSK (when the AKM

negotiated is 00-0F-AC:3), the PSK (when the AKM negotiated is 00-0F-AC:4) or the PMK (when the

AKM negotiated is 00-0F-AC:9) or the IKM (when the AKM negotiated is 00-0F-AC:<ANA-3> or 00-0F-AC:<ANA-4>).

***Insert a new subclause after subclause 12.4.3 as follows:***

**12.4.4 FT initial domain association over FILS in an RSN**

STA may perform FT initial mobility domain association with an AP using FILS Authentication as specified in this clause if it receives MDE and FILS Indication element in the Beacon or Probe Response frame from the AP.

A STA indicates its support for the FT procedures by including the MDE in the Authentication frame and indicates its support of security in the RSNE. To establish FT key hierarchy, the AP responds by including the FTE, MDE, and RSNE in the Authentication frame. At the end of the sequence, the FT key hierarchy has been established. The message flow is shown in Figure 12-ai1.

To establish FT key hierarchy, the STA shall send an Authentication frame that includes the MDE to the AP. The contents of the MDE shall be the values advertised by the AP in its Beacon or Probe Response frames. Additionally, the STA includes its security capabilities in the RSNE.

If the contents of the MDE received by the AP do not match the contents advertised in the Beacon and Probe Response frames, the AP shall reject the Authentication frame with status code 54 (“Invalid MDE”). If an MDE is present in the Authentication frame and the contents of the RSNE do not indicate a negotiated AKM of Fast BSS Transition over FILS (suite type 00-0F-AC:<ANA-3> or 00-0F-AC:<ANA-4>), the AP shall reject the Authentication frame with status code 43 (“Invalid AKMP”).

Upon successful completion of FILS authentication request processing, the R0KH on the AP uses the IKM (see 11.11.2.3.1) to establish key hierarchy. If a key hierarchy already exists for this STA belonging to the same mobility domain (i.e., having the same MDID), the R0KH shall delete the existing PMK-R0 security association and PMK-R1 security associations. It then calculates the PMK-R0, PMKR0Name, and PMK-R1 and makes the PMK-R1 available to the R1KH of the AP with which the STA is associated.

Then, the AP shall construct an Authentication frame. The Authentication frame shall contain an MDE, with contents as presented in Beacon and Probe Response frames. The FTE shall include the key holder identities of the AP, the R0KH-ID and R1KH-ID, set to the values of dot11FTR0KeyHolderID and dot11FTR1KeyHolderID, respectively. The FTE shall have a MIC element count of zero (i.e., no MIC present) and have ANonce, SNonce, and MIC fields set to 0.

The S1KH on STA provides the PMKR1Name in the PMKID field of the RSNE to be included in the (Re)Association Request Frame. The PMKR1Name shall be as calculated by the S1KH according to the procedures of 11.6.1.7.4; all other fields of the RSNE shall be identical to the RSNE present in the Authentication frame. The S1KH shall provide the FTE and MDE; the FTE and MDE shall be the same as those provided in the AP’s Authentication frame.

Finally, the R1KH provides the PMKR1Name in the PMKID field of the RSNE to be included in Association Response frame. The PMKR1Name shall be as calculated by the R1KH according to the procedures of 11.6.1.7.4 and shall be the same as the PMKR1Name in the Association Request frame; all other fields of the RSNE shall be identical to the RSNE present in the Beacon or Probe Response frames. The R1KH shall also provide the FTE, the MDE. The FTE and MDE shall be the same as in the Authentication frame.

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 11.11.2.3.2.

***Insert new figure as follows***



**Figure 12-ai1—FT initial mobility domain association using FILS authentication in an RSN**