IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Determine SAE PMK length | | | | |
| Date: 2022-03-01 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Po-Kai Huang | Intel |  |  |  |
| Ido Ouzieli |  |  |  |
| Ilan Peer |  |  |  |
| Johannes Berg |  |  |  |
|  |  |  |  |
|  |  |  |  |

Abstract

This document proposes a way to differentiate when to use 256 bit PMK and when to use 384 bit PMK for SAE.

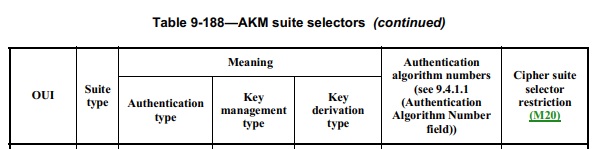
Revisions:

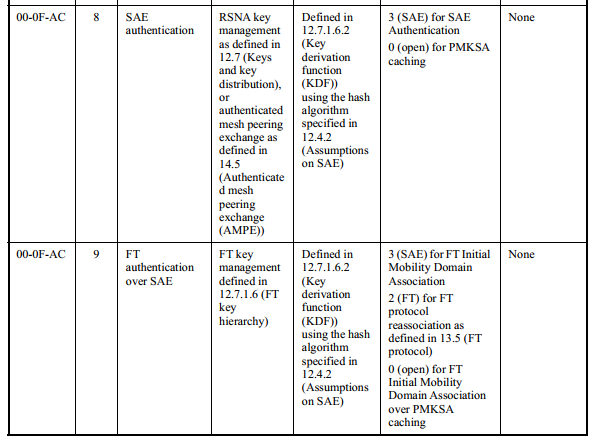
* Rev 0: Initial version of the document.
* Rev 1: Revision based on the comments from Mark Rison and Thomas
* Rev 2: Revision of discussion and texts based on further feedback

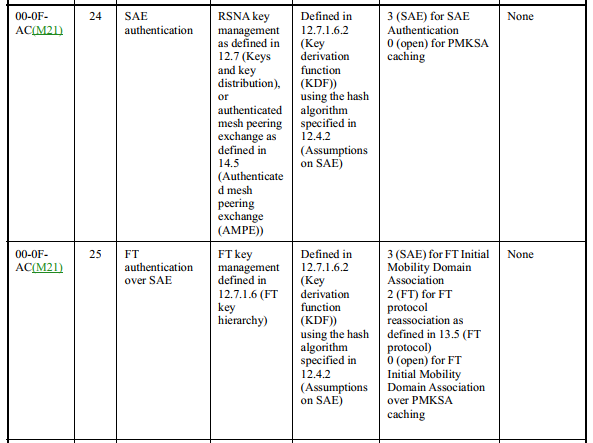
Discussion:

There is a potential issue on how to determine the PMK length for SAE. The reason for the issue is that the current spec ties the length of PMK to the AKM that is used as shown below, but during SAE exchange there is no way to know the AKM since AKM is not carried during SAE exchange because RSNE is not carried during the SAE exchange.

*If used with (M21)AKMs 00-0F-AC:8 or 00-0F-AC:9 and the looping method of PWE generation  
(see 12.4.4.2.2 (Generation of the password element with ECC groups by looping) and 12.4.4.3.2 (Generation of the password element with FFC groups by looping)), both the KCK and PMK shall be 256 bits in length. If used with AKMs 00-0F-AC:8 or 00-0F-AC:9 and the hash-to-element method(#344) of PWE generation (see 12.4.4.2.3 (Hash-to-element(#331) generation of the password element with ECC groups) and 12.4.4.3.3 (Direct generation of the password element with FFC groups)), the KCK shall be the length of the digest generated by H() and the PMK shall be 256 bits in length (M21)(see 12.7.1.3 (Pairwise key hierarchy)). When AKM 00-0F-AC:24 or 00-0F-AC:25 is negotiated, the hash-to-element method(#344) of PWE generation (see 12.4.4.2.3 (Hash-to-element(#331) generation of the password element with ECC groups) and 12.4.4.3.3 (Direct generation of the password element with FFC groups)) shall be used, and the KCK shall be the length of the digest generated by H() and the PMK shall be 384 bits in length (see 12.7.1.3 (Pairwise key hierarchy)).*



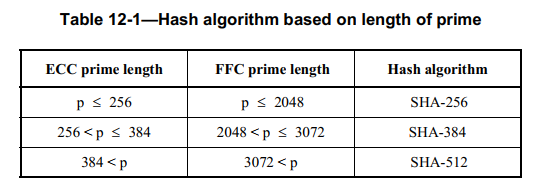




It maybe argued that we can use the Hash algorithm used by SAE to determine the PMK length, since SHA-256 with group 19, where an ECC group defined over a 256-bit prime order field, is mandated to be implemented as shown below and is commonly used for AKMs 00-0F-AC:8 or 00-0F-AC:9

*For the purpose of interoperability, a STA shall implement support for group 19, an ECC group defined over a 256-bit prime order field.*

However, it is also ture that the spec does not limit SHA-384 or SHA-512 to be used for AKMs 00-0F-AC:8 or 00-0F-AC:9 although there is no specific advantage to use longer SHA to increase the security strength. Therefore, using the hash algorithm still may not determine the AKM to be used without considering backward compatibility.



To resolve the problem, the easiest way seems to include AKM in the very first message of SAE so that there will be no ambiguity about the PMK length.



To make sure that we have legacy compatiability, we can also include AKM 00-0F-AC:24 or 00-0F-AC:25 in the first message of SAE only when AKM 00-0F-AC:24 or 00-0F-AC:25 is supported by the AP during discovery and intended to be used by the client.

If AKM is not included in the first message of the SAE, then we can follow the existing procedure of using AKMs 00-0F-AC:8 or 00-0F-AC:9 before AKMs 00-0F-AC:24 and 00-0F-AC:25 are introduced.

We limit the fields in the RSNE to include only fields up to the AKM and put non relvenat fields as reserved, since those fields will not be used.

Verification of AKM under PMKSA is already there in 12.6.10.3 Cached PMKSAs and RSNA key management. See below.

*Upon receipt of a (Re)Association Request frame with one or more PMKIDs following Open System authentication or (only in the case of a DMG AP) if IEEE 802.11 authentication was not performed, an AP checks whether its Authenticator has cached a PMKSA for the PMKIDs and whether the AKM in the cached PMKSA matches the AKM in the (Re)Association Request; and if so, it shall assert possession of that PMKSA by beginning the 4-way handshake after association has completed and shall include the PMKID in message 1 (see 12.7.6.2 (4-way handshake message 1)). If the Authenticator does not have a PMKSA for the PMKIDs in the (re)association request or the AKM does not match, its behavior depends on how the PMKSA was established. If SAE authentication was used to establish the PMKSA, then the AP shall reject (re)association by sending a (Re)Association Response frame with status code STATUS\_INVALID\_PMKID.*

We do not find the texts in the current spec about AKM verification without PMKID. Verification of AKM without PMKID goes beyond the scope of the bug fix for this document.

Following this interpretation, we also note that it is possible to use SHA-256 for AKMs 00-0F-AC:24 or 00-0F-AC:25 since the current spec does not prevent that. However, using SHA-256 will directly limit the key strength of PMK, which goes against the purpose of increasing the key length of PMK, so we also propose to prevent that for SAE.

Texts are provided below based on the above discussion.

**Propose:**

**12.4.5.3 Construction of an SAE Commit message**

The scalar and element in an SAE Commit message shall be produced using ***PWE*** and secrets generated in 12.4.5.2 (PWE and secret generation), as follows:

*commit*-*scalar* = (*rand* + *mask*) mod *r****COMMIT-ELEMENT*** = inverse-op(scalar-op(*mask*, ***PWE***))

This message shall be transmitted to the peer as described in 12.4.7 (Framing of SAE). The temporary secret *mask* may be deleted at this point.

To derive keys for use with AKM 00-0F-AC:24 or AKM 00-0F-AC:25, an RSNE shall be included in an SAE Commit message transmitted to the peer if the peer is discovered to support AKM 00-0F-AC:24 or AKM 00-0F-AC:25 and the state of the SAE finite state machine is *Nothing* (see 12.4.8.2.2). In the RSNE, AKM 00-0F-AC:24 or AKM 00-0F-AC:25 is indicated, only fields up to and including the AKM Suite List field are included, the Group Data Cipher Suite field is reserved, and the Pairwise Cipher Suite Count field is set to 0 (see Figure 9-348).

* Processing of a peer’s SAE Commit message

*TGme editor: Modify the following pagraph as follows (track change on):*

The entropy of *k* shall then be extracted using H to produce *keyseed*. The key derivation function from 12.7.1.6.2 (Key derivation function (KDF)) shall then be used with the hash algorithm identified for H() (see 12.4.2 (Assumptions on SAE)) to derive a key confirmation key, KCK, and a pairwise master key, PMK, from *keyseed*.

The intended AKM for the purpose of PMK and KCK size determination (see below) is determined as follows:

* If an RSNE is not included in the SAE Commit message from the peer and the state of the SAE finite state machine is *Nothing* (see 12.4.8.2.2 Protocol instance states), then AKMs 00-0F-AC:8 or 00-0F-AC:9 shall be the intended AKM.
* If the state of the SAE finite state machine is *Committed* (see 12.4.8.2.2 Protocol instance states) and the SAE Commit message that has been sent by the SAE finite state machine to transition into *Committed* state does not include an RSNE, then AKMs 00-0F-AC:8 or 00-0F-AC:9 shall be the intended AKM.
* If an RSNE that indicates AKM 00-0F-AC:24 or AKM 00-0F-AC:25 is included in the SAE Commit message from the peer and the state of the SAE finite state machine is *Nothing* (see 12.4.8.2.2 Protocol instance states), then the indicated AKM shall be the intended AKM
* If the state of the SAE finite state machine is *Committed* (see 12.4.8.2.2 Protocol instance states) and the SAE Commit message that has been sent by the SAE finite state machine to transition into *Committed* state includes an RSNE that indicates AKM 00-0F-AC:24 or AKM 00-0F-AC:25, then the indicated AKM shall be the intended AKM

If the intended AKM is (M21)AKMs 00-0F-AC:8 or 00-0F-AC:9 and the looping method of PWE generation (see 12.4.4.2.2 (Generation of the password element with ECC groups by looping) and 12.4.4.3.2 (Generation of the password element with FFC groups by looping)), both the KCK and PMK shall be 256 bits in length. If the intended AKM is AKMs 00-0F-AC:8 or 00-0F-AC:9 and the hash-to-element method(#344) of PWE generation (see 12.4.4.2.3 (Hash-to-element(#331) generation of the password element with ECC groups) and 12.4.4.3.3 (Direct generation of the password element with FFC groups)), the KCK shall be the length of the digest generated by H() and the PMK shall be 256 bits in length (M21)(see 12.7.1.3 (Pairwise key hierarchy)). If the intended AKM is AKM 00-0F-AC:24 or 00-0F-AC:25, the hash-to-element method(#344) of PWE generation (see 12.4.4.2.3 (Hash-to-element(#331) generation of the password element with ECC groups) and 12.4.4.3.3 (Direct generation of the password element with FFC groups)) shall be used, the hash algorithm shall not be SHA-256 (see Table 12-1 (Hash algorithm based on length of prime)), and the KCK shall be the length of the digest generated by H() and the PMK shall be 384 bits in length (see 12.7.1.3 (Pairwise key hierarchy)). Use of other AKMs with the hash-to-element method(#344) will require definition of the length of the PMK. If both SAE Commit messages indicated a status code of SAE\_HASH\_TO\_ELEMENT, a salt consisting of the  
concatenation of the rejected groups from each peer’s Rejected Groups element shall be passed to the KDF; those of the peer with the highest MAC address go first (if only one sent a Rejected Groups element then the salt will consist of that list). If neither peer sent a Rejected Groups element or the status code was not SAE\_HASH\_TO\_ELEMENT, the salt shall consist of a series of octets of the value zero whose length equals the length of the digest of the hash function used to instantiate H().

*TGme editor: Modify 9.3.3.11 Authentication frame format as follows (track change on):*

* **Authentication frame format**

The frame body of an Authentication frame contains the information shown in Table 9-68 (Authentication frame body). FT authentication is used when FT support is advertised by the AP and dot11FastBSSTransitionActivated is true in the STA. SAE authentication is used when dot11MeshActiveAuthenticationProtocol is sae (1). FILS authentication is used if support for FILS authentication is advertised by the AP and dot11FILSActivated is true in the STA.

|  |  |  |
| --- | --- | --- |
| * **Authentication frame body** | | |
| **Order** | **Information** | **Notes** |
| 1 | Authentication algorithm number |  |
| 2 | Authentication transaction sequence number |  |
| 3 | Status code | The status code information is reserved in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 4 | Finite Cyclic Group | An unsigned integer indicating a finite cyclic group as described in 9.4.1.42 (Finite Cyclic Group field). This is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 5 | Anti-Clogging Token | A random bit string used for anti-clogging purposes as described in 12.4.6 (Anti-clogging tokens). This is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 6 | Send-Confirm | A binary encoding of an integer used for anti-replay purposes as described in 12.4.7.5 (Encoding and decoding of SAE Confirm messages). This is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 7 | Scalar | An unsigned integer encoded as described in 12.4.7.4 (Encoding and decoding of SAE Commit messages). This is present only in cer-tain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 8 | FFE(#312) | An element in a finite field encoded as described in 12.4.7.4 (Encoding and decoding of SAE Commit messages). This is present only in cer-tain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 9 | Confirm | An unsigned integer encoded as described in 12.4.7.5 (Encoding and decoding of SAE Confirm messages). This is present only in cer-tain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 10 | Challenge text | A Challenge Text element is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 11 | RSN | An RSNE is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 12 | Mobility Domain | An MDE is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 13 | Fast BSS Transition | An FTE is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 14 | Timeout Interval (reassociation deadline) | A TIE containing the reassociation deadline interval is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 15 | RIC | A resource information container, containing a variable number of elements, is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 16 | Multi-band | The Multi-band element is optionally present if dot11MultibandImplemented is true. |
| 17 | Neighbor Report | One or more Neighbor Report elements are present only in cer-tain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 18 | FILS Nonce | The FILS Nonce element is present in FILS Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 19 | FILS Session | The FILS Session element is present in FILS Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 20 | FILS Wrapped Data | The FILS Wrapped Data element is present in FILS Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 21 | Association Delay Info | The Association Delay Info element is present in FILS Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 22 | Password Identifier | The Password Identifier element is optionally present in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 23 | Rejected Groups | The Rejected Groups element is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| 24 | Anti-Clogging Token Container | The Anti-Clogging Token Container element is present only in certain Authentication frames as defined in Table 9-69 (Presence of fields and elements in Authentication frames). |
| Last | Vendor Specific | One or more Vendor Specific elements are optionally present. These elements follow all other elements. |

|  |  |  |  |
| --- | --- | --- | --- |
| * **Presence of fields and elements in Authentication frames** | | | |
| **Authentication algorithm** | **Authentication transaction sequence number** | **Status code** | **Presence of fields and elements  from order 4 onward** |
| Open System | 1 | Reserved | Not present |
| Open System | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | Not present |
| Open System | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) is present |
| Shared Key | 1 | Reserved | Not present |
| Shared Key | 2 | Any | The Challenge Text element is present |
| Shared Key | 3 | Reserved | The Challenge Text element is present |
| Shared Key | 4 | Any | Not present |
| FT | 1 | Reserved | The Mobility Domain element is present.  The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true. |
| FT | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | The Mobility Domain element is present if the Status Code field is 0.  The Fast BSS Transition element and RSNEs are present if the Status Code field is 0 and dot11RSNAActivated is true. |
| FT | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) is present |
| FT | 3 | Reserved | The Mobility Domain element is present.  The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true.  The RIC element is optionally present. |
| FT | 4 | Any | The Mobility Domain element is present if the Status Code field is 0.  The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true.  The RIC element is optionally present if the Status Code field is 0.  The TIE (reassociation deadline) is present if a RIC element is present. |
| SAE | 1 | Any | The Scalar field is present if the Status Code field is zero or 126.  The FFE field is present if the Status Code field is zero or 126.  When the hunting-and-pecking method is used to drive the PWE, the Anti-Clogging Token field is present if the Status Code field is ANTI\_CLOGGING\_TOKEN\_REQUIRED or if the Authentication frame is in response to a previous rejection with the Status Code field equal to ANTI\_CLOGGING\_TOKEN\_REQUIRED.  The Finite Cyclic Group field is present if the Status Code field is zero, ANTI\_CLOGGING\_TOKEN\_REQUIRED, 77 or 126.  The Password Identifier element is optionally present if the Status Code field is zero, 123 or 126.  (#288)If the Status Code field is 126, the Rejected Groups element is conditionally present as described in 12.4.7.4 (Encoding and decoding of SAE Commit messages); otherwise the Rejected Groups element is not present. When the hash-to-element method is used to derive the PWE, the Anti-Clogging Token Container element is present if the Status Code field is ANTI\_CLOGGING\_TOKEN\_REQUIRED or if the Authentication frame is in response to a previous rejection with the Status Code field equal to ANTI\_CLOGGING\_TOKEN\_REQUIRED.  The RSNE is present if dot11RSNAActivated is true, AKM 00-0F-AC:24 or AKM 00-0F-AC:25 is indicated as the intended AKM (see 12.4.5.4 Processing of a peer’s SAE Commit message), the state of the SAE finite state machine is *Nothing* (see 12.4.8.2.2 Protocol instance states), and the peer is discovered to support AKM 00-0F-AC:24 or AKM 00-0F-AC:25; otherwise, it is not present. |
| SAE | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | The Send-Confirm field is present.  The Confirm field is present. |
| SAE | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) are present |
| FILS Shared Key authentication without PFS | 1 | Reserved | The RSNE is present.  The MDE is present if the FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present.  The FILS Session element is present.  The FILS Wrapped Data element is present. |
| FILS Shared Key authentication without PFS | 2 | Status | The RSNE is present.  The MDE and the FTE are present if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present if the Status Code field is 0.  The FILS Session element is present if the Status Code field is 0.  The FILS Wrapped Data element is present if the Status Code field is 0.  The Association Delay Info element is present if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame. |
| FILS Shared Key authentication with PFS | 1 | Reserved | The Finite Cyclic Group field is present.  The FFE field is present.  The RSNE is present.  The MDE is present if the FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present.  The FILS Session element is present.  The FILS Wrapped Data element is present. |
| FILS Shared Key authentication with PFS | 2 | Status | The Finite Cyclic Group field(#313) is present if the Status Code field is 0.  The FFE field is present if the Status Code field is 0.  The RSNE is present.  The MDE and the FTE are present if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present if the Status Code field is 0.  The FILS Session element is present if the Status Code field is 0.  The FILS Wrapped Data element is present if the Status Code field is 0.  The Association Delay Info element is present if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame. |
| FILS Public Key authentication | 1 | Reserved | The Finite Cyclic Group field is present.  The FFE field is present.  The RSNE is present.  The MDE is present if the FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present.  The FILS Session element is present. |
| FILS Public Key authentication | 2 | Status | The Finite Cyclic Group field(#313) is present if the Status Code field is 0.  The FFE field is present if the Status Code field is 0.  The RSNE is present.  The MDE and the FTE are present if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association.  The FILS Nonce element is present if the Status Code field is 0.  The FILS Session element is present if the Status Code field is 0.  The Association Delay Info element is present if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request  frame. |