IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TGai comments assigned to me | | | | |
| Date: 2016-01-19 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Jouni Malinen | Qualcomm |  |  | jouni@qca.qualcomm.com |
|  |  |  |  |  |

Abstract

This document proposes resolutions to the remaining comments assigned to me from the initial TGai SB: CID 10104, CID 10108, CID 10217.

# CID 10104

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clause Number | Page | Line | Comment | Proposed Change | Proposed Resolution |
| 8.4.2.178 | 71 | 55 | While it is fine to leave the exact construction of Cache Identifier outside the scope of this standard, it would be good to describe how this value gets configured on an AP. A new MIB variable would sound like the approach that would best match similar use cases in the current base standard. | Add dot11CacheIdentifier MIB variable into Annex C. | REVISED. Apply changes proposed for CID 10104 in <this document>. |

## Proposed changes to address CID 10104

**8.4.2.178 FILS Indication element**

*Change the following paragraph in D6.3 page 65 lines 43-50 as shown:*

The Cache Identifier Included bit is set in the FILS Information field when PMKSA caching is supported. When the Cache Identifier Included bit is 1, a 2-octet Cache Identifier field is present in the FILS Indication element. When the Cache Identifier Included bit is 0 the Cache Identifier field is not present in the FILS Indication element. The content of the Cache Identifier field is an opaque octet string that identifies the scope that PMKSAs are cached. The assignment of the cache identifier is outside the scope of the standard but its value must be unique per authenticator within an ESS. On the AP, dot11CacheIdentifier contains the value of the Cache Identifier.

**C.3 MIB Detail**

*Change C.3 in D6.3 page 164 line 20 as shown:*

dot11FILSComplianceGroup OBJECT-GROUP

OBJECTS {

dot11FILSActivated,

dot11FILSFDFrameBeaconMinimumInterval,

dot11FILSBeaconResponseWindow,

dot11FILSOmitReplicateProbeResponses,

dot11DILSImplemented,

dot11FILSProbeDelay,

dot11HLPWaitTime,

dot11CacheIdentifier

}

*Insert the following MIB entry into C.3 in D6.3 page 166 line 63 (immediately following dot11HLPWaitTime):*

dot11CacheIdentifier OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(2))

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable. It is written by an external management entity. Changes take effect as soon as practical in the implementation. This value specifies the Cache Identifier that the FILS AP advertises in FILS Indication elements."

::= { dot11FILSConfigEntry 6 }

# CID 10108

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clause Number | Page | Line | Comment | Proposed Change | Proposed Resolution |
| 8.4.2.178 | 71 | 1 | The FILS Indication element does not seem to cover number of currently used mechanism to identify a suitable AP for various Interworking use cases. The Domain Identifier field addresses some of these (NAI Realm list and 3GPP), but others like HESSID and Roaming Consortium OI cannot be used. For Interworking network selection to work efficiently, it would be useful to provide possibility of including such information in the FILS Discovery frame. | Extend FILS Indication element format to allow optional inclusion of HESSID and up to three Roaming Consortium OIs. | REVISED. Apply changes proposed for CID 10108 in <this document>. |

## Proposed changes to address CID 10108

**8.4.2.178 FILS Indication element**

*Change Figure 8-577k (FILS Indication element format) in D6.3 page 64 lines 51-58 as shown (add two new columns):*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Element ID | Length | FILS  Information | Cache Identifier | Realm Iden- tifier | Public Key Iden- tifier | HESSID | Roaming  Consortium  OIs |

Octets: 1 1 2 0 or 2 variable variable 0 or 6 variable

##### **Figure 8-577k—FILS Indication element format**

*Change Figure 8-577l (FILS Information field definition) in D6.3 page 65 lines 8-14 as shown (use three of the previous reserved bits):*

B0 B2 B3 B5 B6 B7 B8 B9 B10 B11 B15

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Number of Public Key Identifiers | Number of Realm Identi- fiers | FILS IP  Address Configura- tion | Cache Iden- tifier Included | HESSID  Included | Number of  Roaming  Consortium  OIs | Reserved |

Bits: 3 3 1 1 1 2 5

##### **Figure 8-577l—FILS Information field definition**

*Insert new paragraphs into D6.3 at page 65 line 52:*

When the HESSID Included bit is 1, a 6-octet HESSID field is present in the FILS Indication element. When the HESSID Included bit is 0, the HESSID field is not present in the FILS Indication element. The HESSID field is set to the value of dot11HESSID.

Number of Roaming Consortium OIs field indicates how many Roaming Consortium OIs are included in the FILS Indication element. Each Roaming Consortium OI is formatted per Figure 9-598 (OI Duple subfield format). The values of the OIs in this field are equal to the values of the up to 3 first OIs in the dot11RoamingConsortiumTable.

# CID 10217

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Clause Number | Page | Line | Comment | Proposed Change | Proposed Resolution |
| 11.6.4 | 139 |  | P802.11ai changed rules on how the Key MIC field in the EAPOL-Key frames is set, but there are no changes to 11.6.4 and 11.6.6 to update the EAPOL-Key(S, M, ...) uses where that M is the Key MIC value. Those places are claiming that M=1 is used in number of cases that conflict with the rules described in P802.11ai for the AEAD cipher case. | Update 11.6.4 and 11.6.6 useds of EAPOL-Key(S, M, ..) to cover AEAD cipher. | REVISED. Apply changes proposed for CID 10217 in <this document>. |

**Discussion**

11.6.2 text describing the changed rule: “Key MIC (bit 8). When AKM negotiated is not 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, 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.“

11.6.11.3 text describing the changed rule: “*MICVerified* – This variable is set to true if the MIC on the received EAPOL-Key frame is verified and is correct or if AEAD cipher is used and AEAD decryption steps succeed. Any EAPOL-Key frames with an invalid MIC are dropped and ignored.“

P802.11REVmc/D5.0 12.7.4: “EAPOL-Key(S, M, A, I, K, SM, KeyRSC, ANonce/SNonce, MIC, DataKDs)” and “where … M means the MIC is available in message. This should be set in all messages except Message 1 of a 4-way handshake. This is the Key MIC bit of the Key Information field.“

## Proposed changes to address CID 10217

**12.7.4 EAPOL-Key frame notation**

*Change the following definition in REVmc/D5.0 page 2005 lines 60-62 as shown:*

where

…

M means the MIC is available in message. This should be set in all messages except Message 1 of a 4-way handshake. This is the Key MIC bit of the Key Information field. When the negotiated AKM is 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, this Key MIC bit is set to 0 regardless of the M parameter value.

*Change the following definition in REVmc/D5.0 page 2006 line 48 as shown:*

MIC is the integrity check, which is generated using the KCK. This is the Key MIC field. When the negotiated AKM is 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, the Key MIC field is not included regardless of the MIC parameter value.

**12.7.6.3 4-way handshake message 2**

*Change the value of Key MIC in REVmc/D5.0 page 2009 line 55 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2010 line 9 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2010 line 43 as shown:*

On reception of Message 2, the Authenticator checks that the key replay counter corresponds to the outstanding Message 1. If not, it silently discards the message. Otherwise, the Authenticator:

1. Derives PTK.
2. Verifies the Message 2 MIC or AEAD decryption operation result.
3. If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or AEAD decryption operation returns failure, the Authenticator silently discards Message 2.
4. If the MIC or AEAD decryption is valid and Message 2 is part of a fast BSS transition Initial Mobility Domain Association, see 13.4.2 (FT initial mobility domain association in an RSN). If the MIC or AEAD decryption is valid and Message 2 is not part of a fast BSS transition Initial Mobility Domain Association, the Authenticator checks that the RSNE bitwise matches that from the (Re)Association Request frame.
5. If these are not exactly the same, the Authenticator uses MLME-DEAUTHENTICATE.request primitive to terminate the association.
6. If they do match bitwise, the Authenticator constructs Message 3.

**12.7.6.4 4-way handshake message 3**

*Change the value of Key MIC in REVmc/D5.0 page 2011 line 37 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2011 line 55 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL) or MIC(SKCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2013 line 4 as shown:*

On reception of Message 3, the Supplicant silently discards the message if the Key Replay Counter field value has already been used or if the ANonce value in Message 3 differs from the ANonce value in Message 1. The Supplicant also:

a) Verifies the RSNE. If it is part of a fast BSS transition Initial Mobility Domain Association, see 13.4.2 (FT initial mobility domain association in an RSN). Otherwise, if it is not identical to that the STA received in the Beacon or Probe Response frame, the STA shall disassociate or deauthenticate. If a second RSNE is provided in the message, the Supplicant uses the pairwise cipher suite specified in the second RSNE or deauthenticates.

a) Verifies the Message 3 MIC or AEAD decryption operation result. If the calculated MIC does not match the MIC that the Authenticator included in the EAPOL-Key frame or AEAD decryption operation returns failure, the Supplicant silently discards Message 3.

**12.7.6.5 4-way handshake message 4**

*Change the value of Key MIC in REVmc/D5.0 page 2014 line 14 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2014 line 31 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL) or MIC(SKCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2014 line 47 as shown:*

On reception of Message 4, the Authenticator verifies that the Key Replay Counter field value is one that it used on this 4-way handshake; if it is not, it silently discards the message. Otherwise:

1. The Authenticator checks the MIC or AEAD decryption operation result. If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or AEAD decryption operation returns failure, the Authenticator silently discards Message 4.

* b)  If the MIC is valid, the Authenticator uses the MLME-SETKEYS.request primitive to configure the IEEE Std 802.11 MAC to send and, if the receive key has not yet been installed, to receive protected, individually addressed MPDUs using for the new PTK.

**12.7.7.2 Group key handshake message 1**

*Change the value of Key MIC in REVmc/D5.0 page 2019 line 18 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2019 line 34 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2019 line 50 as shown:*

On reception of Message 1, the Supplicant:

a) Verifies that the Key Replay Counter field value has not yet been seen before, i.e., its value is strictly larger than that in any other EAPOL-Key frame received thus far during this session.

b) Verifies that the MIC is valid, i.e., it uses the KCK that is part of the PTK to verify that there is no data integrity error, or that the AEAD decryption steps succeed.

**12.7.7.3 Group key handshake message 2**

*Change the value of Key MIC in REVmc/D5.0 page 2020 line 16 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2020 line 33 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC(KCK, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2019 line 43 as shown:*

On reception of Message 2, the Authenticator:

a) Verifies that the Key Replay Counter field value matches one it has used in the group key handshake.

b) Verifies that the MIC is valid, i.e., it uses the KCK that is part of the PTK to verify that there is no data integrity error, or that the AEAD decryption steps succeed.

**12.7.8.2.2 SMK Handshake Message 1**

*Change the value of Key MIC in REVmc/D5.0 page 2023 line 16 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2023 line 32 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (initiating STA’s KCK, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2023 line 43 as shown:*

On receipt of Message 1, the AP checks that the key replay counter corresponds to Message 1. If not, it silently discards the message. Otherwise:

a) The AP verifies the Message 1 MIC using the STA\_I PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the STA\_I included in the EAPOL-Key frame or AEAD decryption operation returns failure, the AP silently discards Message 1.

b) If the MIC is correct or the AEAD decryption steps succeed, the AP checks if the STA\_P is reachable. If it is not reachable, the AP shall send an error EAPOL-Key frame to STA\_I per 12.7.8.5.2 (Error ERR\_STA\_NR). After sending the message, AP silently discards Message 1.

**12.7.8.2.3 SMK Handshake Message 2**

*Change the value of Key MIC in REVmc/D5.0 page 2024 line 11 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2024 line 28 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of the STA\_P, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2024 line 37 as shown:*

The AP sends Message 2 to the STA\_P. On receipt of Message 2, the STA\_P checks that the key replay counter corresponds to Message 2. If not, it silently discards the message. Otherwise,

* a) The STA\_P verifies the Message 2 MIC using the STA\_P PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key frame or AEAD decryption operation returns failure, the STA\_P silently discards Message 2.
* b)  If the MIC is correct or the AEAD decryption steps succeed, the STA\_P checks if it supports at least one cipher suites proposed by the STA\_I. If it does not, the STA\_P shall send an error EAPOL-Key frame to STA\_I through the AP per 12.7.8.5.4 (Error ERR\_CPHR\_NS). After sending the error message, the STA\_P silently discards Message 2.

**12.7.8.2.4 SMK Handshake Message 3**

*Change the value of Key MIC in REVmc/D5.0 page 2025 line 11 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2025 line 28 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of the STA\_I, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2025 line 37 as shown:*

The STA\_P sends Message 3 to the AP. On receipt of Message 3, the AP checks that the key replay counter corresponds to Message 3. If not, it silently discards the message. Otherwise,

* a) The AP verifies the Message 3 MIC using the STA\_I PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the STA\_P included in the EAPOL-Key frame or AEAD decryption operation returns failure, the AP silently discards Message 3.
* b)  If MIC is correct or the AEAD decryption steps succeed, the AP checks if the STA\_I is reachable. If it is not reachable, the AP shall send an error EAPOL-Key frame to the STA\_P per 12.7.8.5.2 (Error ERR\_STA\_NR). After sending the message, the AP silently discards Message 3.

**12.7.8.2.5 SMK Handshake Message 4**

*Change the value of Key MIC in REVmc/D5.0 page 2026 line 6 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2026 line 22 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of the STA\_P, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2026 line 32 as shown:*

The AP sends Message 4 to the STA\_P. On receipt of Message 4, the STA\_P checks that the key replay counter corresponds to Message 4. If not, it silently discards the message. Otherwise,

* a) The STA\_P verifies the Message 4 MIC using the STA\_P PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key frame or AEAD decryption operation returns failure, the STA\_P silently discards Message 4.

b)  If the MIC is correct or the AEAD decryption steps succeed, the STA\_P identifies the PeerKey session using the PNonce sent as part of the Key Nonce field of Message 4. If STA\_P has an existing PeerKey state for this session, i.e., STA\_P has received Message 2 and this message is a follow-up to that. If STA\_P has an existing PeerKey state for this session, STA\_P silently discards Message 4.

**12.7.8.2.6 SMK Handshake Message 5**

*Change the value of Key MIC in REVmc/D5.0 page 2027 line 3 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2027 line 19 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC (KCK of the STA\_I, EAPOL)

*Change the EAPOL-Key reception process in REVmc/D5.0 page 2026 line 32 as shown:*

The AP sends Message 5 to the STA\_I. On receipt of Message 5, the STA\_I checks that the key replay counter corresponds to Message 5. If not, it silently discards the message. Otherwise,

* a) The STA\_I verifies the Message 5 MIC using the STA\_I PTKSA if an AEAD cipher is not used. If the calculated MIC does not match the MIC that the AP included in the EAPOL-Key frame or AEAD decryption operation returns failure, the STA\_I silently discards Message 5.
* b)  If the MIC is correct or the AEAD decryption steps succeed, the STA\_I identifies the PeerKey session using the INonce sent as part of the Key Nonce field of Message 5. If STA\_I has an existing PeerKey state for this session, i.e., STA\_I has initiated this message exchange using Message 1 and this message is a follow-up to that. If STA\_I has an existing PeerKey state for this session, STA\_I shall silently discard Message 5.

**12.7.8.5 Error Reporting**

**12.7.8.5.1 General**

*Change the value of Key MIC in REVmc/D5.0 page 2029 line 49 as shown:*

Key MIC = 0 when using an AEAD cipher or 1 otherwise

*Change the value of Key MIC in REVmc/D5.0 page 2030 line 1 as shown:*

Key MIC = Not present when using an AEAD cipher; otherwise, MIC computed over the body of this EAPOL-Key frame

**12.7.9.7 Supplicant PeerKey state machine variables**

*Change the MICVerified definition in REVmc/D5.0 page 2040 line 40 as shown:*

*MICVerified* **–** This variable is set to true if the MIC on the received EAPOL-Key frame is verified and is correct or if an AEAD cipher is used and the AEAD decryption steps succeed. Any EAPOL-Key frames with an invalid MIC are dropped and ignored.

**12.7.10.3 Supplicant state machine variables**

*Change the MICVerified definition in REVmc/D5.0 page 2042 line 22 as shown:*

*MICVerified* **–** The Supplicant sets this variable to true if the MIC on the received EAPOL-Key frame verifies as correct or if an AEAD cipher is used and the AEAD decryption steps succeed. The Supplicant silently discards any EAPOL-Key frame received with an invalid MIC.

**12.7.11.4 Authenticator state machine procedures**

*Change the MIC(x) definition in REVmc/D5.0 page 2049 line 23 as shown:*

**MIC**(x) – Computes a MIC over the plaintext data. When an AEAD cipher is used, returns an empty string.