IEEE P802.11
Wireless LANs

|  |
| --- |
| Proposed resolutions to some LB258 comments |
| Date: 2022-05-10 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Jouni Malinen | Qualcomm |  |  | jouni@qca.qualcomm.com |
|  |  |  |  |  |

Abstract

This document proposes resolutions to the follow LB258 comments:

CID 1082 (MAC): Secure bit in EAPOL-Key frames

CID 1277 (SEC): H2E to SAE finite state machine (Visio figure)

CID 1278 (ED2): Consistent style for status code in SAE finite state machine (Visio figure)

CID 1699 (MAC): State changes on reassociation-to-same-BSS failure

CID 1813 (MAC): Applicability of management frame protection

# Comments

## CID 1082

12.7.2 P3206 L7

Comment:

make it unambiguous which PTK we're talking about

Proposed Change:

insert "initial" before PTK

Discussion:

SEC: 2022-04-11 15:48:11Z - status set to: Submission Required

The cited text with the proposed change in redline is:

"The Supplicant shall set the Secure bit to 0 in all EAPOL-Key frames it sends before it has the initial PTK and the GTK and before it has received an EAPOL-Key frame from the Authenticator with the Secure bit equal to 1 (this should be before receiving message 3 of the 4-way handshake). The Supplicant shall set the Secure bit to 1 in all EAPOL-Key frames sent after this until it loses the security association it shares with the Authenticator."

Previously reviewed in CID 179 and CID 180 - 11-21/829

Minutes:

* + 1. CID 1082 (SEC)
			1. Review comment
			2. Discussion on when to set the secure bit.
			3. Concern on remembering what we agreed in the past, and this change will make it clear, but we need to make sure it is what we wanted to have happen. Concern with legacy devices.
			4. Need to make sure we are consistent with changes for CID 179 and 180 in doc 11-21/829r10
				1. Resolution for CID 179 and 180:

REJECTED (SEC: 2022-01-07 17:14:58Z) - The group could not come to consensus on a set of changes to the draft that would satisfy the commenter. The group discussed changes to address the comment in <https://mentor.ieee.org/802.11/dcn/21/11-21-0829-10-000m-resolutions-for-some-comments-on-11me-d0-0-cc35.docx> and ran the following straw poll:

Do you agree to resolve CIDs 179/180 with the text changes provided in document 11-21/829r10 limited to the handshake analyis clause (12.7.6.8)?

A.Yes 1

B.No 4

C.Abs 4

* + - 1. While we did not accept the document, we can review the discussion in rejecting the document.
			2. More work offline may need to be done.
			3. Assign to Jouni MALINEN and mark submission required.

CIDs 179 and 180 pointed out a confusing description of how EAPOL-Key msg 3/4 differs from msg 2/4 and 4/4. The proposed changes to address these comments in 829r10 went through significantly larger scope of issues related to the way the 4-way handshake and the group key handshake are described.

The context for the Secure bit and the proposed change in the comment is shown below:

REVme/D1.2 P3210 L25-36:

**12.7.2 EAPOL-Key frames**

...

7) Secure (bit 9) is set to 1 once the initial key exchange is complete.

The Authenticator shall set the Secure bit to 0 in all EAPOL-Key frames sent before the Supplicant has the PTK and the GTK. The Authenticator shall set the Secure bit to 1 in all EAPOL-Key frames it sends to the Supplicant containing the last key needed to complete the Supplicant’s initialization.

The Supplicant shall set the Secure bit to 0 in all EAPOL-Key frames it sends before it has the initial PTK and the GTK and before it has received an EAPOL-Key frame from the Authenticator with the Secure bit equal to 1 (this should be before receiving message 3 of the 4-way handshake). The Supplicant shall set the Secure bit to 1 in all EAPOL-Key frames sent after this until it loses the security association it shares with the Authenticator.

...

The current draft text here (and the earlier forms of it starting from IEEE Std 802.11i-2004) are somewhat unclear. The term "initial key exchange" is not defined. This might be referring to the first 4-way handshake in an association (which was the only option for deriving the PTK in IEEE Std 802.11i-2004) and potentially also to the use of FT protocol and FILS authentication where the PTK is derived during the authentication and association exchange, i.e., without using the 4-way handshake. It should also be noted that this tracking of state would be cleared when the association is lost.

"containing the last key needed to complete" is somewhat confusing. It might be referring to the EAPOL-Key msg 3/4 at the beginning of an association since that frame contains the GTK. However, it is not clear how this description of the Authenticator behavior would apply to any EAPOL-Key frame transmitted after the initial 4-way handshake (and even the first 4-way handshake in an association stated through FT protocol or FILS authentication).

"has the PTK" is unclear (like the comment pointed out) since EAPOL-Key frames are used to generate a new PTKSA during rekeying and that new PTKSA replaces the old PTKSA at the time it is created (since there can be only a single PTKSA between the Authenticator and the Supplicant). This "the PTK" could be a reference the "initial PTK" like the comment is proposing, i.e., the first PTK derived for the association. Or it could be a reference to the PTK that is being derived in this instance of the 4-way handshake (i.e., the new PTKSA in case of rekeying).

"losing the security association" is unclear. The concept of "losing" an SA feels a bit strange. Furthermore, we have multiple different SAs. This might be refering to the PTKSA being deleted and if so, it has the same ambiguity with "the PTK" on which PTKSA is being deleted. Rekeying does result in a PTKSA (the old one) getting deleted. That happens at the same point when the new PTKSA is added. The text here might be interpreted to have to start setting the Secure bit to 0 in such case at the end of the rekeying 4-way handshake, but that would result in strange behavior since the new PTKSA is already available.

The following more detailed description of the 4-way handshake in 12.7.6 is significantly clearer on how the Secure bit is set, by indicating that it is 0 in msg 1/4 and 2/4 and 1 in msg 3/4 and 4/4 regardless of whether those messages are for the initial or rekeying cases.

**12.7.4 EAPOL-Key frame notation**

The following notation is used throughout the remainder of 12.7 (Keys and key distribution) and 13.4 (FT

initial mobility domain association) to represent EAPOL-Key frames:

EAPOL-Key(S, M, A, I, K, Reserved, KeyRSC, ANonce/SNonce, MIC, {Key Data})

where

S means the initial key exchange is complete; this is the Secure bit of the Key Information field

…

**12.7.6 4-way handshake**

**12.7.6.1 General**

RSNA defines a protocol using EAPOL-Key frames called the 4-way handshake. The handshake completes the IEEE 802.1X authentication process. The information flow of the 4-way handshake is as follows:

Message 1: Authenticator 🡪 Supplicant: EAPOL-Key(0,0,1,0,P,0,0,ANonce,0,{} or {PMKID})

Message 2: Supplicant 🡪 Authenticator: EAPOL-Key(0,1,0,0,P,0,0,SNonce,MIC,{RSNE} or

{RSNE, OCI KDE} or {RSNE, RSNXE} or {RSNE, OCI KDE, RSNXE})

Message 3: Authenticator 🡪 Supplicant:

EAPOL-Key(1,1,1,1,P,0,KeyRSC,ANonce,MIC,{RSNE,GTK[N]} or

{RSNE, GTK[N], OCI KDE} or {RSNE, GTK[N], RSNXE} or

{RSNE, GTK[N], OCI KDE, RSNXE})

Message 4: Supplicant 🡪 Authenticator: EAPOL-Key(1,1,0,0,P,0,0,0,MIC,{}).

**12.7.6.2 4-way handshake message 1**

Message 1 uses the following values for each of the EAPOL-Key frame fields:

Descriptor Type = N – see 12.7.2 (EAPOL-Key frames)

Key Information:

Key Descriptor Version = 1 (ARC4 encryption with HMAC-MD5) or 2 (NIST AES key wrap with HMAC-SHA-1-128) or 3 (NIST AES key wrap with AES-128-CMAC), in all other cases 0

Key Type = 1 (Pairwise)

Reserved = 0

Install = 0

Key Ack = 1

Key MIC = 0

Secure = 0

…

In addition to the unclear description in the standard, it should be noted that there are deployed implementations that set the Secure bit value in identical way between the first 4-way handshake (the one that is used to derive the first PTKSA for the association) and the 4-way handshake that is used to rekey the PTK during an association. This matches the description in 12.7.6, but could be interpreted to differ from the one in 12.7.2. It would be problematic if we were to modify the standard at this point to make such implementations non-compliant and potentially imply that new implementations should discard the EAPOL-Key msg 2/4 from them during rekeying 4-way handshake. As such, if we decide to change something, it should be done in a manner that either allows this deployed functionality or at least notes that such behavior is used.

It is also not clear what the actual purpose and use of the Secure bit is. The standard does not seem to describe any particular behavior that would differ on the Authenticator or the Supplicant based on whether this bit is set to 0 or 1. It is not clear why it would be valuable for any 3rd party observer of the EAPOL-Key frames either.

All this seems to point towards us needing to determine do we want to clean this set of three paragraphs now, 18 years after they were introduced in IEEE Std 802.11i-2004. The text looks clearly ambiguous and potentially incorrect or at least misleading. That would seem to justify some work here. However, it is not clear that we can find an acceptable version of the text taken into account the potentially conflicting statements in this area in the standard and also taking into account the behavior in significant set of deployed devices.

If we were to want to clarify this behavior to the direction of the Secure bit indicating that there is a shared PTKSA in general, the changes could be something like following in this location. However, this would also require changing a number of locations within 12.7.6 to cover the different behavior between the initial and rekeying 4-way handshakes.

### CID 1082 changes - option A

*Modify 12.7.2 (REVme/D1.2 P3210 L25-36) as follows:*

**12.7.2 EAPOL-Key frames**

...

7) Secure (bit 9) indicates whether the Authenticator and the Supplicant share a PTKSA. It is set to 0 in messages 1 and 2 of the initial 4-way handshake when the Authenticator and the Supplicant do not share a PTKSA. Otherwise, it is set to 1.

NOTE—Some deployed Authenticator and Supplicant implementations set the Secure bit to 0 in the messages 1 and 2 of the 4-way handshake that is used for rekeying the PTK during an association even when they already share a previously generated PTKSA.

...

If we were to want to clarify this behavior to the direction of the Secure bit being used as it is in 12.7.6, i.e., without making any difference between initial and rekeying exchanges, the changes could be something like following in this location. This would likely not need changes within 12.7.6.

### CID 1082 changes - option B

*Modify 12.7.2 (REVme/D1.2 P3210 L25-36) as follows:*

**12.7.2 EAPOL-Key frames**

...

7) Secure (bit 9) is set to 0 in messages 1 and 2 of the 4-way handshake and to 1 in other messages.

...

## CID 1274 (PLACEHOLDER - NO NEED TO DISCUSS)

11.2.3.16.3 P2715 L5

Comment:

The design for delivering the GTK to a STA waking up from WNM sleep mode does not seem to work correctly for the case where RSN is used without management frame protection. This does not cover the case where there is a pending GTK update in progress (which is defined in the same paragraph for the management frame protection case) and as such, the STA could be left without knowing the new GTK when the AP/Authenticator takes that into use. The current text is as follows: "If RSN is used without management frame protection and a valid PTK is configured for the STA, the current GTK shall be sent to the STA using a group key handshake (see 12.7.7 (Group key handshake)) immediately following the WNM Sleep Mode Response frame."

Proposed Change:

Add after the cited sentence (i.e., at the end of the paragraph): "If a GTK update is in progress, the pending GTK shall be sent to the STA using another group key handshake immediately after the current GTK has been sent."

Resolution: (NOTE: Already discussed and database has a proposed resolution)

REVISED (MAC: 2022-04-27 18:14:06Z): Make the Proposed Change. Also, add a paragraph break after the second sentence in the cited paragraph.

Discussion:

This comment was apparently marked ready for motion, so no need to address in this document.

MAC: 2022-05-02 23:08:11Z - status set to: Ready for Motion

* + 1. CID 1274 (MAC)
			1. Review comment
			2. Discussion on when RSN is used.
			3. Looking at the paragraph, you have “If RSN is used with management” and another sentence of “If RSN is used without management” and some other side sentences, so we should put a paragraph break prior to the “If RSN is used without management” sentence.
			4. So, we add the new paragraph point and the requested sentence.
			5. Proposed Resolution: REVISED (MAC: 2022-04-27 18:14:06Z): Make the Proposed Change. Also, add a paragraph break after the second sentence in the cited paragraph.
			6. No Objection – Mark Ready for Motion

## CID 1277

**Owning Adhoc**: SEC

12.4.8 P3120

Comment:

12.4.8 (SAE finite state machine) subclauses were not updated to know about the new special status code value 126 for H2E. It is used similarly to the value 0 as a success case instead of other nonzero values indicating failures.

Proposed Change:

In Figure 12-4 (P3120 L7, 12.4.8.1):
Replace "1(0)" with "1(0 or 126)" in Nothing->Committed.
Replace "1(0)" with "1(0 or 126)" in Nothing->Confirmed.
Replace "1(0)" with "1(0 or 126)" in Confirmed->Confirmed.
Replace "1(0)" with "1(0 or 126)" in Committed->Confirmed.
Replace "1(0)" with "1(0 or 126)" in Committed->Committed (5 instances).

At P3121 L46 (12.4.8.3.2), replace "Com. Indicates receipt of an SAE Commit message (authentication transaction sequence number 1) with a status of 0." with "Com. Indicates receipt of an SAE Commit message (authentication transaction sequence number 1) with a status of 0 or 126."

At P3124 L30 (12.4.8.6.3), replace "Status code is not SUCCESS" with "Status code is not SUCCESS or SAE\_HASH\_TO\_ELEMENT".

At P3125 L18 (12.4.8.6.4), replace "If the Status is some other nonzero value" with "If the Status is some other nonzero value other than 126".

At P3125 L20 (12.4.8.6.4), replace "If the Status is zero" with "If the Status is 0 or 126".

Discussion:

SEC: 2022-01-12 20:02:28Z - status set to: Submission Required

**Proposed comment resolution for CID 1277**

Accepted.

Note to editors: Figure 12-4 is Figure 12-15 in REVme/D1.2 and the Visio source of the figure with the proposed changes in attached in <this document>.

## CID 1278

**Owning Adhoc**: ED2

12.4.8.1 P3120 L5

Comment:

The SAE Password Identifier mismatch case in Figure 12-4 (SAE finite state machine) uses inconsistent style in indicating the Authentication frame Status Code field value: "<123>" while all other cases use only the integer value in the 1(N) event.

Proposed Change:

Replace "1(<123>)" with "1(123)" in Nothing->Nothing transition of Figure 12-4.

Discussion

ED2: 2022-01-15 12:13:34Z - status set to: Submission Required

**Proposed comment resolution for CID 1278**

Accepted.

Note to editors: Figure 12-4 is Figure 12-15 in REVme/D1.2 and the Visio source of the figure with the proposed changes in attached in <this document>.

For discussion/reference:

Visio file with CID 1277 changes:



Visio file with CID 1278 changes:



Visio file with both CID 1277 and 1278 changes:



This is what the Figure 12-4 (REVme/D1.0), i.e., now Figure 12-15 (REVme/D1.2), would look like after applying the proposed changes from CID 1277 and 1278:



## CID 1699

**Owning Adhoc**: MAC

**Location**: 11.3.5

**Comment**

It is unfortunate that the current association is lost at the non-AP STA if reassociation to the same AP fails, and the SA is lost at the AP

**Proposed Change**

In 11.3.5.4 add a NOTE after f)1) saying "NOTE---This means the STA is disassociated in case of failure of reassociation to the same AP." In 11.3.5.5 add a NOTE after n) saying "NOTE---This means the STA remains associated but loses any security association in case of failure of reassociation to the same AP."

**Discussion**

This was discussed 2022-01-18:

* + 1. CID 1699 (MAC)
			1. Review comment
			2. Review the proposed change.
			3. Discussion on relevance of Note 2 being added.
			4. The comment seems to need more work, the proposed change was not immediately accepted.
			5. More work to resolve the CID. Mark Submission Required.
			6. Assign CID to Jouni MALINEN

MAC: 2022-01-18 22:58:50Z - Some belief that remaining associated is actually incorrect. More work needed.

Regarding the non-AP STA behavior, 11.3.5.4 is quite cleaer that the PTKSA shall be deleted before sending out the Reassociation Request frame. As such, there is no way to recover the previous state of the association if the reassociation attempt were to fail. This means that it is appropriate for the non-AP STA to drop its association state when receiving the Reassociation Response frame with nonzero status code: see 11.3.5.4, f) 1) and the NOTE proposed by the comment seems reasonable to add.

Regarding the AP behavior, the rules need to be different since the AP has to protect against attackers using injected Reassociation Request frames trying to terminate an existing association that has management frame protection enabled. The NOTE proposed by the comment for this case is not correct; the AP is dropping the SAs only in case of a successful reassociation. Failed reassociation does change the State from 4 to 3, though, when management frame protection is not in use and the reassociation is not part of fast BSS transition. This move to State 3 (while maintaining possible SAs) is not exactly ideal for the no-attacker case. However, the non-AP STA would move to State 2 in this case, so there is only limited impact from this to observable behavior. It does not seem necessary to add any note into 11.3.5.5 especially since the hopefully more reasonable case of management frame protection is described quite clearly not to change any state.

The changes proposed in the comment on top of the REVme/D1.2 text

**11.3.5.4 Non-AP and non-PCP STA reassociation initiation procedures**

Except when the association is part of a fast BSS transition, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA, and TPKSA (including temporal keys) held for communication with the AP or PCP by using the MLME-DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) before invoking an MLME-REASSOCIATE.request primitive.

If dot11InterworkingServiceActivated is true and the STA was associated to the ESS for unsecured access to emergency services, the SME shall submit the MLME-REASSOCIATE.request primitive with EmergencyServices parameter set to true.

The MM-SME of a non-AP and non-PCP STA may include an MMS element in an MLME-REASSOCIATE.request primitive. The MM-SME shall include in the MMS element the MAC address associated with the MLME SAP instance to which the primitive is submitted.

Upon receipt of an MLME-REASSOCIATE.request primitive that is part of an on-channel tunneling (see 11.31.5 (On-channel Tunneling (OCT) operation)), a non-AP and non-PCP STA shall follow the rules in 11.31.5 (On-channel Tunneling (OCT) operation) in addition to the reassociation procedures described below.

Upon receipt of an MLME-REASSOCIATE.request primitive, a non-AP and non-PCP STA shall reassociate with an AP or PCP using the following procedure:

a) If the STA is not associated in the same ESS or the state for the new AP or PCP is State 1, the MLME shall inform the SME of the failure of the reassociation by issuing an MLME-REASSOCIATE.confirm primitive, and this procedure ends.

b) The MLME shall transmit a Reassociation Request frame to the new AP or PCP. The RSNE contained in the MLME-ASSOCIATE.request primitive shall be included in the Reassociation Request frame. The RSNE shall specify exactly one pairwise cipher suite and exactly one AKM suite. If the MLME-REASSOCIATE.request primitive contained the EmergencyServices parameter equal to true, an Interworking element with the UESA field set to 1 shall be included in the Reassociation Request frame.

c) If a Reassociation Response frame is received with a status code of SUCCESS, the state variable for the new AP or PCP shall be set to State 4 or to State 3 if dot11RSNAActivated is true and the FT protocol is not used with respect to the new AP or PCP and, unless the old AP or PCP and new AP or PCP are the same, to State 2 with respect to the old AP or PCP, and the MLME shall issue an MLME-REASSOCIATE.confirm primitive to inform the SME of the successful completion of the reassociation.

If the MLME-REASSOCIATION.request primitive has the new AP’s or PCP’s MAC address in the CurrentAPAddress parameter (reassociation to the same AP or PCP), the following states, agreements, and allocations shall be deleted or reset to initial values:

1) All EDCAF state

2) Any block ack agreements that are not GCR agreements

3) Sequence number

4) Packet number

5) Duplicate detection caches

6) Anything queued for transmission

7) Fragmentation and reassembly buffers

8) Power management mode

9) WNM sleep mode

10) TPKSAs established with any peers

11) TSPECs

12) DMG TSPECs

13) GLK-GCR agreement

14) MSCS

15) SCS

If the reassociation is to the same AP (as described above), the following states, agreements, and allocations are not affected by the reassociation procedure:

1) PSMP sessions

2) Enablement/Deenablement

3) GDD enablement

4) TDLS agreements

5) MMSLs

6) GCR agreements that are not GLK-GCR agreements

7) DMS agreements

8) TFS agreements

9) FMS agreements

10) Triggered autonomous reporting agreements

11) FTM sessions

12) DMG SP and CBAP allocations

13) PTP TSPECs

In the case of reassociation to a different AP or PCP (the CurrentAPAddress parameter is not the new AP’s or PCP’s MAC address), all the states, agreements and allocations listed above are deleted or reset to initial values.

d) If a Reassociation Response frame is received with a status code of SUCCESS, a DMG STA shall write to each of the following MIB attributes the corresponding subfield of the DMG BSS Parameter Configuration field of the DMG Operation element received from the AP or PCP to which it requested reassociation:

1) dot11PSRequestSuspensionInterval from the PSRequestSuspensionInterval subfield

2) dot11MinBHIDuration from the MinBHIDuration subfield

3) dot11BroadcastSTAInfoDuration from the BroadcastSTAInfoDuration subfield

4) dot11AssocRespConfirmTime from the AssocRespConfirmTime subfield

5) dot11MinPPDuration from the MinPPDuration subfield

6) dot11SPIdleTimeout from the SPIdleTimeout subfield

7) dot11MaxLostBeacons from the MaxLostBeacons subfield

e) If an Association Response frame is received with a status code of SUCCESS at an MM-SME coordinated STA and the Single AID field within the MMS element is equal to 1, then

— For each of its MAC entities advertised within the MMS element and for which dot11RSNAActivated is true, the state is set to State 3. Progress from State 3 to State 4 occurs independently in each such MAC entity.

— For each of its MAC entities advertised within the MMS element and for which dot11RSNAActivated is false, the state is set to State 4.

— For each of its MAC entities advertised within the MMS element the state for any other AP or PCP which is State 3 or State 4 prior to the association request shall be set to State 2.

f) If a Reassociation Response frame is received with a status code other than SUCCESS or the reassociation fails to complete within dot11AssociationResponseTimeout:

1) Except when the association is part of a fast BSS transition, the state for the AP or PCP shall be set to State 2 with respect to the new AP or PCP.

NOTE—This means the STA is disassociated in case of failure of reassociation to the same AP.

2) The MLME shall issue an MLME-REASSOCIATE.confirm primitive to inform the SME of the failure of the reassociation. The ResultCode returned in the MLME-REASSOCIATE.confirm primitive indicates the cause of the failed reassociation attempt. Any misconfiguration or parameter mismatch, e.g., data rates required as basic rates that the STA did not indicate as supported in the STA’s Supported Rates and BSS Membership Selectors element, shall be corrected before the SME issues an MLME-REASSOCIATE.request primitive for the same AP or PCP. If the status code indicates the reassociation failed because of a reason that is not related to configuration (e.g., the AP or PCP is unable to support additional associations) and the Reassociation Response frame does not include a Timeout Interval element with Timeout Interval Type equal to 3 the SME shall not issue an MLME-REASSOCIATE.request primitive for the same AP or PCP until a period of at least 2 s has elapsed. If the status code indicates the reassociation failed and the Reassociation Response frame contains a Timeout Interval element with Timeout Interval Type equal to 3, the SME shall not issue an MLME-REASSOCIATE.request primitive for the same AP or PCP until the period specified in the Timeout Interval element has elapsed.

g) If an MLME-REASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and RSNA is required, and FILS authentication was not used, and the STA is in State 3, then the SME shall perform a 4-way handshake to establish an RSNA. As a part of a successful 4-way handshake, the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx\_Tx) primitive. If an MLME-REASSOCIATE.confirm primitive is received with a ResultCode of SUCCESS, and FILS authentication was used, and the STA is in State 3, then the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx\_Tx) primitive.

h) Upon receipt of the MLME-SETPROTECTION.request(Rx\_Tx) primitive, the MLME shall set the state of the STA to State 4.

**11.3.5.5 AP or PCP reassociation receipt procedures**

Upon receipt of a Reassociation Request frame from a STA the AP or PCP shall use the following procedure:

a) The MLME shall issue an MLME-REASSOCIATE.indication primitive to inform the SME of the reassociation request. The SME shall issue an MLME-REASSOCIATE.response primitive addressed to the STA identified by the PeerSTAAddress parameter of the MLME-REASSOCIATE.indication primitive. If the reassociation is not successful, the SME shall indicate a specific reason for the failure to reassociate in the ResultCode parameter. Upon receipt of the MLME-REASSOCIATE.response primitive, the MLME shall transmit a Reassociation Response frame.

b) If the state for the STA is State 1 and the STA is a non-DMG STA, the SME shall refuse the reassociation request by issuing an MLME-REASSOCIATE.response primitive with ResultCode NOT\_AUTHENTICATED.

c) AP with dot11InterworkingServiceActivated true only: If the MLME-REASSOCIATE.indication primitive has the EmergencyServices parameter set to true and the RSN parameter does not include an RSNE, the SME shall not reject the reassociation request on the basis that dot11RSNAActivated is true and dot11PrivacyInvoked is true thereby granting access, using unprotected frames (see 9.2.4.1.9 (Protected Frame subfield)), to the network for emergency services purposes.

d) Otherwise, in an RSNA the SME shall check the values received in the RSN parameter to see whether the values received match the security policy. If they do not, SME shall refuse the reassociation by issuing an MLME-REASSOCIATE.response primitive with a ResultCode indicating the security policy mismatch.

e) Otherwise, if the state for the STA is State 4, the STA has a valid security association, the STA has negotiated management frame protection, the reassociation is not a part of a fast BSS transition, the STA has not performed a successful SAE authentication after the current association was established, and there has been no earlier, timed out SA Query procedure with the STA (which would have allowed a new reassociation process to be started, without an additional SA Query procedure):

1) The SME shall refuse the reassociation request by issuing an MLME-REASSOCIATE.response primitive with ResultCode REFUSED\_TEMPORARILY and TimeoutInterval containing a Timeout Interval element with the Timeout Interval Type field set to 3 (Association Comeback time). If the SME is in an ongoing SA Query with the STA, the Timeout Interval Value field shall be set to the remaining SA Query period, otherwise it shall be set to dot11AssociationSAQueryMaximumTimeout.

2) The state for the STA shall be left unchanged.

3) Following this, if the SME is not in an ongoing SA Query with the STA, the SME shall issue one MLME-SA-QUERY.request primitive addressed to the STA every dot11AssociationSAQueryRetryTimeout TUs until an MLME-SA-QUERY.confirm primitive for the STA is received or dot11AssociationSAQueryMaximumTimeout TUs from the beginning of the SA Query procedure have passed. The SME shall increment the TransactionIdentifier by 1 for each MLME-SA-QUERY.request primitive, rolling it over to 0 after the maximum allowed value is reached.

4) If no MLME-SA-QUERY.confirm primitive for a STA is received within the dot11AssociationSAQueryMaximumTimeout period, the SME shall allow a subsequent reassociation process to be started without starting an additional SA Query procedure, except that the SME may deny a subsequent reassociation process with the STA if an MSDU was received from the STA within this period.

NOTE 1—Reception of an MSDU implies reception of a valid protected frame, which obviates the need for the SA Query procedure.

f) The SME shall refuse a reassociation request from a STA that does not support all the rates in the BSSBasicRateSet parameter and all of the membership selectors in the BSSMembershipSelectorSet parameter in the MLME-START.request primitive.

g) The SME shall refuse a reassociation request from an HT STA that does not support all of the MCSs in the Basic HT-MCS Set field of the HT Operation parameter in the MLME-START.request primitive.

h) The SME shall refuse a reassociation request from a VHT STA that does not support all of the <VHT-MCS, NSS> tuples indicated by the Basic VHT-MCS And NSS Set field of the VHT Operation parameter in the MLME-START.request primitive.

i) The SME shall refuse a reassociation request from an HE STA that does not support all of the <HE-MCS, NSS> tuples indicated by the Basic HE-MCS And NSS Set field of the HE Operation parameter in the MLME-START.request primitive.

j) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS, the SME has an existing SA with the STA, and an SA Query procedure with that STA has failed to receive a valid response (i.e., has not received an MLME-SA-QUERY.confirm primitive within the dot11AssociationSAQueryMaximumTimeout period), the SME shall issue an MLME-DISASSOCIATE.request primitive addressed to the STA with ReasonCode INVALID\_AUTHENTICATION.

NOTE 2—This MLME-DISASSOCIATE.request primitive generates a protected Disassociation frame. If the reassociation request was genuine, the STA has deleted the PTKSA by this point and so the protected Disassociation frame is ignored. The purpose is to inform a STA which has for some reason failed to respond to an SA Query procedure triggered by a forged reassociation request.

k) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS and the reassociation is not part of a fast BSS transition, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA, and TPKSA (including temporal keys) held for communication with the STA by using the MLME-DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)).

l) If the MLME-REASSOCIATE.indication primitive includes an MMS parameter, the AP or PCP shall take the following additional action, as appropriate:

1) If the Single AID field in the MMS parameter of the MLME-REASSOCIATE.indication primitive is equal to 1, the AP or PCP may allocate a single AID for all of the STAs included in the MMS element. If the AP or PCP allocates the same AID to all STAs whose MAC address was included in the MMS element, it shall include the MMS element received from the MM-SME coordinated STA in the MLME-REASSOCIATE.response primitive.

2) If the Single AID field is 0, the AP or PCP shall allocate a distinct AID for each STA specified in the MMS element.

NOTE 3—When the Single AID field is 0, a separate reassociation request/response exchange is performed for each STA specified in the MMS element, and this assigns the multiple AIDs for the STAs.

m) If a Reassociation Response frame with a status code of SUCCESS is acknowledged by the STA, the state for the STA shall be set to State 4, or to State 3 if dot11RSNAActivated is true and the reassociation is not part of a fast BSS transition.

n) If the ResultCode in the MLME-REASSOCIATE.response primitive is not SUCCESS and management frame protection is in use the state for the STA shall be left unchanged. If the ResultCode is not SUCCESS, management frame protection is not in use, and the reassociation is part of a fast BSS transition, the state for the STA shall be left unchanged. If the ResultCode is not SUCCESS, management frame protection is not in use, and the reassociation is not part of a fast BSS transition, the state for the STA shall be set to State 3 if it was State 4.

NOTE—This means the STA remains associated but loses any security association in case of failure of reassociation to the same AP.

o) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS, RSNA establishment is required, and the reassociation is not part of a fast BSS transition, and FILS is not in use, the SME shall attempt a 4-way handshake. Upon a successful completion of a 4-way handshake, the SME shall enable protection by issuing an MLME-SETPROTECTION.request(Rx\_Tx) primitive. If FILS authentication was used, the SME shall enable protection by generating an MLME-SETPROTECTION.request(Rx\_Tx) primitive. In either case, upon receipt of the MLME-SETPROTECTION.request(Rx\_Tx) primitive, the MLME shall set the state for the STA to State 4.

p) AP only: The SME shall inform the DS of any changes in the state of the STA.

q) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS and the CurrentAPAddress parameter in the MLME-REASSOCIATION.indication primitive is this AP’s or PCP’s MAC address (reassociation to the same AP or PCP), the AP or PCP shall match the non-AP STA’s treatment of the listed agreements and allocations as described in 11.3.5.4 (Non-AP and non-PCP STA reassociation initiation procedures) item c). The AP or PCP deletes or resets to initial values those items that the non-AP STA is required in 11.3.5.4 (Non-AP and non-PCP STA reassociation initiation procedures) item c) to delete or reset to initial values, and the AP or PCP does not modify the states, agreements and allocations that are listed as not affected by the reassociation procedure.

r) If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS and the CurrentAPAddress parameter in the MLME-REASSOCIATION.indication primitive is not this AP’s or PCP’s MAC address (reassociation to a different AP or PCP), all the states, agreements and allocations pertaining to the associating STA and listed in both numbered lists in 11.3.5.4 (Non-AP and non-PCP STA reassociation initiation procedures) item c) are deleted or reset to initial values.

**Proposed comment resolution for CID 1699**

REVISED. Make the changes proposed in the comment in 11.3.5.4. Do not change 11.3.5.5 since the SAs are not dropped on the AP in this case and even the State remains unchanged if management frame protection is used.

## CID 1813

**Owning Adhoc**: MAC

**Location**: 11.12

**Comment**

It is not clear whether, under MFP, BIP is used for all broadcast Management frames, or only robust broadcast Management frames (robust Management frames being defined in 12.2.7 as "Disassociation, Deauthentication, and robust Action frames.") E.g. the following suggest they apply to all broadcast Management frames, not just robust ones:

"Management frame protection protocols in an infrastructure BSS or IBSS apply to robust Management frames after RSNA PTK establishment for protection of individually addressed frames is completed and after delivery of the IGTK to protect group addressed frames."

"For all other group addressed Management frames [...]
-- The frames shall be encapsulated and protected using BIP (see 12.5.4 (Broadcast/multicast integrity protocol (BIP)))."

**Proposed Change**

In 4.5.4.9 change

"Management frame protection protocols in an infrastructure BSS or IBSS apply to robust Management frames after RSNA PTK establishment for protection of individually addressed frames is completed and
after delivery of the IGTK to protect group addressed frames. Beacon frames are protected in an
infrastructure BSS after delivery of the BIGTK."

to

"Management frame protection protocols in an infrastructure BSS or IBSS apply to robust Management
frames: after delivery of the PTK to protect individually addressed frames and after delivery of the IGTK to protect group addressed frames. Beacon frames are protected in an infrastructure BSS after delivery of the BIGTK.".

In 11.12 change

"For all other group addressed Management frames, the group addressed frame protection service shall take the following actions:"

to

"For all other group addressed robust Management frames, the group addressed frame protection service shall take the following actions:"

**Discussion**

This comment was discussed 2022-01-2119:03:51Z:

* + 1. CID 1813 (MAC)
			1. Review Comment
			2. Review context.
			3. Discussion on the reason for the need for the change.
			4. Discussion on if “robust” should be added or not.
			5. This is in Clause 4, so the existing text may have an error in it that needs to be reviewed outside the context of this CID, but if we are making changes in the clause, then maybe we should fix the “after deliver of the IGTK” phrase. To “after deliver of the BIGTK”
			6. Assign the comment to Jouni to complete the clean-up of the clause.

The identified additional issue with the current text is in the concept of protection of group addressed frames applying only when the applicable key has been delivered. However, the management frame protection protocol applies on the transmitter (e.g., an AP in the case of infrastructure BSS) all the time independent of IGTK/BIGTK delivery while the "after delivery" part is applicable on the receivers of these group addressed frames.

The changes proposed in the comment on top of the REVme/D1.2 text

**4.5.4.9 Management frame protection**

Robust Management frames are a set of Management frames that can be protected by the management frame protection service.

Management frame protection protocols in an infrastructure BSS or IBSS apply to robust Management frames: after delivery of the PTK to protect individually addressed frames and after delivery of the IGTK to protect group addressed frames. Beacon frames are protected in an infrastructure BSS after delivery of the BIGTK.

Management frame protection protocols in an MBSS apply to the following frames:

— Individually addressed robust Management frames after establishment of the RSNA MTK,

— Group addressed robust Management frames that are specified with Yes in the “Group Addressed Privacy” column of Table 9-79 (Category values) after establishment of the MGTK, and

— Group addressed robust Management frames that are specified with No in the “Group Addressed Privacy” column of Table 9-79 (Category values) after establishment of the IGTK.

See 14.7 (Mesh security) for details.

Management frame protection is implemented by CCMP, GCMP, and BIP confidentiality protocols and the SA Query procedure.

**11.12 Group addressed management frame protection procedures**

When management frame protection is negotiated, the MLME shall provide an encapsulation service for group addressed robust Management frames. All group addressed robust Management frames shall be submitted to this service for encapsulation and transmission.

In an MBSS, for group addressed Management frames that are specified with Yes in the Group Addressed Privacy column of Table 9-79 (Category values), the group addressed frame protection service shall take the following actions:

— The frames shall be encapsulated and protected with the MGTK using the group cipher negotiated during the AMPE exchange.

For all other group addressed robust Management frames, the group addressed frame protection service shall take the following actions:

— Management frame protection for multicast/broadcast shall be set using the MLMESETPROTECTION.request primitive with the Protectlist including a Key Type value of IGTK. A non-AP STA shall also set the Protect Type value to Rx. In an IBSS, a STA shall set the ProtectType value to Rx\_Tx. An AP shall set the Protect Type value to Tx.

— The IGTK shall be installed using the MLME-SETKEYS.request primitive with the value IGTK for the Key Type parameter of the SetKeyDescriptor.

— The frames shall be encapsulated and protected using BIP (see 12.5.3 (Broadcast/multicast integrity protocol (BIP))).

**The proposed changes for CID 1813 - START**

*Change 4.5.4.9 as shown (based on REVme/D1.2):*

**4.5.4.9 Management frame protection**

Robust Management frames are a set of Management frames that can be protected by the management frame protection service.

Management frame protection protocols in an infrastructure BSS or IBSS apply to individually addressed robust Management frames after delivery of the PTK, transmission of group addressed robust Management frames, and reception of group addressed robust Management frames after delivery of the IGTK. Beacon frames are protected in an infrastructure BSS and their validity can be verified after delivery of the BIGTK.

Management frame protection protocols in an MBSS apply to the following frames:

— Individually addressed robust Management frames after establishment of the RSNA MTK,

— Group addressed robust Management frames that are specified with Yes in the “Group Addressed Privacy” column of Table 9-79 (Category values) after establishment of the MGTK, and

— Group addressed robust Management frames that are specified with No in the “Group Addressed Privacy” column of Table 9-79 (Category values) after establishment of the IGTK.

See 14.7 (Mesh security) for details.

Management frame protection is implemented by CCMP, GCMP, and BIP confidentiality protocols and the SA Query procedure.

*Change 11.12 as shown (based on REVme/D1.2):*

**11.12 Group addressed management frame protection procedures**

When management frame protection is negotiated, the MLME shall provide an encapsulation service for group addressed robust Management frames. All group addressed robust Management frames shall be submitted to this service for encapsulation and transmission.

In an MBSS, for group addressed Management frames that are specified with Yes in the Group Addressed Privacy column of Table 9-79 (Category values), the group addressed frame protection service shall take the following actions:

— The frames shall be encapsulated and protected with the MGTK using the group cipher negotiated during the AMPE exchange.

For all other group addressed robust Management frames, the group addressed frame protection service shall take the following actions:

— Management frame protection for multicast/broadcast shall be set using the MLMESETPROTECTION.request primitive with the Protectlist including a Key Type value of IGTK. A non-AP STA shall also set the Protect Type value to Rx. In an IBSS, a STA shall set the ProtectType value to Rx\_Tx. An AP shall set the Protect Type value to Tx.

— The IGTK shall be installed using the MLME-SETKEYS.request primitive with the value IGTK for the Key Type parameter of the SetKeyDescriptor.

— The frames shall be encapsulated and protected using BIP (see 12.5.3 (Broadcast/multicast integrity protocol (BIP))).

**The proposed changes for CID 1813 - END**

**Proposed comment resolution for CID 1813**

REVISED. Make the changes marked as "The proposed changes for CID 1813" in <this document>. This includes the changes proposed in the comment and additional changes to 4.5.4.9 to clean up the description of management frame protection protocol applicability to group addressed frames.