IEEE P802.11
Wireless LANs

|  |
| --- |
| TGbe CC-34  |
| Date: 2021-03-17 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Michael Montemurro | Huawei |  |  | montemurro.michael@gmail.com  |

Background

This contribution proposes comment resolutions to comments received in CC34 on Clause 12 of TGbe D0.3

CIDs 2089, 1578, 2482, 2086, 2283, 2087, 2287, 2288, 2289, 2578, 2577, 2290, 1030, 2490, 1579, 2491, 2492, 1583, 2579

Straw\_poll: Adopt the resolutions to CIDs 2089, 1578, 2482, 2086, 2283, 2087, 2287, 2288, 2289, 2578, 2577, 2290, 1030, 2490, 1579, 2491, 2492, 1583, 2579

### General behavior for Security for MLO in TGbe

For MLO operation, Authentication, Association and the 4-way handshake work. (RSN negotiation and identity of Authenticator/Supplicant) work as follows:

* + In the MAC header for Authentication/Association frames indicates that the frame is sent from an Affilated STA to an Affiliated AP (and visa versa).
	+ For MLO, the ML element is included in Authentication/Association frame body and provides the non-AP MLD and AP MLD MAC addresses of the respective MLDs.
	+ For MLO, Authentication/Association occurs between non-AP MLD and AP MLD – therefore the Supplicant identity is the non-AP MLD and the Authenticator identify is the AP MLD.
	+ The non-AP MLD selects an RSNE for the association request based on the RSNE advertised by APs affiliated with the AP MLD.
	+ The addresses used in the 4-way handshake header are the affiliated STA and affiliated AP. The ML element must be added to the 4-way handshake to indicate the non-AP MLD MAC address, the AP MLD MAC address, and the link MAC addresses of the affiliated STAs and APs
	+ There is an upper and lower MAC associated with the MLD and affiliated STAs. The upper MAC is associated with the MLD and the lower MAC is associated with the affiliated STA/AP.
		1. Upper MAC includes:
			1. Distribution Service
			2. Integration Service
			3. Power management (with per-link power-save)
			4. Fragmentation/defragementation (unresolved in TGbe)
			5. Association/Dissasociation/Reassociation
			6. Authentication and Key Management (Authenticator/Supplicant)
			7. Encryption/Decryption of unicast frames
			8. Single Block ACK reordering buffer and common BA scoreboard across all links
			9. MLD discovery
		2. Lower MAC includes:
			1. Frame reception and MAC address filtering
			2. Scheduling/Queuing/Transmission
			3. Medium Access
			4. Radio Resource Management, Spectrum Management
			5. Group-addressed frame delivery
			6. Encapsulation/Decapsulation of Group-addressed Management frames
			7. Link-specific Management info exchange/indication (e.g. Beacons)
			8. Link-specific control info exchange/indication (e.g. RTS, CTS, ACKs, NDP, etc.)
			9. BlockAckReq/BlockAck, but “pull” from the common scoreboard
	+ Group-addressed frame transmission is not completely sorted out.
	+ The state machines for an affiliated link are inherited from the MLD state machine – therefore transition from State 0-4 should take place on a single link.
	+ For MLO, the GTK, iGTK, or BIGTK are link specific and are distributed in the 4-way handshake and group-key handshake between the AP MLD and non-AP MLD.
	+ When an Affiliated AP updates a group-key, it performs a group key exchange with non-AP MLDs through the AP MLD.
	+ Group key exchanges use the PTK for encapsulation so the exchange for a group key would need to be exchanged between the AP MLD, non-AP MLD for a given affiliated AP key update. The affiliated AP would need to know when key updates to affiliated STAs was complete.
	+ Group key exchanges can take place for one or more links simultaneously.
	+ Once the key is updated, the non-AP MLD would have to update the key for the affiliated STA
	+ It seems to make sense to allow for a single affiliated AP group key to be updated at one time.
	+ Example description of the procedure to update the GTK for the affiliated AP on link 1:
		1. Affiliated AP for link 1 generates new key.
		2. Affiliated AP instructs AP MLD Authenticator to update the key on link 1
		3. AP MLD Authenticator constructs GK message 1 for the lkey on link 1
		4. The AP MLD sends GK message 1 to non-AP MLD (on any link)
		5. Non-AP MLD supplicant receives message 1 and instructs the affiliated STA for link 1 to install the key on link 1
		6. The non-AP MLD constructs and sends the GK message 2 on the link it received message 1 from the AP MLD.

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2089 | 0.00 | 12 |   |   | I have concerns that the current specification does not provide clarity as to the exchange of frames and security procedures. It is unclear how MAC address are dealt with and how security is maintained during MLO procedures. | Please clarify how MAC address used in MLO work and how the MLD level peer to peer link maintains its security. The current complex use of STA/AP MAC address for associated STAs and APs seem to be out of step with MLD level peer to peer security. This approach seems needlessly complex and is not clearly specified. This must be clarified. |

### Discussion:

A security association for MLO is established between the two MLD peers. However, the frame exchanges to establish the RSNA take place between their respective affiliated STAs. Since the identities of the MLD Authenticator and the MLD Supplicant are required to establish a security association, the MLD peer addresses are included in Authentication, Association, and EAPoL-key frames.

### Proposed Resolution:

Revised. Make changes to the draft to clarify that the MLD MAC addresses are passed between affiliated STAs of the MLD peers so that an RSNA can be established between peer MLD SMEs.

***Insert the following paragraph at then end of clause 12.2.4***

When an RSNA is established between peer MLD SMEs, the MLD MAC address of the MLD transmitting the frame in an Authentication, Association, or EAPoL-key frames shall be included in that frame.

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 1578 | 117.21 | 12.6 |   |   | Restrictions on RSNE and RSNXE. Since each AP sends its own RSNE/RSNXE on its beacon, we need some consistent info between the beacons of the APs belonging to the same AP MLD for the non-AP MLD association to work | Require the RSNE to carry the same parameters except the AKM, which could be different for different APs (belonging to the same AP MLD). Similar requirements for RSNEX across all the APs. |
| 2482 | 111.12 | 12 |   |   | There is a general question on how each AP in AP MLD sets the RSNE and RSNXE capability. A lot of security capabilities are MLD level, and if each AP in an AP MLD has different security capbiilty, then non-AP MLD may need to determine capabilty support based on specific setup link, which is a bad idea. Suggest to clarify that most of the capability indication is the same except AKM due to possiblity of 6 GHz AP in an AP MLD. | Suggest to add the following: "Each AP in the AP MLD shall have the same RSNE/RSNXE indication except AKM suite. APs affiliated with an AP MLD shall have at least one common AKM suite. Non-AP MLD only provides one RSNE and one RSNXE in (Re)Assocaition Request frame" |

### Discussion:

The RSNA is established between MLD SME peers. Affiliated STAs peers communicate with each other based on that state.

The non-AP MLD and AP MLD can only negotiate one AKM across all links. If the non-AP MLD negotiates a link with the affiliated AP in the 6 GHz band, the non-AP MLD would be bound by the AKM requirements for the 6 GHz link.

The AP MLD can advertise multiple AKMs, so if a non-AP MLD negotiates an association that does not include the affiliated 6 GHz AP, it could select an AKM that does not meet the requirements for connecting to the 6 GHz AP.

With respect to the RSNE and RSNXE varying across links for the AP:

RSNE:

* Version: shall be same
* Group Data Cipher Suite: shall be the same
* Pairwise Cipher Suite List: shall be the same
* AKM Suite List: may have link-specific differences for non-MLO cases; shall include a common selector for all MLO links
* PMKID List: shall be empty – the same
* Group Management Cipher Suite: shall be the same)

RSN capabilities: (Shall be the same)

RSNE capabilities:

* Preauthentication: Same because Auth is MLD-level
* No Pairwise: Same and shall be set to 0 (must be 0 in practice)
* PTKSA Replay Counter: Same because PTK is MLD-level
* GTKSA Replay Counter: Even though GTK is per-link, Same should be OK
* MFPR: may differ between links because MFPR needs to be 0 on links that support legacy-non-6-GHz-STAs while it shall be 1 on the 6 GHz band. This has no connection to how Deauthentication/Disassociation frames are used since use of management frame protection is determined based on MFPC, not MFPR.
* MFPC: shall be same: Same and shall be set to 1
* Joint Multi-Band RSNA: Same. This field will not be used by MLD
* PeerKey Enable: Same and shall be set to 0
* SPP A-MSDU Capable: shall be same: Same because A-MSDU MLD-level
* SPP A-MSDU Required: shall be same: Same because A-MSDU MLD-level
* PBAC: Same because BA session is MLD-level
* Ext Key ID: Same because keys are MLD-level
* OCVC: Same

 Extended RSNE capabilities:

* Protected TWT Operations Support (Same)
* SAE hash-to-element (Same)”

RSNXE: (the same)

* Protected TWT: no strong opinion, but I guess TWT might not be supported on all bands(?)
* SAE H2E: shall be set to same (which shall be 1 if there is a link on 6 GHz)

### Proposed Resolution:

Revised. The RSNE and RSNXE will be the same except for the AKM Suite List. At least AKM suite selector for affiliated STAs needs to be the common among all affiliated APs since the MLD SME negotiates the RSNA. Also, since the encapsulation/decapsulation is done by the MLD peers, there can only be a single AKM negotiated between MLD peers across all affiliated STA links.

Instruct the editor to make the changes under Proposed Resolution in <this>.

***Insert the following paragraph after the third paragraph of clause 12.6.2***

"All APs affiliated with an AP MLD shall advertise the same RSNE and RSNXE if included, with the exception of the AKM Suite List and the MFPR subfield of the RSN Capabilities field. All APs affiliated with an AP MLD shall advertise at last one common AKM suite selector in the AKM Suite List field.”

***Modify clause 12.6.3 as follows.***

* RSNA policy selection in an infrastructure BSS

The requirements in this subclause apply to a STA when dot11RSNAActivated is true.

RSNA policy selection in an infrastructure BSS utilizes the normal IEEE 802.11 association procedure. RSNA policy selection is performed by the associating STA. The STA does this by including an RSNE in its (Re)Association Requests.

In an RSN, an AP shall not associate with pre-RSNA STAs, i.e., with STAs that fail to include the RSNE in the (Re)Association Request frame.

An SME initiating an association shall insert an RSNE into its (Re)Association Request via the MLME-ASSOCIATE.request or MLME-REASSOCIATE.request primitive, when the targeted AP indicates RSNA support. The initiating STA’s RSNE shall include one authentication and pairwise cipher suite from among those advertised by the targeted AP in its Beacon and Probe Response frames. For MLO, the initiating non-AP MLD’s RSNE shall include one AKM suite selector and one pairwise cipher suite selector that are common among those advertised by the APs affiliated with the targeted AP MLD in its MLD Probe Response frame. It shall also specify the group cipher suite specified by the targeted AP. If at least one RSNE field from the AP’s RSNE fails to overlap with any value the STA supports, the STA shall decline to associate with that AP. An HT STA shall eliminate TKIP as a choice for the pairwise cipher suite if CCMP-128 or CCMP-256 is advertised by the AP or if the AP included an HT Capabilities element in its Beacon and Probe Response frames. The elimination of TKIP as a choice for the pairwise cipher suite may result in a lack of overlap of the remaining pairwise cipher suite choices, in which case the STA shall decline to create an RSN association with that AP.

If an (#136)(#2512) AP receives a (Re)Association Request frame that includes an RSNE and if it chooses to accept the association as a secure association, then it shall use the authentication and pairwise cipher suites in the (Re)Association Request frame, unless the AP includes an optional second RSNE in message 3 of the 4-way handshake. If the second RSNE is supplied in message 3, then the pairwise cipher suite used by the security association, if established, shall be the pairwise cipher from the second RSNE.

Note 1 – APs affiliated with AP MLD indicate the same pairwise cipher suite list. Hence, any pairwise cipher suite selected from an AP affiliated with the AP MLD will be common among those advertised by the APs affiliated with the targeted AP MLD.

Note 2 – For MLO, (Re)Association frames are transmitted between an affiliated STA and affiliated AP, however the association takes place between the non-AP MLD and the AP MLD.

**12.6.3.1 RSNA policy selection for MLO**

If an AP MLD Authenticator receives a (Re)Association Request frame that includes an RSNE and if it chooses to accept the association as a secure association, then it shall use the AKM suite and pairwise cipher suite in the (Re)Association Request frame to establish an RSNA with a non-AP MLD. APs affiliated with the AP MLD shall use the group cipher suite indicated by the non-AP MLD in the (Re)Association Request frame.

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2086 | 111.25 | 12.3.3.1 |   |   | An IEEE 802.11 authentication exchange is not an exchange between two STAs or an non-AP MLD and an AP MLD. While authentication frames must be exchanged as individually addressed frames between a pair of STAs, this does not mean authentication is performed by the AP in the pair or in the MLD case by the AP MLD. Authentication is a higher layer service which is provided to 802.11 it is not part of 802.11. So stating that the authentication exchange is between an AP MLD and non-AP MLD does not make any sense. | clarify that an non-AP MLD shall completed an IEEE 802.11 authentication exchange prior to association. Remove any reference to the AP MLD. |
| 2283 | 111.25 | 12.3.3.1 |   |   | This cited sentence is not required. | Make the following change to the previous sentence. Change "non-DMG STA" to "non-DMG STA or non-AP MLD" |
|   | 111.14 | 12.3.3.1 |   |   | "14 In 12.3.3 (Pre-RSNA authentication), the reference of a “STA” means that the “STA” is not affiliated with an15 MLD unless specified otherwise.1718 In 12.3.3 (Pre-RSNA authentication), when referring to MLD authentication, “SME” is the entity that manages19 the MLD." -- see Clause 11 comments | As it says in the comment |

### Discussion:

It’s not clear what the unnumbered comment is proposing as a change to the draft, but “SME” is used only once in clause 12.3.3 (note that shared-key authentication is obsolete. SME is used extensively throughout clause 12. It would be better to move the text at the beginning of the clause referring to “SME” to the Conventions section, clause 12.

Similarly, for “STA” and affiliated STA, it would be better to move this text to clause 12.1. There are numerous references to STA in clause 12.

With respect to CID 2086, in an infrastructure network, an IEEE 802.11 Authentication exchange takes place between a non-AP STA and an AP. The text in the base standard is generalized because it can refer to IBSS operation.

The first two paragraphs of base text with the changes are:

“In 12.3.3 (Pre-RSNA authentication), the reference of a “STA” means that the “STA” is not affiliated with an MLD unless specified otherwise.

In 12.3.3 (Pre-RSNA authentication), when referring to MLD authentication, “SME” is the entity that manages the MLD.

In an infrastructure BSS, a non-DMG STA shall complete an IEEE 802.11 authentication exchange prior to association. A non-AP MLD shall complete an IEEE 802.11 authentication exchange with an AP MLD prior to association with the AP MLD. A DMG STA not in an IBSS shall complete an IEEE 802.11 authentication exchange prior to association when an authentication algorithm other than the Open System authentication algorithm is requested. A DMG STA shall not perform an IEEE 802.11 authentication exchange using the Open System authentication algorithm. An IEEE 802.11 authentication exchange is optional in an IBSS.

All Authentication frames shall be individually addressed, as IEEE 802.11 authentication is performed between pairs of STAs, i.e., group addressed authentication is not allowed. Deauthentication frames are advisory and may be sent as group addressed frames.”

CIDs 2086 and 2283 refer to the second paragraph above. The base text describes the requirement that a STA complete successful authentication prior to association. The only difference between STA behavior and MLD behavior is that the authentication exchange for MLD take place through frame exchanges over an affiliated AP link. Therefore a change which describes the requirement and process would be:

Change the cited text to “In an infrastructure BSS, a non-DMG STA or MLD shall complete an IEEE 802.11 authentication exchange prior to association. A non-AP MLD and AP MLD exchange authentication frames through an affiliated STA and affiliated AP, respectively.”

### Proposed Resolution:

Revised. Make the following changes to the draft to clarify the use of the terms SME in clause 12 and distinguish the affiliated STA and affiliated AP from STA and AP in the context of MLO.

Instruct the editor to make the changes under Proposed Resolution in <this>.

In clause 12.1, add the following paragraph at the end of the clause:

“ “STA” means that the “STA” is not affiliated with an MLD unless specified otherwise.

For MLO, the “SME” is the entity that manages the MLD.”

***In clause 12.3.3.2.1, change:***

Delete

“In 12.3.3 (Pre-RSNA authentication), the reference of a “STA” means that the “STA” is not affiliated with an MLD unless specified otherwise.

In 12.3.3 (Pre-RSNA authentication), when referring to MLD authentication, “SME” is the entity that manages the MLD.”

Change

“In an infrastructure BSS, a non-DMG STA shall complete an IEEE 802.11 authentication exchange prior to association. A non-AP MLD shall complete an IEEE 802.11 authentication exchange with an AP MLD prior to association with the AP MLD.”

To

“In an infrastructure BSS, a non-DMG STA shall complete an IEEE 802.11 authentication exchange prior to association. For MLO a non-AP MLD and AP MLD shall complete an IEEE 802.11 authentication prior to association through the exchange of Authentication frames between an affiliated STA and affiliated AP, respectively.”

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
|   | 112.04 | 12.3.3.2.1 |   |   | "4 NOTE—Open system authentication between MLDs is done before multi-link (re)setup (see 35.3.5.1 (Multi-link (re)setup5 procedure) and 11.3 (STA/MLD authentication and association))." -- Not clear. 1) If done before ML setup, then can’t be between MLDs. 2) If doesn’t involve MLDs, then shouldn’t refer to MLDs in12.3.3 | As it says in the comment |
| 2087 | 112.04 | 12.3.3.2.1 |   |   | If authentication between MLDs is done before multi-link setup what is the MAC address used for authentication? Is it the one of the STAs affiliated with the non-AP MLD? If so how does the authentication transfer to the no-AP MLD? If it is at the MLD level how can it be know prior to multi-link setup? This just doesn't make any sense to me. | Please consider how authentication, MAC addresses, association, and ML setup interact. I don't think this has been thought through properly. |

### Discussion:

The two comments are referring to the following note:

“NOTE—Open system authentication between MLDs is done before multi-link (re)setup (see 35.3.5.1 (Multi-link (re)setup procedure) and 11.3 (STA/MLD authentication and association)).”

In this case, there is no need to refer to 35.3.5.1 or 11.3 because neither clause provides additional detail relative to Open Authentication frame exchanges. The key information is:

* One MLD will act as the requester and the other MLD will act as the responder.
* The frame exchange between the requester and responder occurs between STAs that are affiliated with the MLDs.

### Proposed Resolution:

Revised. Make the changes to the note to clarify how Open System authentication works for MLO.

At the cited location, change

“NOTE—Open system authentication between MLDs is done before multi-link (re)setup (see 35.3.5.1 (Multi-link (re)setup procedure) and 11.3 (STA/MLD authentication and association)).”

To

“NOTE—Open system authentication between the requester MLD and the responder MLD is completed through frame exchanges between STAs affiliated with the MLDs, respectively.”

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
|   | 112.2 | 12.4.1 |   |   | "14 In 12.3.3 (Pre-RSNA authentication), the reference of a “STA” means that the “STA” is not affiliated with an15 MLD unless specified otherwise.1718 In 12.3.3 (Pre-RSNA authentication), when referring to MLD authentication, “SME” is the entity that manages19 the MLD." -- see Clause 11 comments | As it says in the comment |
| 2864 | 112.30 | 12.4.1 |   |   | An "AP STA" is an "AP". | Change all occurances of "AP STA" to "AP". |

### Discussion:

This is covered by 11-21/260

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
|   | 113.32 | 12.4.3 |   |   | "32 In an infrastructure BSS or an AP MLD" -- A BSS and an MLD are not the same kind of thing. | delete and change “with the AP MLD, respectively,” to “with an AP MLD” |
| 2284 | 113.32 | 12.4.3 |   |   | This should just be MLD AP since that is the identity of the Authenticator. | Change "AP or APs affiliated with the AP MLD, respectively," to "AP MLD" |
| 2285 | 113.32 | 12.4.3 |   |   | I hope for MLD operations that with MLDs this is still a BSS. | Delete "or an AP MLD" |
| 2576 | 114.63 | 12.4.5.2 |   |   | It is not apparent how the two MLDs find out each other's MLD MAC Addresses. Presumably, AP MLD would advertise its MLD MAC Address in Beacon/Probe Response frames and non-AP MLD would include its MLD MAC Address in the Authentication frame. However, can't find this information explicitely anywhere in D0.3. Clause 9 only mentions that ML element may be carried in Beacon/ Probe Response frames and Authentication frames but the content of ML element in these frames are not explained in clause 35. | Explain how the two MLDs would find out each other's MLD MAC Addresses for the PWE generation in SAE. |
| 2286 | 113.36 | 12.4.3 |   |   | This should just be MLD AP since that is the identity of the Authenticator. | Replace "or Aps affiliated with the AP MLD" with "or AP MLD" |

### Discussion:

This is covered by 11-21/260

”

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
|   | 114.63 | 12.4.5.2 |   |   | "between two STAs or MLD-A-MAC and MLD-B-MAC shall be used in the computation of val" is not clear | say MAC-A and MAC-B in the equation, and then in the “where” below describe how they map in the MLD and non-MLD cases |
| 2487 | 114.64 | 12.4.5.2 |   |   | The formula of val computation needs to be specified between two MLDs. Simply changing (STA-A-MAC, STA-B-MAC) to (MLD-A-MAC, MLD-B-MAC) for the formula between two MLDs | Changing (STA-A-MAC, STA-B-MAC) to (MLD-A-MAC, MLD-B-MAC) for the formula between two MLDs. |

### Discussion:

This is covered by 11-21/260.

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2287 | 116.49 | 12.5.3.3.7 |   |   | This should just be MLD AP | Change "STA" to "STA or MLD" |
| 2288 | 116.52 | 12.5.3.3.7 |   |   | This should just be MLD AP | Change "STA" to "STA or MLD" |
| 2289 | 117.13 | 12.5.5.3.6 |   |   | This should just be MLD AP | Change "STA that is not affiliated with an MLD and each MLD" to "STA or MLD" |

### Discussion:

The cited paragraph in clause 12.5.3.3.7 is:

“The PN values sequentially number each MPDU. Each transmitter STA that is not affiliated with an MLD and each MLD shall maintain a single PN (48-bit counter) for each PTKSA and GTKSA. Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA. The PN shall be implemented as a 48-bit strictly increasing integer, initialized to 1 when the corresponding temporal key is initialized or refreshed.”

The cited paragraph in clause 12.5.5.3.6 is:

“The PN values sequentially number each MPDU. Each transmitter STA that is not affiliated with an MLD and each MLD shall maintain a single PN (48-bit counter) for each PTKSA and GTKSA. Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA. The PN shall be implemented as a 48-bit strictly increasing integer, initialized to 1 when the corresponding temporal key is initialized or refreshed.”

There is a convention statement STA unless qualified by “affiliated” means non-MLD STA. That was added to Conventions clause, 12.1, by CID 2086.

For MLD, the encapsulation of data frames is performed by the MLD MAC and the encapsulation of management frames is performed by the affiliated STA MAC.

So, the paragraph in 12.5.3.3.7 should be modified as follows:

““The PN values sequentially number each MPDU. Each STA shall maintain a single PN (48-bit counter) for each PTKSA and GTKSA. Each MLD shall maintain a single PN for each PTKSA. Each STA that is affiliated with an MLD shall maintain a single PN for the GTKSA. The PN shall be implemented as a 48-bit strictly increasing integer, initialized to 1 when the corresponding temporal key is initialized or refreshed.”

Similarly, the paragraph in 12.5.5.3.6 should be modified as follows

“The PN values sequentially number each MPDU. Each STA shall maintain a single PN (48-bit counter) for each PTKSA and GTKSA. Each MLD shall maintain a single PN for each PTKSA. Each STA that is affiliated with an MLD shall maintain a single PN for the GTKSA. The PN shall be implemented as a 48-bit strictly increasing integer, initialized to 1 when the corresponding temporal key is initialized or refreshed.”

### Proposed Resolution:

Revised. The cited text has been updated to state the requirement that the MLD maintains a PN for the PTKSA and the affiliated STAs maintain a PN for the GTKSA.

At the cited location in 12.5.3.3.7:

At 116.49, change “Each transmitter STA that is not affiliated with an MLD and each MLD”

to

“Each STA”

At 116.50, change “Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA.”

To

“Each MLD shall maintain a single PN for each PTKSA. Each STA that is affiliated with an MLD shall maintain a single PN for the GTKSA.”

At the cited location in 12.5.5.3.6

At 117.13, change “Each transmitter STA that is not affiliated with an MLD and each MLD”

To

“Each STA”

At 117.15, change “Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA.”

To

“Each MLD shall maintain a single PN for each PTKSA. Each STA that is affiliated with an MLD shall maintain a single PN for the GTKSA.”

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2578 | 116.17 | 12.5.5.3.6 |   |   | "The PN shall be implemented as a 48-bit strictly increasing integer,..."Presumably the PN would not be incremented when a frame is re-transmitted by an MLD on the same link (baseline rule). The PN should not be incremented when the frame is re-transmitted by the MLD on another link to prevent the frame being dropped as replayed frame by the receiving MLD. | Clarify that the PN shall not be incremented when a frame is re-transmitted by the MLD on another link. |
| 2577 | 116.53 | 12.5.3.3.7 |   |   | "The PN shall be implemented as a 48-bit strictly increasing integer,..."Presumably the PN would not be incremented when a frame is re-transmitted by an MLD on the same link (baseline rule). The PN should not be incremented when the frame is re-transmitted by the MLD on another link to prevent the frame being dropped as replayed frame by the receiving MLD. | Clarify that the PN shall not be incremented when a frame is re-transmitted by the MLD on another link. |

### Discussion:

MPDUs are not modified on retransmission. (see REVmd, page 2594, line 51; page 2595, line 22, and 2608, line 33). Since the MPDU encapsulation is done at the MLD, the MPDU encapsulation will not be modified, even if it is transmitted over another link.

### Proposed Resolution:

Revised. Add a note to clarify that retransmitted MPDUs are not encapsulated with a new PN when transmitted on another link. Incorporate the changes under “Proposed Resolution” in <>

***Add an additional note in clause 12.5.3.3.1 as follows:***

* General

NOTE—Retransmitted MPDUs are not modified on retransmission. For MLO, MPDUs are not encapsulated with a new PN when retransmitted on another link.

* General

 NOTE—Retransmitted MPDUs are not modified on retransmission. For MLO, MPDUs are not encapsulated with a new PN when retransmitted on another link.

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2533 | 116.51 | 12.5.3.3.7 |   |   | "Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA. " It's unclear why PN check for group addressed frames should be at the individual AP level instead of at the MLD level associated with a common GTK used by all links. | Unify the design for unicast and groupcast frame delivery. Specify that a common GTK and PN check applied to group addressed frames delivery over all links. |
| 2534 | 117.14 | 12.6.1.1.2 |   |   | "Each transmitter STA that is affiliated with an MLD shall use the PN that is maintained by the MLD for the PTKSA and the PN that is maintained by the AP affiliated with the AP MLD for the GTKSA. " It's unclear why PN check for group addressed frames should be at the individual AP level instead of at the MLD level associated with a common GTK used by all links. | Unify the design for unicast and groupcast frame delivery. Specify that a common GTK and PN check applied to group addressed frames delivery over all links. |
| 1853 | 119.41 | 12.7.2 |  |  | The 802.11be should have only ML level GTK. The link specific group key makes the group frames reception complicated | Please make GTK to be in ML level. |

### Discussion:

Contacted the commenter. The commenter has prepared <https://mentor.ieee.org/802.11/dcn/21/11-21-0041-01-00be-group-addressed-frame-delivery-methods-for-mlo.pptx>

### Proposed Resolution:

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
|   | 117.42 | 12.6.1.1.2 |   |   | "For MLD ," should be "For MLO", | As it says in the comment |
|   | 118.20 | 12.6.1.1.6 |   |   | "Authenticator MAC address or BSSID or MLD MAC address of the AP MLD " -- Is the Authenticator required to reside in the MLD? | As it says in the comment |

### Discussion:

In 12.6.1.1.2, the cited sentence in the PMKID definition is:

“ For MLD, the Authenticator’s MAC address is the MLD MAC address of the AP MLD and the peer’s (Supplicant’s) MAC address is the MLD MAC address of the non-AP MLD.”

“For MLO” would make more sense than “For MLD”.

In 12.6.1.1.6, the cited text is in the PTKSA definition:

" The PTKSA consists of the following:

* PTK, where the PTK includes the KDK when WUR frame protection is negotiated
* Pairwise cipher suite selector, and when WUR frame protection is negotiated, the cipher suite selector 00-0F-AC:6 (BIP-CMAC-128) for individually addressed WUR Wake-up frames
* Supplicant MAC address or STA’s MAC address or MLD MAC address of the non-AP MLD
* Authenticator MAC address or BSSID or MLD MAC address of the AP MLD”

Authentication, association and the 4-way handshake establish a security association between the AP MLD and non-AP MLD. The authenticator identity is the MAC address of the AP MLD and the supplicant identity is the MAC address of the non-AP MLD.

### Proposed Resolution:

Revised. The Authenticator or Supplicant resides in the SME for the MLD. Therefore the Authenticator identity is the AP MLD MAC address and the Supplicant identity is the non-AP MLD MAC address.

At 117.42, change “For MLD” to “For MLO”

At 118.13, make the following changes:

“The PTKSA consists of the following:

* PTK, where the PTK includes the KDK when WUR frame protection is negotiated
* Pairwise cipher suite selector, and when WUR frame protection is negotiated, the cipher suite selector 00-0F-AC:6 (BIP-CMAC-128) for individually addressed WUR Wake-up frames
* Supplicant MAC address or STA’s MAC address ~~or MLD MAC address of the non-AP MLD~~
* Authenticator MAC address or BSSID ~~or MLD MAC address of the AP MLD~~
* For MLO, the Authenticator’s MAC address is the AP MLD MAC address and the Supplicant’s MAC address is the the non-AP MLD MAC address”

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 1029 | 118.34 | 12.7.8 |  |  | Update 12.7.8 to cover PTK establishment for a TDLS link involving a STA of a non-AP MLD | Commenter will provide a contribution (also see details in 11-20/1692) |

### Discussion:

Reached out to commenter on status of contribution. The proposed resolution in captured in DCN 11-20/1692.

### Proposed Resolution:

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 1028 | 118.34 | 12.7.7 |  |  | Update clause 12.7.7 to handle GTK/IGTK/BIGTK delivery for other links of the AP MLD | As in comment |
| 2505 | 122.01 | 12.7.7 |  |  | Group handshake is used to update the group key. Similar to the design we add in 12.7.6. Allow group key handshake to update keys of all setup links in one excahgne. | Follow the design in 12.7.6 for group key handshake by allowing KDE of GTK, IGTK, BIGTK of other links to be included in group key handshake to complete update in one handshake. |

### Discussion:

These CIDs are covered by 11-21/300

### Proposed Resolution:

### Comment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Page** | **Clause** | **Duplicate of CID** | **Resn Status** | **Comment** | **Proposed Change** |
| 2290 | 122.07 | 12.7.6.4.4 |  |  | A BIGTK for the AP MLD should not be required. It should only be required for the affiliated AP-STA links | change "the BIGTK KDE for the current link and the MLO BIGTK KDE for each of the other setup links," to "the MLO BIGTK KDE for each of the setup links". |
|   | 119.36 | 12.7.2 |   |   | "The format of the MLO GTK KDE is shown in Figure 12-36a (MLO GTK KDE format). " -- What is Length in this figure? | As it says in the comment |
|   | 119.44 | 12.7.2 |   |   | Figure 12-36a—MLO GTK KDE format -- The two reserved fields are pointless | Delete them |
|   | 119.47 | 12.7.2 |   |   | Figure 12-36a—MLO GTK KDE format -- What is in the Key ID field? | As it says in the comment |
|   | 119.56 | 12.7.2 |   |   | "56 The KeyRSC field contains the Key RSC field that corresponds to the link for which this GTK applies (see57 Table 12-9 (KDE selectors))." -- Why is this being passed? It’s not in the one for the non-MLO GTK KDE | As it says in the comment |
| 1030 | 119.39 | 12.7.2 |   |   | Resolve TBD (applies to MLO IGTK & MLO BIGTK). Link ID size 4 bits | As in comment |
| 2490 | 119.40 | 12.7.2 |  |  | Link ID can be 4 bits as defined in Figure 9-632b--MLD Parameters subfield format. | make Link ID 4 bits. |
| 1579 | 119.42 | 12.7.2 |  |  | The link ID field in the GTK/IGTK/BIGTK KDEs is 4 bits | k=4 because Link ID field is 4 bits. Adjust the k in MLO GTK KDE, MLO IGTK KDE and MLO BIGTK KDE |
| 2491 | 120.04 | 12.7.2 |  |  | Link ID can be 4 bits as defined in Figure 9-632b--MLD Parameters subfield format. | make Link ID 4 bits. |
| 2492 | 121.24 | 12.7.2 |  |  | Link ID can be 4 bits as defined in Figure 9-632b--MLD Parameters subfield format. | make Link ID 4 bits. |
| 1583 | 121.47 | 12.7.6 |  |  | Currently the AP MAC addresses and the STA MAC addresses of the ML setup links are not protected during the ML setup. Need to exchange the MAC addresses of the STAs of the non-AP MLD and the MAC addresses of the APs of the AP MLD during 4WH (where the payload is protected) | Adopt the SPs in 20/727R0 |
| 2579 | 122.01 | 12.7.6.4 |  |  | "For generating a single PTK between a non-AP MLD associated with an AP MLD,.."This sentence is incomplete, RSNE and possibly ML element is required to generate the single PTK. | Complete the phrase regarding generation of the single PTK and clarify how the RSNE and MLD MAC Address is obtained for use in the PTK generation. |

### Discussion:

See general discussion at the beginning of the document. The 4-way handshake and group key handshake changes are given below

From Jouni:

“My understanding is that there shall be a common pairwise cipher and a common AKM included in RSNE from the AP side on all links. The non-AP STA side can learn that through scanning (including option of getting information from all links in a single response). Other parts of RSNE/RSNXE can be different between links, e.g., to support PMF transition (MFPC=1+MFPR=0 on 2.4/5 GHz and MFPC=1+MFPR=1 on 6 GHz), to support different group ciphers or different number of QoS receiver counters, or some future RSN extensions. As such, the payload of RSNE/RSNXE cannot be mandated to be bitwise identical. This also implies that there needs to be an extension to the mechanism in which the unprotected RSNE/RSNXE information from Beacon/Probe Response frames is confirmed in protected frames. This could be done, e.g., by adding a new IE/KDE into EAPOL-Key msg 3/4 to include the RSNE/RSNXE from all links other than the one on which the EAPOL frame is being transmitted. At least in theory, there might be some use for confirming non-AP STA side capabilities for each link as well (e.g., EAPOL-Key msg 2/4), but that may be of less importance due to there being only single RSNE/RSNXE included in the (Re)Association Request frame. If this is left, protection against link-specific downgrade attacks is left on the non-AP STA side.”

An example 4-way handshake is shown below:



### Proposed Resolution for 4-way handshake:

Revised. The 4-way handshake has been modified to include the MLD addresses and link-specific RSNE/RSNXEs, and fixes for the group-key KDEs.

Add a new KDE into EAPOL-Key msg 2/3 to include the RSNE/RSNXE from all links other than the one on which the EAPOL frame is being transmitted.

Instruct the editor to make the changes under Proposed Resolution in <this>.

* + 1. **EAPOL-Key frames**

***Modify item g) as follows:***

* **Key RSC**. This field contains the current receive sequence counter (RSC) for the GTK being installed. It is used in message 3 of the 4-way handshake and message 1 of the group key handshake, where it is used to synchronize the IEEE 802.11 replay state. It may also be used in the Michael MIC Failure Report frame, to report the TSC field value of the frame experiencing a MIC failure. It shall contain 0 in other messages. If the RSC is less than 8 octets in length, it is stored in the first octets and the remaining octets are set to 0. The least significant octet of the RSC is in the first octet of the Key RSC field. The RSC for TKIP is the TKIP sequence number (TSC); for CCMP and GCMP it is the packet number (PN); see Table 12-8 (Key RSC field).

For MLO, the Key RSC field is set to 0 in all messages.

***Insert the following news rows to KDE selectors while maintaining the numerical order and updating the reserved range:***

|  |
| --- |
| * KDE selectors
 |
| OUI | Data type | Meaning |
| 00-0F-AC | 16 | MLO GTK KDE |
| 00-0F-AC | 17 | MLO IGTK KDE |
| 00-0F-AC | 18 | MLO BIGTK KDE |
| 00-0F-AC | 19 | MLO Link KDE |
| 00-0F-AC | 20–255 | Reserved |

***Replace figure 12-36a with the following figure***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Key ID | Tx | Reserved | LinkID | PN | GTK |
| Bits: | 2 | 1 | 1 | 4 | 48 | (Length – 11) × 8 |

***At page 119, line 47, make the following changes:***

~~If the value of the Tx field is 1, then the IEEE 802.1X component shall configure the temporal key derived from this KDE into its IEEE 802.11 MAC for both transmission and reception.~~

~~If the value of the Tx field is 0, then the IEEE 802.1X component shall configure the temporal key derived from this KDE into its IEEE 802.11 MAC for reception only.~~

The definitions of the Key ID, Tx, and GTK fields are the same as in the GTK KDE described above.

The LinkID field contains the link identifier that corresponds to the link this GTK applies.

The PN field contains the packet number (PN) and is formatted as described in Table 12–8 (KeyRSC field).

~~The KeyRSC field contains the Key RSC field that corresponds to the link for which this GTK applies (see Table 12-9 (KDE selectors)).~~

***Replace figure 12-42a with the following figure:***.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Key ID | IPN | Reserved | LinkID | IGTK |
| Bits: | 16 | 48 | 4 | 4 | (Length – 13) × 8 |
|  |

|  |
| --- |
|  |

***At page 120, line 9, make the following changes:***

The definitions of the Key ID, IPN, and IGTK fields are the same as in the IGTK KDE described above.

The LinkID field contains the link identifier that corresponds to the link this IGTK applies.

Change the seventh paragraph as follows:

The following EAPOL-Key frames are used to implement the three different exchanges:

* **4-way handshake message 1** is an EAPOL-Key frame with the Key Type subfield equal to 1. Use of the Key Data field to indicate a PMKID when a cached PMKSA is being used in this key derivation is defined in 12.6.10.3 (Cached PMKSAs and RSNA key management). When a cached PMKSA is not being used, inclusion of the PMKID (if derived) is optional. For MLO, the Key Data field shall include the MAC Address KDE set to the MLD MAC address of the Authenticator. The Key Data field need not be encrypted.
* **4-way handshake message 2** is an EAPOL-Key frame with the Key Type subfield equal to 1. The Key Data field shall contain an RSNE, may contain an RSNXE, and need not be encrypted. For MLO, the Key Data field shall include the MAC Address KDE set to the MLD MAC address of the supplicant.

An ESS Supplicant’s SME shall insert the RSNE it sent in its (Re)Association Request frame, and shall insert the RSNXE it sent in its (Re)Association Request frame if the RSNXE is present in the (Re)Association Request frame it sent. The RSNE and the RSNXE are included as transmitted in the Management frame. For MLO the non-AP MLD shall include a MLO Link KDE containing the LinkID and affiliated STA MAC address for each link included in the Association Request frame. On receipt of message 2, the Authenticator’s SME shall validate the selected security configuration against the RSNE received in the (Re)Association Request frame and shall validate the RSNXE included in message 2 against the RSNXE received in the (Re)Association Request frame from the Supplicant. For MLO, the Authenticator’s SME shall validate that the MAC address used as the Address 2 field of the MAC header of the MPDU containing message 2 matches an affiliated STA MAC address included in one of the MLO Link KDEs and that the LinkID and affiliated STA MAC address for each link matches the LinkID and affiliated STA MAC address included in the ML element received in the (Re)Association Request frame.

An IBSS Supplicant’s SME shall insert an RSNE containing a selected pairwise cipher suite. The Authenticator’s SME shall validate that the pairwise cipher suite selected is one of its configured cipher suites and that the group cipher suite and AKM are consistent.

* **4-way handshake message 3** is an EAPOL-Key frame with the Key Type subfield equal to 1. The Key Data field shall contain one or two RSNEs, and may contain an RSNXE. If a group cipher has been negotiated, this field shall also include a GTK. This field shall be encrypted if a GTK is included. For MLO, the Key Data field shall include the MAC Address KDE set to the MLD MAC address of the Authenticator. When the Authenticator is an AP MLD and the Supplicant is a non-AP MLD, this field shall include one MLO GTK for each setup link (see 35.3.5 (Multi-link (re)setup)).

An Authenticator’s SME shall insert the RSNE it sent in its Beacon or Probe Response frame, and shall insert the RSNXE it sent in its Beacon or Probe Response frame if the RSNXE is present in the Beacon or Probe Response frame it sent. When this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the PMKR1Name is added in the PMKID List field of the RSNE. For MLO, an Authenticator’s SME shall insert a MLO Link KDE that includes the LinkID, affiliated AP MAC address, RSNE, and RSNXE, if it was present, for each affiliated AP link that was advertised in the ML element included in Beacons, Probe Response, and ML Probe Response frames. The Supplicant’s SME shall validate the selected security configuration against the RSNE received in message 3 and shall validate the RSNXE included in message 3 against the RSNXE received in the Beacon or Probe Response frame from the Authenticator. For MLO, the Supplicant’s SME shall validate the security configuration for each LinkID, affiliated AP MAC address, RSNE, and RSNXE for each affiliated AP link included in message 3 against the affiliated AP MAC address, RSNE, and RSNXE received for each link in Beacons, Probe Response, and ML Probe Response frames. For MLO, the Authenticator’s SME validates that the Address 2 field of the MAC header of the MPDU containing message message 3 matches an affiliated AP MAC address included in one of the MLO Link KDEs. If the second optional RSNE is present, the STA shall either use that cipher suite with its pairwise key or deauthenticate. In any of these cases, if the values do not match, then the receiver shall consider the RSNE or the RSNXE modified and shall use the MLME-DEAUTHENTICATE.request primitive to break the association. A security error should be logged at this time.

It may happen, for example, that a Supplicant selects a pairwise cipher suite which is advertised by an AP, but which policy disallows for this particular STA. An Authenticator may, therefore, insert a second RSNE to overrule the STA’s selection. An Authenticator’s SME shall insert the second RSNE, after the first RSNE, only for this purpose. The pairwise cipher suite in the second RSNE included shall be one of the ciphers advertised by the Authenticator. All other fields in the second RSNE shall be identical to the first RSNE.

A GTK shall be included and the unencrypted length of the GTK is six less than the length of the GTK KDE in octets. The entire Key Data field shall be encrypted as specified by the Key Descriptor Version.

* **4-way handshake message 4** is an EAPOL-Key frame with the Key Type subfield equal to 1. The Key Data field can be empty. For MLO, the Key Data field shall include the MAC Address KDE set to the MLD MAC address of the Supplicant

***Replace figure 12-47b with the following figure:***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Key ID | BIPN | Reserved | LinkID | BIGTK |
| Bits: | 16 | 48 | 4 | 4 | (Length – 13) × 8 |

***Insert the following figure and paragraphs at the end of the subclause:***

The format of the MLO Link KDE is shown in Figure 12–47c

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Link Information | MAC Address | RSNE | RSNXE |
| Octets: | 1 | 6 | variable | variable |

**Figure 12–47c MLO Link KDE format**

The Link Information field contains information identifying the presence of fields in the MLO Link KDE.

|  |  |  |  |
| --- | --- | --- | --- |
|  | LinkID | RSNXEInfo | Reserved |
| bits: | 4 | 1 | 3 |

**Figure 12–47c Link Information field**

The LinkID field contains the link identifier for the affiliated STA link.

The RSNXEInfo field indicates that the RSNXE is present in the MLO Link KDE when its value is set to 1.

The MAC Address field contains the MAC Address of the affiliated STA for the link specified in the Link Information field.

The RSNE field contains the RSNE of the affiliated STA for the link specified in the Link Information field. The RSNE is described in 9.4.2.24 (RSNE).

The RSNXE field contains the RSNE of the affiliated STA for the link specified in the Link Information field. The RSNXE is described in 9.4.2.241 (RSNXE).

***Modify the following clauses as follows:***

* + 1. EAPOL-Key frame notation

The following notation is used throughout the remainder of **Error! Reference source not found.** and 13.4 (FT initial mobility domain association) to represent EAPOL-Key frames:

(#59)(#1365)(#2456)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.

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(11ai).

A means a response is required to this message. This is used when the receiver should respond to this message. This is the Key Ack bit of the Key Information field.

I is the Install bit: (#4214)indicates whether to install (1) or not install (0) for the pairwise key. This is the Install bit of the Key Information field.

K is the key type: P (Pairwise), G (Group(#1504)). This is the Key Type bit of the Key Information field.

Reserved reserved.(#59)

KeyRSC is the key RSC. This is the Key RSC field.

ANonce/SNonce is the (#4214)Authenticator or Supplicant nonce, respectively. This is the Key Nonce field.

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, this Key MIC field is not included regardless of the MIC parameter value(11ai).

{Key Data} (#1365)(#2327)is a sequence of zero or more elements and KDEs, (#2457)concatenated and contained in the Key Data field, where(#2451):

 RSNE is described in 9.4.2.24 (RSNE).

 RSNE[KeyName] is the RSNE, with the (#2205)PMKID List field set to KeyName.

 GTK[N] is the GTK, with the key identifier field set to N. The key identifier specifies which index is used for this GTK. Index 0 shall not be used for GTKs, except in mixed environments, as described in **Error! Reference source not found.**.

 MAC\_Address is the MLD MAC address of the MLD with which the transmitting STA is affiliated.

 FTE is the Fast BSS Transition element, described in 9.4.2.47 (Fast BSS Transition element (FTE))

 MDE is the Mobility Domain element, described in 9.4.2.46 (Mobility Domain element (MDE))

 TIE[IntervalType] is a Timeout Interval element of type IntervalType, as described in 9.4.2.48 (Timeout Interval element (TIE)), containing e.g., for type KeyLifetime, the lifetime of the FT key hierarchy.

 IGTK[M] is the IGTK, with key identifier field set to M.

 IPN is the current IGTK replay counter value provided by the IGTK KDE

 BIGTK[Q] is the BIGTK, with key identifier field set to Q(#2116)

 BIPN is the current BIGTK replay counter value provided by the BIGTK KDE(#2116)

 PMKID is of type PMKID KDE and is the key identifier used during 4-way PTK handshake for PMK identification.(#59)

 OCI KDE is a KDE containing operating channel information(#4280)(M58).

 RSNXE is described in 9.4.2.241 (RSN Extension element (RSNXE)(#2715)).(#2715)

 PMKID identifies the PMKSA selected by the Authenticator(#2541).

 MLO GTK is the GTK for the AP affiliated with the AP MLD for the link specified by the value in the LinkID field.

 MLO IGTK is the IGTK for the AP affiliated with the AP MLD for the link specified by the value in the LinkID field.

 MLO BIGTK is the BIGTK for the AP affiliated with the AP MLD for the link specified in the LinkID field.

 MLO Link is the MAC Address, RSNE and the RSNXE, if advertised, for the STA affiliated with the MLD specified by the value of the LinkID field.

 “{a} or {b}” means that exactly one of either {a} or {b} is present as the {Key Data}(#2541).

“a n” means that the KDE could occur multiple times in the field for *n* links

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,{}, {PMKID} or {MAC Address }) (#1365)

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} or {RSNE, OCI KDE, RSNXE, MAC Address, MLO Linkn} or { RSNE, OCI KDE, RSNXE, MAC Address KDE, MLO Linkn, OCI KDE}) (M58)(#1365)(#2715)

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} or {MAC Address, MLO Linkn, MLO GTKn, MLO IGTKn, MLO BIGTKn} or {OCI KDE, MAC Address, MLO Linkn, MLO GTKn, MLO IGTKn, MLO BIGTKn}) (M58)(#1365)(#2715)

Message 4: Supplicant ® Authenticator: EAPOL-Key(1,1,0,0,P,0,0,0,MIC,{} or {MAC Address})(#1365).

The FT initial mobility domain association uses the FT 4-way handshake to establish an initial PTKSA, GTKSA, if management frame protection is enabled, an IGTKSA, and if beacon protection is enabled, a BIGTKSA,(#2116) that is based on this protocol. The FT 4-way handshake protocol is described in 13.4 (FT initial mobility domain association).

(#2454)The following apply:

* EAPOL-Key(´) denotes an EAPOL-Key frame conveying the specified argument list, using the notation introduced in **Error! Reference source not found.**.
* ANonce is a nonce that the Authenticator contributes for PTK generation(#59). ANonce has the same value in message 1 and message 3.
* SNonce is a nonce from the Supplicant for PTK generation(#59).
* P means the pairwise bit is set.
* The MIC is computed over the body of the EAPOL-Key frame (with the Key MIC field first zeroed before the computation) using the KCK defined in **Error! Reference source not found.** for PTK generation(#59).
* RSNE represents the appropriate RSNEs. For MLO, the RSNE is present in the MLO Link KDE.
* GTK[N] represents the GTK with its key identifier.(#59)
* OCI KDE contains the current operating channel information for the operating channel in which the EAPOL-Key frame(#4796) is sent. OCI KDE is present when dot11RSNAOperatingChannelValidationActivated is true on the Supplicant in Message 2 and Authenticator in Message 3. Otherwise it is absent(M58).
* (#2715)RSNXE, when included in message 2, contains the RSNXE that the Supplicant sent in its (Re)Association Request frame, and when included in message 3, contains the RSNXE that the Authenticator sent in its Beacon or Probe Response frame. RSNXE is present in message 2 if this element is present in the (Re)Association Request frame that the Supplicant sent, and is present in message 3 if this element is present in the Beacon or Probe Response frame that the Authenticator sent. For MLO, the RSNXE is present in the MLO Link KDE.
* For MLO, each message of the 4-way handshake contains an MAC Address KDE containing the MLD MAC address of the Authenticator or Supplicant that is sending the message.
* An MLO Link KDE is included for each affiliated STA link of an MLD. When included in message 2, an MLO Link KDE is included for each link and contains the LinkId and corresponding affiliated STA MAC address received in the ML element by the AP MLD in the (Re)Association request frame. When included in message 3, an MLO Link KDE is included for each affiliated AP link and contains the LinkId, corresponding affiliated AP MAC Address, RSNE, and RSNXE for each affiliated AP that was sent by the Authenticator in Beacons, Probe Response, and ML Probe Response frames.

NOTE—While the MIC calculation is the same in each direction, the Key Ack bit is different in each direction. It is set in EAPOL-Key frames from the Authenticator and 0 in EAPOL-Key frames from the Supplicant. 4‑way handshake requests from the Supplicant have the Request bit equal to 1. The Authenticator and Supplicant need to check these bits to stop reflection attacks. It is important that message 1 contents not be used to update state, in particular the keys in use, until the data are validated with message 3.

(M58)If dot11RSNAOperatingChannelValidationActivated is true and a channel switch is requested while the handshake is in progress, the handshake should be aborted.

* 4-way handshake message 1

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

Descriptor Type **=** N – see 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)

(#59)Reserved = 0

Install = 0

Key Ack = 1

Key MIC = 0

Secure = 0

Error = 0

Request = 0

Encrypted Key Data = 0

Reserved = 0 – unused by this protocol version

Key Length = Cipher-suite dependent(#1408); see **Error! Reference source not found.**

Key Replay Counter = *n* – to allow Authenticator or initiator STA to match the right message 2 from Supplicant or peer STA

Key Nonce = ANonce

EAPOL-Key IV = 0

Key RSC = 0

Key MIC = 0

Key Data Length = length of Key Data field in octets

Key Data =

* PMKID for the PMK being used during PTK generation
* For, MLO, a MAC Address KDE containing the MLD MAC address of the Authenticator.

Processing for PTK generation is as follows:

The Authenticator sends message 1 to the Supplicant at the end of a successful IEEE 802.1X authentication, after (re)association completes for a STA that has authenticated with SAE or PSK authentication is negotiated, when a cached PMKSA is used, or after a STA requests a new key. On reception of message 1, the Supplicant determines whether the Key Replay Counter field value has been used before with the current PMKSA. If the Key Replay Counter field value is less than or equal to the current local value, the Supplicant discards the message. Otherwise, the Supplicant:

* Generates a new nonce SNonce.
* Derives PTK.
* Constructs message 2.(#59)
* 4-way handshake message 2

Message 2 uses the following values for each of the EAPOL-Key frame fields:

Descriptor Type **=** N – see 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 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

(#59)Reserved = 0

Install = 0

Key Ack = 0

Key MIC = 0 when using an AEAD cipher or 1 otherwise(11ai)

Secure = 0 – same as message 1

Error = 0 – same as message 1

Request = 0 – same as message 1

(#1538)Encrypted Key Data = 1 when using an AEAD cipher or 0 otherwise

Reserved = 0 – unused by this protocol version

Key Length = 0

Key Replay Counter = *n* – to let the Authenticator or initiator STA know to which message 1 this corresponds

Key Nonce = SNonce

EAPOL-Key IV = 0

Key RSC = 0

Key MIC = Not present when using an AEAD cipher; otherwise(11ai), MIC(KCK, EAPOL) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

Key Data Length = length of Key Data field in octets

* Key Data =
* included RSNE – the sending STA’s RSNE for PTK generation or peer RSNE for the current operating band, and when this message 2 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the PMKR1Name calculated by the S1KH according to the procedures of **Error! Reference source not found.** is included in the (#2205)PMKID List field of the RSNE and the FTE and MDE are also included, or;
* The sending STA’s Multi-band element for PTK generation for a supported band other than the current operating band if dot11MultibandImplemented is true, or;
* The sending STA’s RSNE and Multi-band element(s) for generating a single PTK for all involved bands, if dot11MultibandImplemented is true and both the Authenticator and the Supplicant use the same MAC address in the current operating band and the other supported band(s); or;
* The sending STA’s RSNE and Multi-band element(s) for generating a different PTK for each involved band, if dot11MultibandImplemented is true and the Joint Multi-band RSNA subfield of the RSN capabilities field is 1 for both the Authenticator and the Supplicant, and either the Authenticator or the Supplicant uses different MAC addresses for different bands.(#59)
* (#2329)Additionally, contains an OCI KDE when dot11RSNAOperatingChannelValidationActivated is true(#2439) on the Supplicant.(M58)
* (#2715)The RSNXE that the Supplicant sent in its (Re)Association Request frame, if this element is present in the (Re)Association Request frame that the Supplicant sent.
* For MLO, a MAC Address KDE containing the MLD MAC Address of the Supplicant.
* For MLO, an MLO Link KDE for each affiliated STA link containing the affiliated STA MAC address included by the non-AP MLD in the Multi-Link element in (Re)Association Request frame.

Processing for PTK generation is as follows:

The Supplicant sends message 2 to the Authenticator.

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.

(M58)If dot11RSNAOperatingChannelValidationActivated is true and Supplicant RSNE indicates OCVC capability, the Authenticator silently discards message 2 if any of the following are true:

* OCI KDE or FTE OCI subelement is missing in the message
* Channel information in the OCI does not match current operating channel parameters (see **Error! Reference source not found.**)

Otherwise, the Authenticator:

* Derives PTK.
* Verifies the message 2 MIC or AEAD decryption operation result(11ai).
* If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or the AEAD decryption operation returns failure(11ai), the Authenticator silently discards message 2.
* If the MIC or AEAD decryption(11ai) is valid and this message 2 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the Authenticator checks that all fields of the RSNE other than the (#2205)PMKID List field (#2715)and, if present, the RSNXE bitwise matches the fields from the (Re)Association Request frame and that the FTE and MDE are the same as those provided in the AP’s (Re)Association Response frame. If the MIC or AEAD decryption(11ai) is valid and this message 2 is not part of a fast BSS transition initial mobility domain association and this message 2 is not part of an association started through the FT protocol, the Authenticator checks that the RSNE(#2715) and, if present, the RSNXE bitwise matches that from the (Re)Association Request frame. For MLO, validates that the affiliated STA MAC addresses are the same for each affiliated STA MAC address included in the ML element in the (Re)Association Request frame.
* If these are not exactly the same, the Authenticator uses MLME-DEAUTHENTICATE.request primitive to terminate the association.
* If they do match bitwise, the Authenticator constructs message 3.
* If management frame protection is being negotiated, the AP initializes the SA Query Transaction Identifier to an implementation-specific non-negative integer value, valid for the current pairwise security association.(#59)
* 4-way handshake message 3

Message 3 uses the following values for each of the EAPOL-Key frame fields:

Descriptor Type **=** N – see 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 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

(#59)Reserved = 0

Install = 0/1 – For PTK generation, 0 only if the AP does not support key mapping keys, or if the STA has the No Pairwise bit (in the RSN Capabilities field) equal to 1and only the group key is used.(#59)

Key Ack = 1

Key MIC = 0 when using an AEAD cipher or 1 otherwise(11ai)

Secure = 1 (keys installed)

Error = 0 – same as message 1

Request = 0 – same as message 1

Encrypted Key Data = 1

Reserved = 0 – unused by this protocol version

Key Length = Cipher-suite dependent(#1408); see **Error! Reference source not found.**

Key Replay Counter = *n+1*

Key Nonce = ANonce – same as message 1

EAPOL-Key IV = 0 (Version 2) or random (Version 1)

Key RSC = For PTK generation, starting TSC or PN that the Authenticator’s STA uses in MPDUs protected by GTK.(#59)

Key MIC = Not present when using an(Ed) AEAD cipher; or otherwise(11ai), 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

(#2440)Key Data Length = length of Key Data field in octets

Key Data =

* For PTK generation for the current operating band, the AP’s Beacon/Probe Response frame’s RSNE for the current operating band, and, optionally, a second RSNE that is the Authenticator’s pairwise cipher suite assignment for the current operating band, and, if a group cipher has been negotiated, the GTK and the GTK’s key identifier (see EAPOL-Key frames) for the current operating band, and if management frame protection is negotiated, the IGTK KDE, and if beacon protection is enabled, the BIGTK KDE(#2116), and when this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the PMKR1Name calculated according to the procedures of **Error! Reference source not found.** in the (#2205)PMKID List field of the RSNE and the FTE with the same contents as in the (Re)Association Response frame, the MDE with the same contents as in the (Re)Association Response frame, the reassociation deadline timeout set to the minimum of dot11FTReassociationDeadline and the key lifetime in the TIE[ReassociationDeadline], and the PTK lifetime in the TIE[KeyLifetime]; or
* For MLO, the MLO GTK KDE for each setup link (see 35.3.5.1 (Multi-link (re)setup procedure)). If management frame protection is negotiated, the MLO IGTK KDE for each setup link. If beacon protection is enabled, the MLO BIGTK KDE for each setup link. When this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the PMKR1Name calculated according to the procedures of 12.7.1.6.4 (PMK-R1) in the PMKID List field of the RSNE and the FTE with the same contents as in the (Re)Association Response frame, the MDE with the same contents as in the (Re)Association Response frame, the reassociation deadline timeout set to the minimum of dot11FTReassociationDeadline and the key lifetime in the TIE[ReassociationDeadline], and the PTK lifetime in the TIE[KeyLifetime]
* For PTK generation for a supported band other than the current operating band, the Authenticator’s Beacon/DMG Beacon/Announce/Probe Response/Information Response frame’s Multi-band element associated with the supported band, and optionally a second Multi-band element that indicates the Authenticator’s pairwise cipher suite assignment for the supported band, and, if group cipher for the supported band is negotiated, the Multi-band GTK KDE for the supported band if dot11MultibandImplemented is true, or;
* For generating a single PTK for all involved bands, the Authenticator’s Beacon/DMG Beacon/Announce/Probe Response/Information Response frame’s RSNE and Multi-band element(s), and optionally, additional RSNE and Multi-band element(s) that indicate the Authenticator’s assignment of one pairwise cipher suite for all involved bands; if a group cipher for all involved bands is negotiated, the GTK and the GTK’s key identifier for all involved bands, if dot11MultibandImplemented is true and both the Authenticator and the Supplicant use the same MAC address in the current operating band and the other supported band(s), or;
* For generating different PTKs for the current operating band and other supported band(s), the Authenticator’s Beacon/DMG Beacon/Announce/Probe Response/Information Response frame’s RSNE and Multi-band element(s), and optionally, additional RSNE and Multi-band elements that are the Authenticator’s pairwise cipher suite assignments for one or more involved bands; if group ciphers for the involved bands are negotiated, the Multi-band GTK KDEs for the involved bands, if dot11MultibandImplemented is true and the Joint Multi-band RSNA subfield is 1 for both the Authenticator and Supplicant, and either the Authenticator or the Supplicant uses different MAC addresses for different bands.(#59)
* (#2329)Additionally, contains an OCI KDE when dot11RSNAOperatingChannelValidationActivated is true on the Authenticator.(M58)
* (#2715)The RSNXE that the Authenticator sent in its Beacon or Probe Response frame, if this element is present in the Beacon or Probe Response frame that the Authenticator sent.
* For MLO, a MAC Address KDE containing the MLD MAC Address of the Authenticator.
* For MLO, a MLO Link KDE containing the LinkId, the affiliated AP MAC address, RSNE, and RSNXE for each affiliated AP that was sent by the Authenticator in Beacons, Probe Response, and ML Probe Response frames

Processing for PTK generation is as follows:

If the Extended Key ID for Individually Addressed Frames subfield of the RSN Capabilities field is 1 for both the Authenticator and the Supplicant, then the Authenticator assigns a new Key ID for the PTKSA in the range 0 to 1 that is different from the Key ID assigned in the previous handshake and uses the MLME-SETKEYS.request primitive to install the new key to receive individually addressed MPDUs protected by the PTK with the assigned Key ID. Otherwise Key ID 0 is used and installation of the key is deferred until after message 4 has been received. The Authenticator sends message 3 to the Supplicant.

NOTE—If an existing PTK is still in effect, the Authenticator IEEE 802.11 MAC continues to transmit protected, individually addressed MPDUs (if any) using the existing key. With the installation of the new key for receive, the Authenticator is able to receive protected, individually addressed MPDUs using either the old key (if present) or the new key.

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.

(M58)If dot11RSNAOperatingChannelValidationActivated is true and Authenticator RSNE indicates OCVC capability, the Supplicant silently discards message 3 if any of the following are true:

* OCI KDE or FTE OCI subelement is missing in the message
* Channel information in the OCI does not match current operating channel parameters (see **Error! Reference source not found.**)

The Supplicant also:

* Verifies the RSNE(#2715) and, if present, the RSNXE. If this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the Supplicant verifies that the PMKR1Name in the (#2205)PMKID List field of the RSNE is identical to the value it sent in message 2 and verifies that all other fields of the RSNE are identical to the fields in the RSNE present in the Beacon or Probe Response frames and verifies that the FTE and MDE are the same as in the (Re)Association Response frame. Otherwise, the Supplicant verifies that the RSNE is identical to that the STA received in the Beacon or Probe Response frame. (#2715)If the RSNXE is present, the Supplicant verifies that the RSNXE is identical to that the STA received in the Beacon or Probe Response frame. If any of these verification steps indicates a mismatch, 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.
* For MLO, verifies that the affiliated AP MAC address, the RSNE, and the RSNXE, if present, are the same as advertised by the affiliated APs of the AP MLD in Beacon, Probe Response, and ML Probe Response frames.
* Verifies the message 3 MIC or AEAD decryption operation result(11ai). If the calculated MIC does not match the MIC that the Authenticator included in the EAPOL-Key frame or AEAD decryption operation returns failure(11ai), the Supplicant silently discards message 3.
* Updates the last-seen value of the Key Replay Counter field.
* If the Extended Key ID for Individually Addressed Frames subfield of the RSN Capabilities field is 1 for both the Authenticator and Supplicant: Uses the MLME-SETKEYS.request primitive to configure the IEEE 802.11 MAC to receive individually addressed MPDUs protected by the PTK with the assigned Key ID.
* Constructs message 4.
* Sends message 4 to the Authenticator.
* Uses the MLME-SETKEYS.request primitive to configure the IEEE 802.11 MAC to send and, if the receive key has not yet been installed, to receive individually addressed MPDUs protected by the PTK. The GTK is also configured by MLME-SETKEYS primitive.(#59)
* 4-way handshake message 4

Message 4 uses the following values for each of the EAPOL-Key frame fields:

Descriptor Type **=** N – see 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 – same as message 1

Key Type = 1 (Pairwise) – same as message 1

(#59)Reserved = 0

Install = 0

Key Ack = 0 – this is the last message

Key MIC = 0 when using an(Ed) AEAD cipher or 1 otherwise(11ai)

Secure = 1

Error = 0

Request = 0

(#1538)Encrypted Key Data = 1 when using an AEAD cipher or 0 otherwise

Reserved = 0 – unused by this protocol version

Key Length = 0

Key Replay Counter = *n+1*

Key Nonce = 0

EAPOL-Key IV = 0

Key RSC = 0

Key MIC = Not present when using an AEAD cipher; or otherwise(11ai), MIC(KCK, EAPOL)(#59) – MIC computed over the body of this EAPOL-Key frame with the Key MIC field first initialized to 0

Key Data Length = length of Key Data field in octets

Key Data = For MLO, a MLO Link KDE containing the MLD MAC address of the Supplicant, otherwise there is no Key Data.

Processing for PTK generation is as follows:

The Supplicant sends message 4 to the Authenticator. Note that when the 4-way handshake is first used, message 4 is sent in the clear.

(#1322)On reception of message 4, the Authenticator verifies that the (#2390)Key Replay Counter field value is one that it used on this 4-way handshake and is strictly larger than that in any other EAPOL-Key frame that has the Request bit in the Key Information field set to 0 and that has been received during this session; if it is not, it silently discards the message. Otherwise:

* The Authenticator checks the MIC or AEAD decryption operation result(11ai). If the calculated MIC does not match the MIC that the Supplicant included in the EAPOL-Key frame or AEAD decryption operation returns failure(11ai), the Authenticator silently discards message 4.
* If the MIC is valid, the Authenticator uses the MLME-SETKEYS.request primitive to configure the IEEE 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.
* The Authenticator updates the Key Replay Counter field so that it uses a fresh value if a rekey becomes necessary.(#59)