IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Extending MLO to 11s mesh | | | | |
| Date: January 10, 2025 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Abhishek Patil | Qualcomm Technologies Inc. |  |  | appatil@qti.qualcomm.com |
| Gaurang Naik |  |  |  |
| Duncan Ho |  |  |  |
| Jouni Malinen |  |  |  |
| George Cherian |  |  |  |
|  |  |  |  |  |

Abstract

This submission extends multi-link operation to mesh STA while considering various aspects such as mesh discovery, peering, addressing and security.

**Revisions:**

* Rev 0: Initial version of the document.

***TGm editor: Baseline for this document is REVme D7.0 and TGbe D7.0.***

Interpretation of a Motion to Adopt

A motion to approve this submission means that the editing instructions and any changed or added material are actioned in the TGm Draft. This introduction is not part of the adopted material.

***TGm Editor: Editing instructions preceded by “TGm Editor” are instructions to the TGm editor to modify existing material in the TGm draft. As a result of adopting the changes, the TGm editor will execute the instructions rather than copy them to the TGm Draft.***

*TGm editor, please insert the following new section after 35.3.26 as shown below:*

**35.3.27 Mesh procedures with MLO**

A mesh STA with dot11MultiLinkActivated equal to true shall be affiliated with a mesh MLD and shall follow the procedures described in clause 14 (MLME mesh procedures) with additional considerations as described in this subclause.

**35.3.27.1 Multi-link mesh discovery**

All mesh STAs affiliated with the same mesh MLD shall advertise the same mesh ID in the Mesh ID element (also see 14.3.2 (Mesh identifier)) and the same mesh profile in the Mesh Configuration element (also see 14.3.3 (Mesh profile).

A mesh STA affiliated with a mesh MLD shall include the Basic Multi-Link element in the Beacon and Probe Response frames that it transmits. A mesh STA affiliated with a mesh MLD shall include Reduced Neighbor Report element in the Beacon and Probe Response it transmits to provide information of the other mesh STA(s) affiliated with the same mesh MLD. The Link ID field in the Basic Multi-link element and the Reduced Neighbor Report element identify each link of the mesh MLD.

NOTE 1 – The link identifier is unique to a mesh STA affiliated with a mesh MLD and no two affiliated mesh STAs of the same mesh MLD have the same link ID.

NOTE 2 – Each mesh MLD in a neighborhood can have different number of affiliated mesh STAs. Furthermore, each mesh MLD independently assigns link IDs to each of its links.

A mesh STA with dot11MultiLinkActivated equal to true may transmit a multi-link probe request to a neighboring mesh STA that is affiliated with a mesh MLD to gather information of other mesh STA(s) affiliated with that mesh MLD.

**35.3.27.2 Multi-link mesh peering**

A mesh STA that is affiliated with a mesh MLD may perform mesh peering as described in 14.4 (Mesh peering management (MPM)) with a neighbor mesh STA that is not affiliated with a mesh MLD. In this scenario:

* The mesh STA shall not include the Basic Multi-link element in the Mesh Peering Open and Mesh Peering Confirm frames that it transmits, and the mesh peering is established as a single link mesh peering between the two mesh STAs.
* When communicating with the peer mesh STA (during and after mesh peering), the mesh STA shall set the TA field to the MLD MAC address of the mesh MLD with whom the mesh STA is affiliated with.

NOTE 1 – A mesh STA determines that a neighboring mesh STA is not affiliated with a mesh MLD based on absence of Basic Multi-Link element in the Beacon or Probe Response frames transmitted by the neighboring mesh STA.

Note 2 – Since route discovery (HWMP) is based on the MLD MAC Address, an affiliated mesh STA needs to use the MLD MAC address in a single link mesh peering since it’s mesh MLD can be simultaneously involved with multi-link and single link mesh peering depending on the type of neighboring mesh STA.

A mesh STA that is affiliated with a mesh MLD may perform mesh peering with a neighbor mesh STA that is affiliated with a mesh MLD. In this scenario, the mesh STA shall include the Basic Multi-Link element in the Mesh Peering Open frame that it transmits if it determines that the neighboring mesh STA's mesh MLD has at least one other affiliated mesh STA operating on the same channel as another STA of its own mesh MLD.

NOTE 3 – A mesh STA can identify the channel(s) where the neighboring mesh MLD's other STA(s) are operating by examining the Reduced Neighbor Report element in the Beacon or Probe Response frames sent by the affiliated mesh STA of the neighboring mesh MLD.

A mesh STA that is affiliated with a mesh MLD shall include Basic Multi-Link element in the Mesh Peering Confirm frame that it transmits only if it receives a Mesh Peering Open frame containing Basic Multi-Link element. Such peering is referred to as multi-link mesh peering between the two mesh MLDs.

NOTE 4 – Since multi-link mesh peering is established at the MLD level, once this peering is established between the two MLDs, no other mesh STA from either mesh MLD is allowed to transmit a Mesh Peering Open frame to the peer mesh STA of the other mesh MLD.

The AID assigned by a mesh STA during the establishment of multi-link mesh peering is given to the neighboring peer mesh MLD and is the same for all affiliated STAs of the neighboring mesh MLD that are part of the multi-link mesh peering.

Once multi-link mesh peering is established between two MLDs, each link in multi-link operations shall be identified by the link ID assigned by the MLD whose affiliated mesh STA transmitted the Mesh Peering Confirm frame. For instance, a mesh STA initiating TTLM negotiation will refer to links using the link ID provided by the peer (responding) MLD.

NOTE 5 – In a given neighborhood, it is possible that each mesh MLD has different number of affiliated mesh STAs. Furthermore, an MLD independently assigns link ID values to each of its affiliated mesh STAs (i.e., links). Therefore, it is possible that the same link between two peer mesh STAs is be referred by different link ID values.

NOTE 6 – The Local Link ID and Peer Link ID values in the Mesh Peering Management element during mesh peering are unrelated to the Link ID values in the Basic Multi-Link element and the Reduced Neighbor Report element.

**35.3.27.3** **Contents of Basic Multi-Link element transmitted by an affiliated mesh STA**

The Basic Multi-Link element in a Beacon frame or a Probe Response frame, which is not a multi-link probe response, transmitted by a mesh STA affiliated with a mesh MLD, shall not include per-STA profiles for other affiliated STAs of the mesh MLD. This exclusion applies unless a reported STA is advertising an (Extended) Channel Switch Announcement element, Mesh Channel Switch Parameters element, Max Channel Switch Time element, Quiet element, or Quiet Channel element in its Beacon or Probe Response frames. In such exceptional cases, the STA Profile field in the Link Info field of the Basic Multi-Link element shall only include the relevant element(s) from the above list.

The Basic Multi-Link element in the multi-link probe response frame transmitted by a mesh STA affiliated with a mesh MLD shall include a complete or partial per-STA profile for one or more requested affiliated STAs, based on the soliciting multi-link probe request.

The Basic Multi-Link element carried in the Mesh Peering Open and Mesh Peering Confirm frames shall include complete per-STA profile (with inheritance applied) for each of the other affiliated mesh STA(s) that are operating on the same channel(s) as each of the other STA(s) of the neighboring mesh MLD. In this scenario, the mesh peering is established as multi-link mesh peering between the two mesh MLDs and shall apply to all the links of the two mesh MLDs that are operating on overlapping channels.

A mesh STA shall set the Medium Synchronization Delay Information Present, the EML Capabilities Present, the AP MLD ID Present and the Extended MLD Capabilities And Operations Present subfields of the Presence Bitmap subfield of the Basic Multi-Link element to 0. When a mesh STA transmits the Basic Multi-Link element, the following fields are reserved:

* NSTR Link Pair Present field and NSTR Bitmap Size field of the STA Control field of the Per-STA Profile subelement.
* SRS Support field, Frequency Separation For STR/AP MLD Type field, AAR Support field, and the Link Reconfiguration Operation Support field of the MLD Capabilities And Operations field of the Common Info field.

If dot11MeshSecurityActivated is true and the mesh MLD shares a PMK with the candidate peer mesh MLD, then the per-STA profile for each of the affiliated mesh STA carried within the Basic Multi-Link, included in the Mesh Peering Open and Mesh Peering Confirm frames, shall also include Authenticated Mesh Peering Exchange element containing the GTKdata and the IGTKdata fields for the link corresponding to the reported mesh STA. The Selected Pairwise Cipher Suite, the Local Nonce and the Peer Nonce fields of the Authenticated Mesh Peering Exchange element carry the same value as the Authenticated Mesh Peering Exchange element carried in the frame outside the Basic Multi-Link element.

**35.3.27.4 Multi-link mesh procedures**

A mesh MLD shall maintain a counter for each of its affiliated mesh STAs. The counter corresponding to an affiliated STA shall be incremented (modulo 256 excluding the value 255) by 1 when a critical update occurs to the operational parameters of that mesh STA as defined in 11.2.3.14 (TIM Broadcast). The counter value shall be advertised in the BSS Parameter Change Count subfield of the Common Info field of the Basic Multi-Link element for a transmitting mesh STA and in the BSS Parameter Change Count subfield of the Reduced Neighbor Report element for a reported mesh STA.

The Critical Update Flag subfield of the Capability Information field in Beacon and Probe Response frames is set to 1 until and including the next DTIM beacon on the link on which the reporting mesh STA is operating if there is a change to a value carried in the BSS Parameters Change Count subfield of the MLD Parameters field in the Reduced Neighbor Report element for any mesh STA affiliated with the same mesh MLD as the reporting mesh or a value carried in the BSS Parameters Change Count subfield in the Common Info field of the Basic Multi-Link element.

NOTE – The All Updates Included field in the Reduced Neighbor Report element corresponding to a reported mesh STA is set to 1 when the per-STA profile for that reported mesh STA is included (carrying the applicable element(s)) in the Basic Multi-Link element contained in the Beacon frame as described in 35.3.27.3 (Contents of Basic Multi-Link element transmitted by a mesh STA).

A mesh MLD shall follow the addressing scheme as described in 35.3.2 (MLD addressing).

A mesh MLD shall follow the block ack procedure as described in 35.3.8 (Block ack procedure in MLO).

A mesh MLD shall not announce (via Beacon or Probe Response frames) the removal of an affiliated mesh STA if the mesh STA is part of at least one mesh peering with another mesh STA or a mesh MLD. A mesh MLD shall not request addition or deletion of links to a multi-link mesh peering established with another mesh MLD.

A mesh MLD shall not announce (via Beacon or Probe Response frames) the disablement of a link if the link is part of at least one mesh peering with another mesh STA or a mesh MLD.

A mesh STA affiliated with a mesh MLD that intends to establish an SCS agreement with a peer mesh STA that is affiliated with a peer mesh MLD shall follow the procedures described in 35.17 (EHT SCS procedure). In such a scenario, the requesting mesh MLD shall take on the role of a non-AP MLD while the responding mesh MLD takes the role of an AP MLD. Since an SCS agreement can be initiated by either mesh peer, the SCSID space is shared between the peer MLDs such that each SCS agreement between the two mesh MLDs is uniquely identified.

**3.2 Definitions specific to IEEE 802.11**

*TGm editor, please insert the following definition (maintaining alphabetical order):*

**mesh multi-link device (MLD):** [mesh MLD] An MLD, where each station (STA) affiliated with the MLD is a mesh STA.

* + - * 1. **Mesh Peering Open frame details**

*TGm editor, please insert the following new row to the table as shown below:*

**Table 9-547—Mesh Peering Open frame Action field format**

|  |  |  |
| --- | --- | --- |
| **Order** | **Information** | **Notes** |
| <Last assigned + 1> | Basic Multi-Link element | The Basic Multi-Link element is present if dot11MultiLinkActivated is true; otherwise, it is not present. |

* + - * 1. **Mesh Peering Confirm frame details**

*TGm editor, please insert the following new row to the table as shown below:*

**Table 9-548—Mesh Peering Confirm frame Action field format**

|  |  |  |
| --- | --- | --- |
| **Order** | **Information** | **Notes** |
| <Last assigned + 1> | Basic Multi-Link element | The Basic Multi-Link element is present if dot11MultiLinkActivated is true; otherwise, it is not present. |

* **Mesh STA configuration**

*TGm editor, please update the contents of this subclause as shown below:*

The mesh STA configuration is a set of parameters that determines if two mesh STAs can communicate. The mesh STA configuration consists of the mesh profile (see 14.3.3 (Mesh profile)), the Supported Rates and BSS Membership Selectors element, the Extended Supported Rates and BSS Membership Selectors element, the HT Operations element (if present), the VHT Operations element (if present), the HE Operation element (if present), and the EHT Operation element (if present).

Mesh STA configurations are identical if the following conditions hold:

* The mesh profiles are identical.
* The BSSBasicRateSet parameter of the MLME-START.request primitive is identical to the basic rate set indicated by the Supported Rates and BSS Membership Selectors element and Extended Supported Rates and BSS Membership Selectors element, if present, received in the MLME-MESHPEERINGMANAGEMENT.indication primitive.
* For HT mesh STAs, the Basic HT-MCS Set field of the HT Operation parameter of the MLME-START.request primitive is identical to the HT Operation element received in the MLME-MESHPEERINGMANAGEMENT.indication primitive.
* For VHT mesh STAs, the Basic VHT-MCS and NSS Set field in the VHT Operation element of the MLME-START.request primitive is identical to the Basic VHT-MCS and NSS Set field in the VHT Operation element received in the MLME-MESHPEERINGMANAGEMENT.indication primitive.
* For HE mesh STAs, the Basic HE-MCS And NSS Set field in the HE Operation element of the MLME-START.request primitive is identical to the Basic HE-MCS And NSS Set field in the HE Operation element received in the MLME-MESHPEERINGMANAGEMENT.indication primitive.
* For EHT mesh STAs, the Basic EHT-MCS And NSS Set field in the EHT Operation element of the MLME-START.request primitive is identical to the Basic EHT-MCS And NSS Set field in the EHT Operation element received in the MLME-MESHPEERINGMANAGEMENT.indication primitive.
  + 1. **Block ack procedures in MLO**

*TGm editor, please update the contents of the 4th paragraph in this subclause as shown below:*

To setup a block ack agreement between two MLDs, an originator MLD shall send, through any affiliated STA operating on an enabled link, an ADDBA Request frame to the recipient MLD, subject to the power state of the non-AP STA or the peer mesh STA operating on that link. The ADDBA Request frame shall indicate the TID for which the block ack agreement is being set up. Upon receiving an ADDBA Request frame, the recipient MLD shall respond through any affiliated STA operating on an enabled link, with an ADDBA Response frame subject to the power state of the non-AP STA or the peer mesh STA operating on that link. The recipient MLD has the option of accepting or rejecting the request. If the recipient MLD accepts the request, then a block ack agreement is established between the originator MLD and the recipient MLD for the TID specified in the ADDBA frames as defined in 10.25.2 (Setup and modification of the block ack parameters).

**35.3.2 MLD addressing**

*TGm editor, please update the contents of the 3rd paragraph in this subclause as shown below:*

For an individually addressed frame sent on a link between two MLDs, the following apply:

* the value of the Address 2 (TA) field (if present) in the MAC header of the frame that is not a Probe Response frame shall be the MAC address of the transmitting STA affiliated with the MLD corresponding to that link except for the Individual/Group bit, which is set to 1 when the TA field value is a bandwidth signaling TA and set to 0 otherwise.
* if the frame is a Probe Response frame from an AP affiliated with the AP MLD operating on the link
  + if the AP does not belong to a multiple BSSID set or corresponds to the transmitted BSSID in a multiple BSSID set, then the value of the Address 2 (TA) field in the MAC header of the frame shall be set to the MAC address of the AP.
  + if the AP corresponds to the nontransmitted BSSID in a multiple BSSID set, then the value of the Address 2 (TA) field in the MAC header of the frame shall be set to the transmitted BSSID in the multiple BSSID set (see 11.1.4.3.4 (Criteria for sending a response)).
* if the frame is a Probe Response frame from a mesh STA affiliated with a mesh MLD operating on the link, then the value of the Address 2 (TA) field in the MAC header of the frame shall be set to the MAC address of the mesh STA.
* the value of the Address 1 (RA) field in the MAC header of the frame shall be the MAC address of the receiving STA affiliated with the MLD corresponding to that link.
* if the frame is a Management frame, the value of the Address 3 field in the MAC header of the Management frame shall be set based on 9.3.3.1 (Format of (PV0) Management frames).
* if the transmitting STA is a nonmesh STA and the frame is a Data frame, the value of the Address 3 field and the Address 4 field (if present) in the MAC header of the Data frame shall be set based on Table 9-60 (Address field contents for Data frames transmitted by nonmesh STAs) and the settings of the To DS and From DS bits in the MAC header of the Data frame (see 9.2.4.1.4 (To DS and From DS subfields)), where the BSSID is the MAC address of the AP affiliated with the AP MLD corresponding to that link.
* If the transmitting STA is a mesh STA and the frame is a Data frame or a Multihop Action frame, the value of the Address 3 and the Address 4 field in the MAC header of the frame shall be set as described in 9.3.5 (Frame addressing in an MBSS).

*TGm editor, please update the contents of the 4th paragraph in this subclause as shown below:*

For a frame sent by a nonmesh STA affiliated with the MLD with Address 1 field set to a group address (if allowed as described in 9.3.1 (Control frames), 9.3.2 (Data frames), and 9.3.3 ((PV0) Management frames)), the value of the Address 2 field, the Address 3 field (if present), and the Address 4 field (if present) in the MAC header of the frame shall be set as defined in 9.3.1 (Control frames), 9.3.2 (Data frames), and 9.3.3 ((PV0) Management frames), where the BSSID is the following:

* if the STA is an AP, then the BSSID is the MAC address of the AP
* if the STA is a non-AP STA affiliated with the non-AP MLD that has performed ML setup with an AP MLD, and a link is set up between the non-AP STA affiliated with the non-AP MLD and an AP affiliated with the AP MLD, then the BSSID is set to the MAC address of the AP affiliated with the AP MLD.

For a Control or a Management frame sent by a mesh STA affiliated with the MLD with Address 1 field set to a group address (if allowed as described in 9.3.1 (Control frames) and 9.3.3 ((PV0) Management frames)), the value of the Address 2 field, the Address 3 field (if present) in the MAC header of the frame shall be set as defined in 9.3.1 (Control frames) and 9.3.3 ((PV0) Management frames).

For a Data frame sent by a mesh STA affiliated with the MLD with Address 1 field set to group address, the value of Address 2 field, the Address 3 field and Address 4 field (if present) in the MAC header of the frame shall be set as described in 9.3.5 (Frame addressing in an MBSS).

**9.3.5 Frame addressing in an MBSS**

*TGm editor, please update the contents of 3rd, 4th, 5th and 6th paragraphs in this subclause as shown below:*

In individually addressed mesh Data and Multihop Action frames, Address 1 and Address 2 correspond to the mesh STA receiver address (RA) and the mesh STA transmitter address (TA) for a particular mesh link. Address 3 and Address 4 correspond to the destination mesh STA or the destination mesh MLD and the source mesh STA or the source mesh MLD of a mesh path respectively. The Address Extension Mode subfield in the Mesh Control field indicates the presence of an optional Mesh Address Extension subfield in the Mesh Control field. When the Extension Mode subfield equals “Address5&6” (see Table 9-35 (Valid values for the Address Extension Mode subfield)), the Mesh Control field includes Address 5 and Address 6 that correspond to the end-to-end destination address (DA) and source address (SA) of STAs that communicate over the mesh path, for instance, external STAs that communicate over the mesh BSS via proxy mesh gates (see Figure 9-136 (Example addressing for a mesh Data frame)).

NOTE 3—The forwarding of individually addressed mesh Data frames uses mesh STA addresses in fields Address 1 and Address 2. The forwarding of individually addressed mesh Data frames uses mesh STA address if the device is non-MLD or mesh MLD addresses if the device is an MLD in fields Address 3, and Address 4. This allows intermediate mesh STAs to forward mesh Data frames without necessarily having any knowledge of the addresses of the source and destination end stations or MLDs, which might be external addresses. Thus, proxy information only needs to be maintained by proxy mesh gates and by source mesh STAs.

The term source mesh STA or source mesh MLD refers to the first mesh STA or first mesh MLD on a mesh path. A source mesh STA or source mesh MLD is either a mesh STA or a mesh MLD that is the initial source of an MSDU/MMPDU or a mesh STA or a mesh MLD that receives an MSDU/MMPDU from a mesh path or from a STA outside the mesh BSS and translates and forwards the MSDU/MMPDU on the mesh path. The address of the source mesh STA or source mesh MLD is referred to as the Mesh SA.

The term destination mesh STA or destination mesh MLD refers to the final mesh STA or mesh MLD on a mesh path. A destination mesh STA or destination mesh MLD is either a mesh STA or mesh MLD that is the final destination of an MSDU/MMPDU or a mesh STA or a mesh MLD that receives an MSDU/MMPDU from a mesh path and translates and forwards the MSDU/MMPDU on another mesh path or to a STA outside of the mesh BSS. The address of the destination mesh STA or destination mesh MLD is referred to as the Mesh DA.

In group addressed mesh Data frames, Address 1 and Address 2 correspond to the group address and the mesh STA transmitter address (TA) respectively. Address 3 corresponds to the mesh source address (mesh SA) of the group addressed mesh Data frame. The Address Extension Mode indicates the presence of an optional address extension field Address 4 in the Mesh Control field that corresponds to the source address (SA) of external STAs that communicate over the mesh BSS via proxy mesh gates.

**9.4.2.110 RANN element**

*TGm editor, please update the following paragraph in this subclause as shown below:*

The Root Mesh STA Address field is set to the MAC address of the root mesh STA or the root mesh MLD.

**9.4.2.111 PREQ element**

*TGm editor, please insert the following note after the cited paragraph in this subclause as shown below:*

The Originator Mesh STA Address field is represented as a MAC address and is set to the originator MAC address.

NOTE – When the originator is a mesh MLD, the MAC address is the MLD MAC address of the originator mesh MLD.

*TGm editor, please insert the following note after the cited paragraph in this subclause as shown below:*

The Target Address field is set to a MAC address.

NOTE – The MAC address is the MLD MAC address if the target is a mesh MLD.

**9.4.2.112 PREP element**

*TGm editor, please update the following paragraph in this subclause as shown below:*

The Target Mesh STA Address is the MAC address of the target mesh STA or target mesh MLD or target proxy mesh gate and is represented as a MAC address.

*TGm editor, please insert the following note after the cited paragraph in this subclause as shown below:*

The Originator Mesh STA Address field is represented as a MAC address and is set to the MAC address of the originator, which is contained in the PREQ element.

NOTE – When the originator is a mesh MLD, the MAC address is the MLD MAC address of the originator mesh MLD.

**9.4.2.113 PERR element**

*TGm editor, please insert the following note after the cited paragraph in this subclause as shown below:*

The Destination Address field is represented as a MAC address and indicates the detected unreachable destination MAC address.

NOTE – The MAC address is the MLD MAC address if the destination is a mesh MLD.

**9.4.2.114 PXU element**

*TGm editor, please update the following paragraph in this subclause as shown below:*

The PXU Originator MAC Address field is represented as a MAC address and is the MAC address of the mesh STA or mesh MLD that originates this PXU element.

**9.4.2.115 PXUC element**

*TGm editor, please insert the following note after the cited paragraph in this subclause as shown below:*

The PXU Recipient MAC Address is represented as a MAC address and is set to the MAC address of the recipient of the PXU, i.e., the originator of the PXUC element.

NOTE – The MAC address is the MLD MAC address if the originator is a mesh MLD.

**9.4.2.116 Authenticated Mesh Peering Exchange element**

*TGm editor, please update the following paragraphs in this subclause as shown below:*

The GTKdata field is present when the Authenticated Mesh Peering Exchange element is carried within the Per-STA Profile subelement of Basic Multi-Link element; otherwise, it is optionally present. When present, it contains the bit string of {GTK || RSC || GTKExpirationTime} as the GTK data material. When present, the GTKdata field is protected by the exchange in which it is contained (see 14.6 (Authenticated mesh peering exchange (AMPE))). The RSC denotes the last TSC or PN sent using the GTK and is specified in Table 12-9 (RSC field) of 12.7.2 (EAPOL-Key frames). GTKExpirationTime denotes the key lifetime of the GTK in seconds and the format is specified in Figure 12-42 (Lifetime KDE format) of 12.7.2 (EAPOL-Key frames).

The IGTKdata field is present when the Authenticated Mesh Peering Exchange element is carried within the Per-STA Profile subelement of Basic Multi-Link element; otherwise, it is optionally present. When present, it contains the key ID, IPN and IGTK used with BIP for management frame protection. The format of the IGTKdata field is specified in Figure 12-44 (IGTK KDE format) of 12.7.2 (EAPOL-Key frames).

* **Mesh peering instance controller**
* **Overview**

*TGm editor, please update the 5th paragraph in this subclause as shown below:*

The localMAC is the MAC address of the mesh STA that is managing this mesh peering instance if the mesh STA is not affiliated with a mesh MLD; otherwise, the localMAC is the MLD MAC address of the mesh MLD that is managing the mesh peering instance. The peerMAC is the MAC address of the peer mesh STA or the candidate peer mesh STA if the peer mesh STA or the candidate peer mesh STA is not affiliated with a mesh MLD; otherwise, the peerMAC is the MLD MAC address of the peer mesh MLD or the candidate peer mesh MLD. The localLinkID is an integer generated by the mesh STA. The localLinkID shall be unique among all existing link identifiers used by the mesh STA or mesh MLD for its MPM finite state machines. The mesh STA or mesh MLD selects the localLinkID to provide high assurance that the same number has not been used to identify a recent MPM finite state machine. The peerLinkID is the localLinkID of the peer mesh STA or peer mesh MLD or candidate peer mesh STA or candidate peer mesh MLD and is supplied in the Mesh Peering Management element (see 9.4.2.100 (Mesh Peering Management element)) of the Mesh Peering Open and Mesh Peering Confirm frames.

* **Mesh GTKSA**

*TGm editor, please update the contents of this subclause as shown below:*

The mesh GTKSA results from a successful AMPE or mesh group key handshake and is unidirectional. In an MBSS, each mesh STA defines its own “transmit mesh GTKSA,” which is used to encrypt its group addressed transmissions. Also, each mesh STA stores a separate “receive mesh GTKSA” for each peer mesh STA so that encrypted group addressed traffic received from the peer mesh STAs may be decrypted.

A transmit mesh GTKSA is created by a mesh STA after the SME has changed the mesh GTK (MGTK) and the new MGTK has been sent to all peer mesh STAs. A receive mesh GTKSA is created by a mesh STA after successfully completing the AMPE in which a MGTK has been received (see 14.6.4 (Distribution of group keys in an MBSS)), or after receiving a valid message 1 of the mesh group key handshake. The receive mesh GTKSA shall be deleted when the lifetime expires or a new receive mesh GTKSA is created with the same key ID for the same MGTK source mesh STA. See 14.7.1 (General).

The MGTK and the GTK shall be independently selected from a uniform distribution. The MGTK source mesh STA MAC address in the mesh GTKSA shall not be the same as the Authenticator MAC address in the GTKSA.

Each mesh STA affiliated with a mesh MLD shall independently create its own transmit mesh GTKSA and maintain the received mesh GTKSA(s).

NOTE—The use of a distinct Transmit MGTK and ESS GTK with identical transmit MAC addresses is precluded by limitations on key rollover and reception by STAs in an infrastructure BSS (see 14.12.5 (Mesh STA collocation) for colocated mesh STA rules). If the distinct MGTKs were to use different key IDs, then rollover would be impossible. If the distinct MGTKs were to use the same key IDs, then STAs would incorrectly attempt to decrypt mesh broadcast traffic using the ESS GTK, causing error counters (such as dot11RSNAStatsCCMPDecryptErrors) to continuously increment.

The mesh GTKSA contains the following:

* MGTK
* MGTK source mesh STA MAC address (mesh STA that uses this GTK to encrypt transmissions)
* Group data cipher suite selector
* Direction vector (whether this is a receive mesh GTKSA or transmit mesh GTKSA)
* Key Index

**14.6 Authenticated mesh peering exchange (AMPE)**

* **Overview**

*TGm editor, please update the contents of this subclause as shown below:*

The authenticated mesh peering exchange (AMPE) establishes an authenticated mesh peering between mesh STAs or mesh MLDs that have established a mesh PMKSA before the initiation of the protocol. An authenticated mesh peering includes a mesh peering, corresponding mesh PTKSA, and the two mesh STAs’ mesh GTKSAs.

NOTE – Each mesh STA affiliated with a mesh MLD independently creates and maintains mesh GTKSAs (see 12.6.1.1.10 (Mesh GTKSA))

The AMPE is also used to establish an authenticated peering between two APs that support the AP PeerKey protocol (as defined in 12.10 (AP PeerKey support)) and that have established a PMK and PMKID before the initiation of the AMPE exchange.

The AMPE uses mesh peering Management frames. Parameters are exchanged via the RSNE, the Authenticated Mesh Peering Exchange element, and the MIC element.

The major functions provided by AMPE are security capabilities selection, key confirmation, and key management.

* The security capabilities selection function (specified in 14.6.2 (Security capabilities selection)) is performed by agreeing on the security parameters used for the protocol instance.
* Key confirmation using the shared PMK is performed by verifying that the protection on the mesh peering Management frame is correct.
* Key management (specified in 14.6.7 (Keys and key derivation algorithm for the authenticated mesh peering exchange (AMPE))) is performed by the derivation of the temporal key in the mesh PTKSA and the exchange of each mesh STA’s MGTK.

During the AMPE handshake, the mesh STAs generate nonces and transmit them via mesh peering Management frames. The mesh STA shall generate a random value for its localNonce, as specified in 12.7.5 (Nonce generation). The candidate peer mesh STA is expected to generate a random value for the peerNonce, which the mesh STA receives from the candidate peer mesh STA in Confirm and Close Action frames.

NOTE – The mesh STA that transmits the Mesh peering Management frame might be affiliated with a mesh MLD.

Mesh peering Management frames used in the AMPE are protected using the deterministic authenticated encryption mode of AES-SIV (IETF RFC 5297).

* **Security capabilities selection**
* **Instance Pairwise Cipher Suite selection**

*TGm editor, please update the contents of this subclause as shown below:*

Pairwise cipher suite selector TKIP shall not be used as the pairwise cipher suite when dot11MeshSecurityActivated, dot11ProtectedTXOPNegotiationActivated, or dot11ProtectedQLoadReportActivated is true.

If the pairwise cipher suite has not been selected, a STA or an MLD shall attempt to reach the agreement on the pairwise cipher suite using the following procedure in four steps:

* The STA or the MLD shall announce the list of pairwise cipher suites it supports using an ordered list in the RSNE in the Mesh Peering Open frame. The first value in the list is the STA’s or the MLD’s most preferred cipher suite, and the last value the least preferred.
* If the STA or the MLD receives a Mesh Peering Open frame from the candidate peer STA or the candidate peer MLD, the STA or the MLD shall make its decision on the selected pairwise cipher suite based on the intersection of its own ordered list and the received ordered list.
* If the intersection is empty, the pairwise cipher suite selection fails and the STA or the MLD generates the failure reason code MESH-INVALID-SECURITY-CAPABILITY and then takes the corresponding actions specified in 14.6.6 (AMPE finite state machine).
* If the intersection contains more than one value, the selected cipher suite shall be the entry in the intersection list most preferred by the STA or the MLD that has the largest MAC address in the lexicographic ordering.
* If the STA or MLD receives a Mesh Peering Confirm frame from the candidate peer STA or the candidate peer MLD before receiving a Mesh Peering Open frame, the STA or the MLD shall verify that it supports the pairwise cipher suite chosen by the candidate peer STA. Otherwise, the selection fails and the STA shall generate the failure reason code MESH-INVALID-SECURITY-CAPABILITY.   
  Furthermore, upon receiving a Mesh Peering Open frame, the STA or the MLD shall verify that the accepted selected pairwise cipher suite matches the pairwise cipher suite chosen in step b). If they do not match, the selection fails and the STA or the MLD shall generate the failure reason code MESH-INVALID-SECURITY-CAPABILITY. Otherwise, the pairwise cipher suite selection succeeds, and the STA or the MLD shall proceed to step d).
* If the STA or the MLD is generating a Mesh Peering Confirm frame, it shall set the Selected Pairwise Cipher Suite to the selected pairwise cipher suite upon successful pairwise cipher suite selection.
* **Construction and processing AES-SIV-protected mesh peering Management frames**

*TGm editor, please add a NOTE at the end of this subclause as shown below:*

AES-SIV performs deterministic authenticated encryption and takes additional data that is authenticated but not encrypted (AAD). When encrypting and authenticating, AES-SIV takes a key, plaintext data to protect, and multiple distinct components of AAD, to produce a synthetic initialization vector and a cipher text. When verifying encrypted and authenticated data AES-SIV takes a key, a synthetic initialization vector, cipher text data to decrypt and verify, and AAD, to produce either plaintext or the symbol “FAIL,” indicating failure to decrypt and verify. Note that the AAD used in the encryption process shall be identical to the AAD used in the decryption process and the synthetic initialization vector produced by the encryption process shall be used in the decryption process.

When the mesh STA constructs a mesh peering Management frame, it shall follow the following procedure:

* The input key shall be the AEK
* The input plaintext shall be the Authenticated Mesh Peering Exchange element (see 9.6.15.2 (Mesh Peering Open frame format), 9.6.15.3 (Mesh Peering Confirm frame format), and 9.6.15.4 (Mesh Peering Close frame format))
* The input AAD shall be three distinct components consisting of
* The localMAC
* The peerMAC
* The contents of the mesh peering Management frame from the category (inclusive) to the MIC element (exclusive)
* The output synthetic initialization vector shall be copied into the MIC field of the MIC element in the mesh peering Management frame
* The output cipher text shall become the remainder of the mesh peering management frame body after the MIC element

When the mesh STA verifies a mesh peering Management frame, it shall follow the following procedure:

* The input key shall be the AEK
* The input synthetic initialization vector shall be the MIC field of the MIC element in the mesh peering Management frame
* The input cipher text shall be the part of the mesh peering management frame body following the MIC element
* The input AAD shall be three distinct components consisting of
* The peerMAC
* The localMAC
* The contents of the mesh peering Management frame from the category (inclusive) to the MIC element (exclusive)
* If AES-SIV returns the symbol “FAIL” processing of the frame shall be deemed a failure with a behavior dependent on the type of mesh peering Management frame
* If AES-SIV returns plaintext it shall be treated as the components of the mesh peering Management frame and processed accordingly.

NOTE – The localMAC or the peerMAC is the MLD MAC address of the corresponding mesh MLD if the mesh STA is affiliated with a mesh MLD (see 14.4.4.1 (Overview)).

* **Distribution of group keys in an MBSS**

*TGm editor, please add the following NOTEs at the end of this subclause as shown below:*

The MGTK shall be a random or pseudorandom number. The mesh STA shall distribute the MGTK to the peer mesh STA using the Mesh Peering Open frame during the AMPE. Upon successful completion of AMPE, each mesh STA shall establish states for the peer mesh STA’s mesh GTKSA. The GTKdata subfield in the Authenticated Mesh Peering Exchange element shall contain the MGTK concatenated with the RSC and the GTKExpirationTime (as specified in 9.4.2.116 (Authenticated Mesh Peering Exchange element)).

When dot11RSNAProtectedManagementFramesActivated is true, a mesh STA shall distribute the IGTK to the peer mesh STA using the Mesh Peering Open frame during the AMPE. Upon successful completion of AMPE, each mesh STA shall establish an IGTKSA (see 12.6.1.1.9 (IGTKSA)) with the mesh peer. The IGTKdata subfield in the Authenticated Mesh Peering Exchange element shall contain the key ID concatenated with the IPN and the IGTK (as specified in 9.4.2.116 (Authenticated Mesh Peering Exchange element)).

NOTE 1 – Each mesh STA affiliated with a mesh MLD independently creates and maintains mesh GTKSAs (see 12.6.1.1.10 (Mesh GTKSA)).

NOTE 2 – During multi-link mesh peering between two mesh MLDs, the Authenticated Mesh Peering Exchange element contained within the Per STA Profile subelement of the Basic Multi-Link element carried within the Mesh Peering Open and the Mesh Peering Confirm frames includes the GTKdata and the IGTKdata fields for the link corresponding to the reported affiliated mesh STA (see 35.3.27.3 (Contents of Basic Multi-Link element transmitted by an affiliated mesh STA)).

* **Generating Mesh Peering Open frames for AMPE**

*TGm editor, please update the contents of this subclause as shown below:*

In addition to contents for establishing a mesh peering as specified in 14.4.6.1 (Generating Mesh Peering Open frames), the Mesh Peering Open frame, if used for the AMPE, shall contain the following:

* In the Mesh Peering Management element, the Mesh Peering Protocol Identifier shall be set to 1 “authenticated mesh peering exchange protocol.”
* In the Mesh Peering Management element, the Chosen PMK field shall be set to PMKID that identifies the mesh PMKSA the mesh STA established with the candidate peer mesh STA.
* The RSNE shall be identical to the RSNE in the STA’s Beacon and Probe Response frames.
* In the Authenticated Mesh Peering Exchange element:
* The Selected Pairwise Cipher Suite field shall be set to the first cipher suite selector in the Pairwise Cipher Suite List field in the RSNE.
* The Local Nonce field shall be set to the localNonce value generated by the mesh STA for identifying the current mesh peering instance.
* The Peer Nonce field shall be set to 0.
* If dot11MeshSecurityActivated is true, the GTKdata field shall be present and shall contain the data for the mesh STA’s MGTK. The GTKdata field shall not be present when AMPE is being used as part of the AP PeerKey protocol (12.10.2 (AP PeerKey protocol)). The components of the GTKdata field are specified in 14.6.4 (Distribution of group keys in an MBSS).

Also see 35.3.27.3 (Contents of Basic Multi-Link element transmitted by an affiliated mesh STA) for contents of the Authenticated Mesh Peering Exchange element when carried within Basic Multi-Link element during multi-link mesh peering between two mesh MLDs.

The Mesh Peering Open frame shall be protected using AES-SIV as specified in 14.6.3 (Construction and processing AES-SIV-protected mesh peering Management frames).

* **Processing Mesh Peering Open frames for AMPE**

*TGm editor, please update the contents of this subclause as shown below:*

On receiving a Mesh Peering Open frame, the mesh STA shall verify the received frame. If AES-SIV returns the symbol “FAIL” the OPN\_RJCT event shall be invoked to the corresponding AMPE finite state machine and the reason code “MESH-INVALID-GTK” is generated. Otherwise, processing continues.

The received frame shall be rejected if the security capability selection fails (see 14.6.2 (Security capabilities selection)). The OPN\_RJCT event shall be invoked to the corresponding AMPE finite state machine.

If dot11RSNAOperatingChannelValidationActivated is true and the received RSNE indicates OCVC, the mesh STA shall validate the OCI element in the Mesh Peering Open frame by ensuring that all of the following are true:

* OCI element is present
* Channel information in the OCI matches current operating channel parameters (see 12.2.10 (Requirements for Operating Channel Validation))

Otherwise, the mesh STA shall discard the frame.

The peer mesh STA’s MGTK extracted from the Mesh Peering Open frame shall be added to the Receive MGTK SA in which the peer’s MAC address equals the MGTK source mesh STA MAC address.

NOTE 1 – Each mesh STA affiliated with a mesh MLD independently creates and maintains mesh GTKSAs (see 12.6.1.1.10 (Mesh GTKSA)).

NOTE 2 – During multi-link mesh peering between two mesh MLDs, the Authenticated Mesh Peering Exchange element contained within the Per STA Profile subelement of the Basic Multi-Link element carried within the Mesh Peering Open and the Mesh Peering Confirm frames includes the GTKdata and the IGTKdata fields for the link corresponding to the reported affiliated mesh STA (see 35.3.27.3 (Contents of Basic Multi-Link element transmitted by an affiliated mesh STA)).

If all operations succeed, the mesh STA shall proceed to process the Mesh Peering Open frame on basic parameters as specified in 14.4.6.2 (Mesh Peering Open frame processing).

* **Mesh peering confirm for AMPE**
* **Generating Mesh Peering Confirm frames for AMPE**

*TGm editor, please update the contents of this subclause as shown below:*

In addition to contents for establishing a mesh peering as specified in 14.4.7.1 (Generating Mesh Peering Confirm frames), the Mesh Peering Confirm frame, when used with the AMPE, shall contain the following:

* In the Mesh Peering Management element, the Mesh Peering Protocol Identifier shall be set to 1 “authenticated mesh peering exchange protocol.”
* The RSNE shall be the same as sent in the Mesh Peering Open frame.
* In the Authenticated Mesh Peering Exchange element:
* The Selected Pairwise Cipher Suite field shall be set to the cipher suite selector that indicates the successfully selected pairwise cipher suite (specified in 14.6.2.1 (Instance Pairwise Cipher Suite selection)).
* The Peer Nonce field shall be set to the nonce value chosen by the peer mesh STA as received in the Local Nonce field in the Mesh Peering Open frame from the candidate peer mesh STA.
* The GTKdata field shall not be present.
* The IGTKdata field shall not be present.
* The rest of fields are set to the same values sent in the Mesh Peering Open frame.

Also see 35.3.27.3 (Contents of Basic Multi-Link element transmitted by an affiliated mesh STA) for contents of the Authenticated Mesh Peering Exchange element when carried within Basic Multi-Link element during multi-link mesh peering between two mesh MLDs.

The Mesh Peering Confirm frame shall be protected using AES-SIV as specified in 14.6.3 (Construction and processing AES-SIV-protected mesh peering Management frames).

* **Keys and key derivation algorithm for the authenticated mesh peering exchange (AMPE)**

*TGm editor, please update the contents of this subclause as shown below:*

To execute the AMPE and mesh group key handshake with a candidate peer STA or candidate peer MLD, the local STA or local MLD shall derive an authenticated encryption key (AEK) and a mesh temporal key (MTK) using the PMK it shares with the candidate peer STA or candidate peer MLD.

The AEK is derived statically from the shared PMK. The MTK is derived from the shared PMK and dynamic information provided by the STA or MLD and candidate peer STA or candidate peer MLD.

The AEK is mutually derived by the local STA or local MLD and the peer STA or peer MLD once a new PMK has been selected. The AEK shall be derived from the PMK by

AEK 🡨 KDF-*Hash*-256(PMK, “AEK Derivation”, Selected AKM Suite ||   
min(localMAC, peerMAC) || max(localMAC, peerMAC))

where

KDF-*Hash*-256 is the key derivation function defined in 12.7.1.6.2 (Key derivation function (KDF)) using the hash algorithm identified by the AKM suite selector (see Table 9-190 (AKM suite selectors))

Selected AKM Suite is a 4-octet string formed by concatenating the OUI or CID and suite type

NOTE – The localMAC or the peerMAC is the MLD MAC address of the corresponding mesh MLD if the mesh STA is affiliated with a mesh MLD (see 14.4.4.1 (Overview)).

The temporal key (MTK) shall be derived from the PMK by

MTK 🡨 KDF-*Hash*-*Length*(PMK, “Temporal Key Derivation”, min(localNonce, peerNonce) || max(localNonce, peerNonce) || min(localLinkID, peerLinkID) || max(localLinkID, peerLinkID) || Selected AKM Suite || min(localMAC, peerMAC) || max(localMAC, peerMAC))

where

KDF-*Hash*-*Length* is the key derivation function defined in 12.7.1.6.2 (Key derivation function (KDF)) using the hash algorithm identified by the AKM suite selector (see Table 9-190 (AKM suite selectors))

Length is cipher suite dependent and defined by the TK\_bits value in Table 12-8 (Cipher suite key lengths)

Selected AKM Suite is a 4-octet string formed by concatenating the OUI or CID and suite type

“min” and “max” operations for IEEE 802 addresses  
are with the address converted to a positive integer, treating the first transmitted octet as the most significant octet of the integer as specified in 12.7.1.3 (Pairwise key hierarchy)

“min” and “max” operations for nonces   
are encoded as specified in 9.2.2 (Conventions)

“min” and “max” operations for LinkIDs   
select the minimum and maximum, respectively, of the two unsigned integers (In the KDF context, LinkIDs are encoded as 16-bit unsigned integers, represented using the bit ordering conventions of 9.2.2 (Conventions).)

NOTE – The localMAC or the peerMAC is the MLD MAC address of the corresponding mesh MLD if the mesh STA is affiliated with a mesh MLD (see 14.4.4.1 (Overview)).

The MTK is used to protect communications between two peer STAs or peer MLDs. The local STA or local MLD and peer STA or peer MLD derive an MTK per peering instance and may rekey the MTK using AMPE.