IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11bi D2.0 CR for Miscellaneous CIDs | | | | |
| Date: 2025-09-09 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | Email |
| Po-Kai Huang | Intel |  |  | po-kai.huang@intel.com |

Abstract

This submission resolves the following CIDs:

2001, 2002, 2010, 2021, 2045, 2050, 2065, 2091, 2115, 2116,

2117, 2118, 2119, 2120, 2335, 2175, 2088, 2089, 2098, 2093,

2094, 2097, 2100, 2104, 2114, 2090, 2164, 2165, 2166, 2167,

2174, 2176, 2168, 2179, 2169, 2178, 2177, 2267, 2268, 2281,

2367, 2465, 2468, 2469, 2488, 2489, 2095, 2096

Revisions:

* Rev 0: Initial version of the document.
* Rev 1: Add CID 2177 and 2168 that are similar comments with the comments in revision 0. Add CID 2095 and CID 2096.
* Rev 2: Revision for CID 2175 and CID 2335.

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 TGbi D2.0 Draft. This introduction is not part of the adopted material.

Editing instructions formatted like this are intended to be copied into the TGbi D2.0 Draft. (i.e. they are instructions to the 802.11 editor on how to merge the text with the baseline documents). TGbi Editor: Editing instructions preceded by “TGbi Editor” are instructions to the TGbi editor to modify existing material in the TGbi draft. As a result of adopting the changes, the TGbi editor will execute the instructions rather than copy them to the TGbi Draft.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Clause** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 2001 |  | 0.00 | Agree to forward it to SA ballot |  | Rejected –  Thanks for participating in the comment collection. |
| 2002 | 12.16.5 | 152.02 | Add "by transmitting the Authentication frames to the AP" at the end of the sentence, to keep consistent with the paragraph after this one. | As in comment | Accepted - |
| 2010 | 12.16.2 | 147.08 | Pleaes clarify the relationship between EPP AP and BPE AP? Are they mutual exclusive? | Add the definition of EPP and clarify the necessarity for "that is not BPE AP". | Rejected –  EPP is defined in Enhanced Privacy Protection (EPP) enhancements, and BPE AP is defined in 3.2. Basically, it is AP that implements BPE features. |
| 2021 | 9.4.2.348 | 76.35 | "The DS MAC Address field indicates the DS MAC address." That does not read right. Copy similar text from elsewhere. | Replace with "The DS MAC Address contains a MAC address." | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2021 |
| 2045 | 12.16.6.2 | 157.01 | "NOTE 1--If the DS MAC Address element is included in the (Re)Association Request frame, the source address or destination address parameters of the MAC service tuples (see 5.2.4.2 (Semantics of the service primitive)) for the EPP non-AP STA are set to the DS MAC address, which is the identity of the non-AP STA known by the DS." Until I read this note I was struggling to understand how this DS MAC thingy worked (mind you I might still have it wrong as there is no simple text explaining the feature). I think this Note needs to be normative text. While here, it is a pity that TGbi did not build on TGbh as Device ID and'or IRM could have been used to identify the non-AP across the ESS instead of inventing something new. | Make cited Note normative. | Rejected –  The note follows similar style in 11be where similar descriptions are a note.  *NOTE—For MLDs, the source address or destination address parameters of the MAC service tuples (see 5.2.3.2 (Semantics of the service primitive)) are set to the MLD MAC address of the non-AP MLD, which is the identity of the non-AP MLD known by the DS.* |
| 2050 | 12.16.5 | 153.21 | The sentence describes a comparison, not an assignment. | Change the text: "Upon receiving the Authentication frame with Authentication Transaction Sequence Number field set to a value that is larger than or equal to 3" to "Upon receiving the Authentication frame with the Authentication Transaction Sequence Number field larger than or equal to 3" There is the same issue at P153.33. | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2050 |
| 2065 | 9.4.2.46 | 70.22 | "if the frame contains the FTE is encrypted..." this is not very clear. | "if the frame containing..." or please make it clearer, same for IGTK, BIGTK and WIGTK (same page) | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2065 |
| 2091 | 9.4.2.46 | 0.00 | This kind of addition is all over 9.4.2.46, "if the frame contains the FTE is not encrypted and contains the GTK being distributed if the frame contains the FTE is encrypted." That's not a proper English sentence. "...the frame containing the FTE..."? | Fix the English. Make a new sentence if you can't properly glom on a new condition to the existing sentence. | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2065 |
| 2115 | 9.4.2.46 | 70.22 | Missing words in sentence | Change this clause: "if the frame that contains the FTE is not encrypted and contains the GTK being distributed if the frame contains the FTE is encrypted" to "if the frame contains the FTE is not encrypted and alternatively contains the GTK being distributed if the frame that contains the FTE is encrypted". | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2116 | 9.4.2.46 | 70.42 | Missing words in sentence | Change this clause: "if the frame contains the FTE is not encrypted and contains the IGTK being distributed if the frame contains the FTE is encrypted." to "if the frame that contains the FTE is not encrypted and alternatively contains the IGTK being distributed if the frame that contains the FTE is encrypted.". | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2117 | 9.4.2.46 | 70.61 | Missing words in sentence | Change this clause: "if the frame contains the FTE is not encrypted and contains the BIGTK being distributed if the frame contains the FTE is encrypted." to "if the frame that contains the FTE is not encrypted and alternatively contains the BIGTK being distributed if the frame that contains the FTE is encrypted." | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2118 | 9.4.2.46 | 71.13 | Missing words in sentence | Change the clause: " if the frame contains the FTE is not encrypted and contains the WIGTK being distributed if the frame contains the FTE is encrypted." to " if the frame that contains the FTE is not encrypted and alternatively contains the WIGTK being distributed if the frame that contains the FTE is encrypted." | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2119 | 9.4.2.46 | 72.41 | Missing words in sentence | Change the clause: "if the frame contains the FTE is not encrypted and contains the PGTK being distributed if the frame contains the FTE is encrypted." to "if the frame that contains the FTE is not encrypted and alternatively contains the PGTK being distributed if the frame that contains the FTE is encrypted." | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2120 | 9.4.2.46 | 72.59 | Missing words in sentence | Change the clause: "if the frame contains the FTE is not encrypted and contains the identity key being distributed if the frame contains the FTE is encrypted." to "if the frame that contains the FTE is not encrypted and alternatively contains the identity key being distributed if the frame that contains the FTE is encrypted." | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2115 |
| 2335 | 12.2.7 | 130.21 | Presumably encryption is negotiated, not just used | Change "frames is used" to "frames is negotiated" | Revised –  Agree in principle with the commenter.  TGbi editor to make the changes shown in this document under all headings that include CID 2335 |
| 2175 | 12.16.8 | 162.03 | The 12.16.8 introduction clause talks about temporary key estavblishment, but the clause 12.16.8.2 creates complete PTK by using this scheme. This mismatch causes confusion. | Please, change the introduction clause to describe that TK is created for FT, while the 802.1X creates a complete PTK (TK, KEK, KDK, KCK). | Revised –  Agree in principle. We simply say to derive a PTK with TK.  Also, based on the feedback of the commenter, we add a MIC in the second Authentication frame for PMKSA caching.  TGbi editor to make the changes shown in this document under all headings that include CID 2175 |
| 2088 | 9.3.3.11 | 44.53 | To do 802.1X in Authentication frames the only fields necessary are the one to hold the EAPOL PDU and one for the AKM Suite Selector. There is no length needed because it can be inferred from the remainder of the frame. | Make an "EAPOL PDU" field after the AKM Suite Selector. And that's all that's needed. In table 9-71 for 802.1X authentication, none of that stuff is needed because it's not part of 802.1X authentication. There's no diffie-hellmans, no nonces, nothing. | Rejected –  Fields are before the elements rather than after the elements. DH element is need for PFS. Nonce and RSNE are needed for PTKSA generation if encryption of (Re)Association Request/Response is used.. |
| 2089 | 9.4.1.1 | 52.49 | Define a new authenticaiton algorithm for PMK caching and PTK derivation | Ask ANA to assign a separate authentication algorithm to do PMK caching and PTK derivation. Then address the comment in 12.16.8.2 about this new exchange. | Rejected –  In all the existing AKMs defined in Table 9-190—AKM suite selectors, PMKSA caching are part of the AKM rather than a separate AKM. |
| 2098 | 12.16.8.2 | 164.56 | PMK caching should be a separate auth algorithm. Do not overload the EAPOL auth with things like PTK derivation and PMK caching. | Make a separate authentication algorithm to perform PMK caching. | Rejected –  In all the existing AKMs defined in Table 9-190—AKM suite selectors, PMKSA caching are part of the AKM rather than a separate AKM. |
| 2093 | 12.7.1.3 | 140.59 | What's the reason for requiring a DHss here? | Get rid of this. A DH is not needed just because an EAPOL over Auth exchange was done. Let the PTK be derived rationally and securely by binding a transcript of the EAPOL exchange to the PMK. | Rejected –  DHss is required in EPPKE (the SAE version with (re)association request/response encryption) due to adoption from PASN and. As a result, we have all the modes to have PFS by having DH exchange. |
| 2094 | 12.16.5 | 151.01 | 802.1X over Authentication frames is not a "client privacy enhancement". | make the contents of 12.16.5 into 12.16 and make the rest of 12.16.X into 12.17.X "Client Privacy Enhancements" | Rejected –  802.1X over Authentication frame is the necessary mechanism to enable encryption of (re)association request/response, a privacy feature, when 802.1X is used. |
| 2097 | 12.16.8.2 | 164.07 | since association has not happened yet it is not clear what point a pairwise cipher is being checked here. The EAPOL exchange will include the AKM Suite Selector so that's taken care of, both sides agreed to the AKM and they both agreed to the EAP method and they authenticated. There is no need for a pairwise cipher at this point. | Get rid of the RSNE. It has not been used yet and there is no need to bind it to anything. | Rejected –  The RSNE is needed when the PTKSA derivation happens later. This is needed before the exchange of (Re)Association Request/response. If RSNE is removed, then we do not have necessary information to derive PTKSA to encrypt (Re)Association Request/Response. |
| 2100 | 9.3.3.11 | 45.08 | Nonce's Notes might make confusion. Nonce element is not mandatory element. | Nonce's Notes should be change as ff: "The Nonce element is present only in certain Authentication frames as defined in Table 9-71" | Accepted - |
| 2104 |  | 0.00 | Cover page needs to be updated | Change REVme/D7.0 to P802.11-2024 | Revised –  TGbi editor updates REVme/D7.0 to IEEE Std 802.11-2024, IEEE P802.11bh™/D6.0 to IEEE Std 802.11bh-2024, and IEEE P802.11be/D7.0 to IEEE Std 802.11be-2024. |
| 2114 | 9.4.2.46 | 69.18 | Missing words in sentence | Change this clause: "or the FTE is included in a frame between FTO with the (Re)Association Frame Encryption Support field in the RSNXE set to 1 and FTR with the (Re)Association Frame Encryption Support field in the RSNXE set to 1," to "or if the FTE is included in a frame between the FTO with the (Re)Association Frame Encryption Support field in the RSNXE set to 1 and the FTR with the (Re)Association Frame Encryption Support field in the RSNXE set to 1,". | Revised –  Agree in principle.  TGbi editor to make the changes shown in this document under all headings that include CID 2114 |
| 2090 | 9.4.2.46 | 69.18 | The text says, "When using AKM 00-0F-AC:25 or the FTE is included in a frame between FTO with the (Re)Association Frame Encryption Support field in the RSNXE set to 1 and FTR with the (Re)Association Frame Encryption Support field in the RSNXE set to 1, the MIC Length subfield defines the length of the MIC field, as defined in Table 9-220 (MIC Length subfield values)." What does the MIC length subfield define when those conditions are not met? | Fix the runon, this is a very confusing sentence. You can't just add mult-conditional statements to existing text and expect it to make sense. | Revised –  When the conditions are not met, the field is reserved. *This subfield is reserved otherwise.*  We improve the sentence.  TGbi editor to make the changes shown in this document under all headings that include CID 2114 |
| 2164 | 6.5.5.2.2 | 35.25 | TIM element is not needed in the authentication frame carrying 802.1X information. | Remove TIM element | Revised –  Agree in principle. The correct reference is RSNE. We update the reference.  TGbi editor to make the changes shown in this document under all headings that include CID 2164 |
| 2165 | 6.5.5.2.2 | 35.34 | TIM element is not needed in the authentication frame carrying EPPKE information. | Remove TIM element | Revised –  Agree in principle. The correct reference is RSNE. We update the reference.  TGbi editor to make the changes shown in this document under all headings that include CID 2164 |
| 2166 | 6.5.14.1.2 | 37.21 | TIM element is not needed in the authentication frame carrying 802.1X information. | Remove TIM element | Revised –  Agree in principle. The correct reference is RSNE. We update the reference.  TGbi editor to make the changes shown in this document under all headings that include CID 2164 |
| 2167 | 6.5.14.1.2 | 37.32 | TIM element is not needed in the authentication frame carrying EPPKE information. | Remove TIM element | Revised –  Agree in principle. The correct reference is RSNE. We update the reference.  TGbi editor to make the changes shown in this document under all headings that include CID 2164 |
| 2174 | 12.16.8 | 162.01 | Many Diffie-Hellman ephemeral public keys based PTK creation procols are proposed. The rekey operation is still based on the nonces and 4-way handshake. The rekey operation that uses Diffe-Hellman ephemeral public keysshould be defined to support all PTK creation principles. | Please define PTK rekey protocol that uses Diffie-Hellman ephemeral public keys. | Rejected –  Diffie-Hellman ephemeral public key exchange is useful for the initial PTK derivation because the nonce exchange is on the clear, and if PMK is leaked, then attacker can record the exchange to derive PTK. For the PTK rekey using four way handshake, the data frame is encrypted with the existing PTK, so the nonce is not on the clear, and even if the PMK is leaked, then the attacker can not recover the nonce from the encrypted data frame. |
| 2176 | 12.16.8.2 | 163.59 | The 802.1X shall use MLD addresses when it creates PTK for the MLO supplicant and authenticator identifying, as written in 802.11mf d1.0 p.3345 L1. The 802.1X Authentication frame does not specity the use of the ML element clearly. The ML element needs to be present, because the associating non-AP MLD needs to be identified by the MLD address. | Please add normative text how the MLD addresses are signaled and used in the 802.1X in authentication frames scheme. | Revised –  Agree in principle with the commenter. Note that 11bi does not inherit revmf. However, in 11be we already mandate for MLO, inclusion of Basic multi-link element is mandated as defined in Table 9-70—Authentication frame body in 11be.  *The Basic Multi-Link element is present if dot11MultiLinkActi-vated is true and the frame exchange is with a peer STA that is affiliated with an MLD. Otherwise, it is not present.*  Then it is not needed to specify again in other places like Table 9-83—Presence of fields and elements in Authentication frames.  In any case, we add a note to clarify this point.  TGbi editor to make the changes shown in this document under all headings that include CID 2176 |
| 2168 | 6.5.5.2.2 | 35.30 | Text for EPPKE defines that ML element shalle be included to the 802.11 authentication frames that carry EPPKE information, but ML element is not included here. | Please add ML element to EPPKE signalign. | Revised –  Agree in principle with the commenter. Note that 11bi does not inherit revmf. However, in 11be we already mandate for MLO, inclusion of Basic multi-link element is mandated as defined in Table 9-70—Authentication frame body in 11be.  *The Basic Multi-Link element is present if dot11MultiLinkActi-vated is true and the frame exchange is with a peer STA that is affiliated with an MLD. Otherwise, it is not present.*  Then it is not needed to specify again in other places like Table 9-83—Presence of fields and elements in Authentication frames.  In any case, we add a note to clarify this point.  TGbi editor to make the changes shown in this document under all headings that include CID 2176 |
| 2179 | 12.16.6 | 153.61 | The encrypted association frames are protected for MLO and non-MLO, but EPPKE only supports MLDs. Also 802.1X authentication signaling is broken, because signaling does not explain how ML elements are applied. | Please clarify encrypted (re)association frames dependency with EPPKE and 802.1X and clarify can non-MLO use these authentication schemes and thus the encrypted association signaling. | Revised -  EPPKE can be used for non-MLO. Note that in the EPPKE texts, we have texts for MLO, which means the case for MLO, and not the case for non-MLO.  802.1X can also be used for MLO. 11be defines a general statement in Table 9-70—Authentication frame body in 11be.  *The Basic Multi-Link element is present if dot11MultiLinkActi-vated is true and the frame exchange is with a peer STA that is affiliated with an MLD. Otherwise, it is not present.*  Then it is not needed to specify again in other places like Table 9-83—Presence of fields and elements in Authentication frames. Hence, the design is not broken.  In any case, we add a note to clarify this point.  TGbi editor to make the changes shown in this document under all headings that include CID 2176 |
| 2169 | 6.5.5.2.2 | 35.15 | The 802.1X shall use MLD addresses when it creates PTK for the MLO supplicant and authenticator identifying, as written in 802.11mf d1.0 p.3345 L1. The 802.1X Authentication frame does not specity the use of the ML element clearly. The ML element needs to be present, because the associating non-AP MLD needs to be identified by the MLD address. | Please add ML element to 802.1X Authentication frame signalign. | Revised –  Agree in principle with the commenter. Note that 11bi does not inherit revmf. However, in 11be we already mandate for MLO, inclusion of Basic multi-link element is mandated as defined in Table 9-70—Authentication frame body in 11be.  *The Basic Multi-Link element is present if dot11MultiLinkActi-vated is true and the frame exchange is with a peer STA that is affiliated with an MLD. Otherwise, it is not present.*  Then it is not needed to specify again in other places like Table 9-83—Presence of fields and elements in Authentication frames.  In any case, we add a note to clarify this point.  TGbi editor to make the changes shown in this document under all headings that include CID 2176 |
| 2178 | 12.16.8.2 | 163.59 | The 802.1X in authentication frames have 802.1X port open already for the authentication frames. If the encrypted association is rejected after 802.1X authentication, then are retransmissions of the protected association frame possible? | Please clarify how 802.1X in authentcation frames works when the associaiton is rejected. For instance, is association request possible to send on the same link again? Is it possible to send the the encrypted association request on other link? | Rejected -  In MLO defined in 11be, it is not allowed to send association request/response on other links that is not used for authentication frame exchange. See 35.3.5.1 ML (re)setup procedure.  As for retransmitting the association request, we have the sentence “On a failed (re)association, the established PTKSA shall be irretrievably deleted.” Hence, retransmitting (re)association request is not possible. |
| 2177 | 12.16.9.1 | 167.17 | If STA authenticates succesfully with EPPKE, but the encrypted association request gets an encrypted association response with reject reason, then what STA needs to do? The rejected association means that the current link association will fail, and the PTK is only set for this link, so the STA needs to redo the authentication in other link? | Please clarify how EPPKE works when the associaiton is rejected. For instance, is association request possible to send on the same link again? Is it possible to send the the encrypted association request on other link? | Rejected -  In MLO defined in 11be, it is not allowed to send association request/response on other links that is not used for authentication frame exchange. See 35.3.5.1 ML (re)setup procedure.  As for retransmitting the association request, we have the sentence “On a failed (re)association, the established PTKSA shall be irretrievably deleted.” Hence, retransmitting (re)association request is not possible. |
| 2267 | 10.71.7 | 116.16 | Use consistent terms. There are 2 occurrences of "non-AP CPE MLD" and 53 occurrences of "CPE non-AP MLD". | Use consistent term "CPE non-AP MLD" | Revised –  Agree in principle with the commenter.  TGbi editor to make the changes shown in this document under all headings that include CID 2267 |
| 2268 | 10.71.8 | 117.20 | Use consistent terms.There are 2 occurrences of "non-AP BPE MLDs" and 22 occurrences of "BPE non-AP MLD". | Use consistent term "BPE non-AP MLD" | Revised –  Agree in principle with the commenter.  TGbi editor to make the changes shown in this document under all headings that include CID 2268 |
| 2281 | 12.16.6.2 | 159.27 | How about an unassociated non-AP MLD which never receives Beacon or Probe Response from the BPE AP? Similar issue on P159L28, P159L34, P159L35, P159L54, P159L55. | Change "received" to "(if received)" | Revised –  Agree in principle with the commenter. We simply remove “received” to align with the baseline texts.  TGbi editor to make the changes shown in this document under all headings that include CID 2281 |
| 2367 | 9.4.1.11 | 54.15 | The EPP Action frame should be a protected management frame. | Change the EPP to the Protected EPP. | Rejected –  EPP action frame is defined to be robust already. In the baseline, protected is added only if there is a corresponding frame that is not defined as robust. |
| 2465 | 4.5.4.2 | 28.16 | Text "This standard does not specify an EAP method that is mandatory to implement." refers to an EAP method however the previous sentence referring to EAP has been deleted. So, the cited text needs to modified to provide some context for EAP reference. | Suggest to change to "This standard does not specify an EAP method that is mandatory to implement for authentication based on IEEE Std 802.1X-2020" | Accepted - |
| 2468 | ï»¿6.5.14.1.2 | 38.04 | This MLME-SETKEYS is about setting keys in the MAC. What is the reason to include Authentication frame parameters in this MLME: "Content of 802.1X Authentication frame" "Content of EPPKE Authentication frame" | Either remove these parameters or if needed, add a NOTE to clarify why these authentication parameters needed for SETKEYs MLME. Also, in the baseline the MLME-SETKEYS.request includes one parameter called "KeyList". Is the first table proposing to add more parameters to the MLME or add more parameters to "SetKeyDescriptors"? If it is the 2nd one then new parameters need to part of the same table where parameters related to Key are defined. | Revised –  The table is supposed to be before 6.5.14.1.2 right after *The primitive parameters are as follows:*  *MLME-AUTHENTICATE.response(*  *....*  *Content of 802.1X Authentication frame,*  *Content of EPPKE Authentication frame,*  *VendorSpecificInfo*  )  TGbi editor to make sure that the table stays before 6.5.14.1.2. |
| 2469 | ï»¿9.3.3.5 | 42.06 | Association Request in baseline already includes Mobility Domain element (MDE) at Order 11. Why do we need to include MDE again at Order 72? | Remove inclusion of MDE again in the Association Request frame. Modify text for MDE inclusion at Order 11 to include aspects for 'encryption of Association Request' | Revised –  Agree in principle with the commenter. The baseline condition already covers the case. We simply remove the addition.  TGbi editor to make the changes shown in this document under all headings that include CID 2469 |
| 2488 | 12.16.4.2 | 150.60 | In 11be, TID-to-Link mapping is defined and a link that does not have any TID mapped to in UL or DL is considered a disabled link. On the disabled link only class 1 and class 2 management frames are allowed per 11be. The EPP Capabilities And Operation Parameters Request/Response frames would be class 3 frames and should be sent on enabled link only (that have at least one TID mapped to it in UL or DL) | Change "...through an affiliated AP over a setup link to the non-AP MLD." to "...through an affiliated AP over an enabled link to the non-AP MLD." | Accepted - |
| 2489 | ï»¿12.16.7.1 | 160.65 | In baseline, PMKID computation is tied to PMK. E.g. from REVmf pg 3349 ln 27: "PMKID = Truncate-128(HMAC-SHA-256(PMK, "PMK Name" || AA || SPA))" Here PMKID generation is not tied to the PMK. Please add a NOTE to clarify why PMKID generation does not need to be tied with the PMK in this case. | As in comment | Rejected –  In the baseline, the input key is not always PMK. In CNSA related AKM, PTK KCK is used to avoid information leakage of PMK. As a result, PMK is not really required to be as the input. The design of 11bi unifies all the case. |
| 2095 | 12.16.5 | 151.01 | Need to create a transcript of the EAPOL exchange and use that in the KDF to generate the PTK later on. | Require STAs (incl APs) that use 802.1X authentication over authentication frames to generate a hashed transcript of the exchange. Each authentication frame in the sequence gets added to the running hash. The result of a successful EAPOL over Auth frames exchange will be a PMK, PMKID, and a transcript hash. Those two things can then be used later on to generate the PTK. There is no need for nonces and DH keys and the like. This will bind the entire exchange to the key it generated. | Rejected –  DH parameter element is needed for perfect forward secrecy. The entire transcript can be recorded by 3 party and does not provide the benefits of perfect forward secrecy.  We also still need the nonce carrying in the frame for randomness even if we use the transcript mechanism.  The entire transcript will mean the entire frame exchange can be protected, but for important elements, all the checks are there.  For MDE, we have MDE check in 12.16.6. For FTE, we have FTE check in 12.16.6. For RSNE, we have RSNE check in 12.16.6. For RSNXE, we have we have RSNXE check in 12.16.6.  Hence, all the important elements are already cross checked following the baseline principle. |
| 2096 | 12.16.8.2 | 163.01 | This key confirmation and key derivation step should be a distinct authentication algorithm. Do not overload the EAPOL exchange with all this nonce and DH stuff. None of that is needed. | Require STAs (incl APs) that do EAPOL over Auth frames to immediately proceed to this exchange after successful completion of the EAPOL exchange. The result of the EAPOL exchange will be a PMK and a transcript hash. Those two things need to be bound with a domain separation tag in a KDF to derive the PTK. | Rejected –  DH parameter element is needed for perfect forward secrecy. The entire transcript can be recorded by 3 party and does not provide the benefits of perfect forward secrecy.  We also still need the nonce carrying in the frame for randomness even if we use the transcript mechanism.  The entire transcript will mean the entire frame exchange can be protected, but for important elements, all the checks are there.  For MDE, we have MDE check in 12.16.6. For FTE, we have FTE check in 12.16.6. For RSNE, we have RSNE check in 12.16.6. For RSNXE, we have we have RSNXE check in 12.16.6.  Hence, all the important elements are already cross checked following the baseline principle. |

***Discussion:***

***Proposal:***

**TGbi Editor: *Instruction: Modify 12.16.5 as follows***

* IEEE 802.1X authentication utilizing Authentication frames

If an AP sets the IEEE 802.1X Authentication Utilizing Authentication Frame Support field in the RSNXE that it transmits to 1, then a non-AP STA (originator) with dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated equal to true may signal its Supplicant to authenticate with the AP (responder) using IEEE Std 802.1X-2020 utilizing Authentication frames by transmitting the Authentication frames to the AP(#2002).

If any AP affiliated with an AP MLD sets the IEEE 802.1X Authentication Utilizing Authentication Frame Support field in the RSNXE that it transmits to 1, then a non-AP MLD (originator) with dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated equal to true may signal its Supplicant to authenticate with the AP MLD (responder) using IEEE Std 802.1X-2020 utilizing Authentication frames by transmitting the Authentication frames to the AP through a non-AP STA affiliated with the non-AP MLD.

When the originator is a non-AP MLD and the responder is an AP MLD, the RA field of an Authentication frame in response to an Authentication frame from the peer shall be set to the TA field of the Authentication frame from the peer.

If an originator chooses to initiate IEEE 802.1X authentication utilizing Authentication frames, it first selects an IEEE 802.1X AKM that is supported by the responder.

The originator then shall construct the first Authentication frame of the exchange as follows:

* Authentication Algorithm Number field is set to 8 (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 1.
* The Encapsulation field carries an EAPOL PDU from the PAE.
* Include the AKM Suite Selector element indicating the selected IEEE 802.1X AKM.

The originator sends the first Authentication frame to the responder.

Upon receiving the first Authentication frame, the responder shall:

* Validate that the AKM indicated in AKM Suite Selector element is an IEEE 802.1X AKM. Otherwise, processing status is set to STATUS\_INVALID\_AKMP.
* Validate that the selected IEEE 802.1X AKM indicated in AKM Suite Selector element is supported. Otherwise processing status is set to STATUS\_INVALID\_AKMP.
* If the validation is successful, extract an EAPOL PDU from the Encapsulation field, and forward the EAPOL PDU to the PAE.

The responder then shall construct the second Authentication frame of the exchange as follows:

* Authentication Algorithm Number field is set to 8 (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 2.
* Status Code field indicates the processing status.
* The Encapsulation Length field indicates 0 if the status is set to STATUS\_INVALID\_AKMP.
* The Encapsulation field (if present) carries an EAPOL PDU from the PAE.
* Include the AKM Suite Selector element indicating the same IEEE 802.1X AKM indicated in the first Authentication frame.

Once the processing is complete, the responder sends the second Authentication frame to the originator. If the processing status returned in the frame was not SUCCESS, the responder shall terminate the authentication.

Upon receiving the second Authentication frame, the originator shall:

* Validate that the Status Code field is SUCCESS. Otherwise, the originator shall terminate the authentication.
* Validate that the AKM indicated in AKM Suite Selector element is the same as the one indicated in the first Authentication frame. Otherwise, processing status is set to STATUS\_INVALID\_AKMP.
* If the validation is successful, extract an EAPOL PDU from the Encapsulation field (if present), and forward the EAPOL PDU to the PAE.

The originator then shall construct the third Authentication of the exchange as follows:

* Authentication Algorithm Number field is set to 8 (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 3.
* Status Code field indicates the processing status.
* The Encapsulation Length field indicates 0 if the status is set to STATUS\_INVALID\_AKMP.
* The Encapsulation field (if present) carries an EAPOL PDU from the PAE.

Once the processing is complete, the originator sends the third Authentication frame to the responder. If the processing status returned in the frame was not SUCCESS, the originator shall terminate the authentication.

Upon receiving the Authentication frame with Authentication Transaction Sequence Number field (#2050)larger than or equal to 3, the originator or the responder:

* Extract an EAPOL PDU (if present) from the Encapsulation field, and forward the EAPOL PDU to the PAE.
* Validate that the Status Code field is SUCCESS or 802\_1\_X\_AUTH\_SUCCESS. Otherwise, the originator shall terminate the authentication after forwarding the EAPOL PDU (if present) to the PAE.

If needed by the EAP method, the originator or the responder then shall construct the Authentication frame of the exchange in response to the Authentication frame with Authentication Transaction Sequence Number field (#2050)larger than or equal to 3, as follows:

* Authentication Algorithm Number field is set to 8 (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to the value of the Authentication Transaction Sequence Number field of the Authentication frame being responded to +1.
* The Encapsulation field (if present) carries an EAPOL PDU from the PAE.
* Status Code field indicates the processing status.
* For the responder, if the 802.1X authentication is successful, the Status Code field is set to 802\_1\_X\_AUTH\_SUCCESS.
* For the responder, if the 802.1X authentication fails, the Status Code field is set to 802\_1\_X\_AUTH\_FAILED.
* Otherwise, the Status Code field is set to the appropriate status code.

NOTE 1(#2176)—The number of Authentication frame exchanges depends on the EAP method in use.

Once the processing is complete, the originator or the responder sends the Authentication frame in response to the Authentication frame with Authentication Transaction Sequence Number field set to a value that is larger than or equal to 3, to its peer (if needed by the EAP method). If the processing status returned in the frame was not SUCCESS or 802\_1\_X\_AUTH\_SUCCESS, the originator or the responder shall terminate the authentication.

NOTE 2(#2176)—For MLO, Basic Multi-link element is included in every Authentication frame as defined in Table 9-70 (Authentication frame body).

**TGbi Editor: *Instruction: Modify 12.16.9 as follows***

* Enhanced Privacy Protection Key Exchange
* General

If dot11EPPKEActivated is true, then dot11EPPReAssociationFrameEncryptionSupportActivated and dot11KEKPASNActivated shall be set to true.

Enhanced Privacy Protection Key Exchange (EPPKE) is an RSNA authentication protocol that uses the PASN procedures (see 12.12 (Preassociation security negotiation)) with the following differences:

* SAE AKMP 00-0F-AC:8, 00-0F-AC:9, 00-0F-AC:24, or 00-0F-AC:25 can be used as the Base AKMP.
* When there is no Base AKMP, EPPKE is not used.
* The three Authentication frames have the Authentication Algorithm Number field set to 9 (EPPKE Authentication).
* The generated PTK is used as the initial PTK once associated.
* For MLO, Basic Multi-link element is included in every Authentication frame as defined in Table 9-70 (Authentication frame body).(#2176)

NOTE 1—The PTK-KEK derived in EPPKE will be used for group key handshake after association even if PTK-KEK is not used in EPPKE frame exchange.

**TGbi Editor: *Instruction: Modify 9.4.2.348 as follows***

* DS MAC Address element

The DS MAC Address element is used by either an EPP non-AP MLD or an EPP non-AP STA that is not affiliated with a non-AP MLD to provide the DS MAC address to an EPP AP MLD or and EPP AP, respectively, for the DS mapping.

The format of the DS MAC Address element is shown in Figure 9-1074dq (DS MAC Address element format).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Element ID | Length | Element ID Extension | DS MAC Address |
| Octets: | 1 | 1 | 1 | 6 |
| * DS MAC Address element format | | | | |

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1 (General).

The DS MAC Address field contains a MAC address that is used for the DS mapping.(#2021)

**TGbi Editor: *Instruction: Modify 9.4.2.46 as follows***

* FTE

***Change the fifth paragraph as follows:***

When using AKM 00-0F-AC:25 or if(#2114) the FTE is included in a frame between the(#2114) FTO with the (Re)Association Frame Encryption Support field in the RSNXE equal to 1 and the(#2114) FTR with the (Re)Association Frame Encryption Support field in the RSNXE equal to 1, the MIC Length subfield defines the length of the MIC field, as defined in Table 9-220 (MIC Length subfield values). This subfield is reserved ~~for other AKMs~~otherwise.

***Modify Table 9-220 (MIC Length subfield value) as follows:***

|  |  |
| --- | --- |
| * MIC Length subfield value | |
| Value | MIC field length in octets |
| 0 | 16 |
| 1 | 24 |
| 2 | 32 |
| 3 | 0 |
| 4~~3~~–7 | Reserved |

***Modify Table 9-221 (Subelement IDs) (not all lines shown) as follows:***

* ANA assignment is done
* Subelement IDs

|  |  |
| --- | --- |
| Value | Subelement Name |
| … |  |
| 11 | PGTK |
| 13 | Identity Key |

* revme D7.0 up to Figure 9-444, 11be D7.0 up to Figure 9-444c

***Modify the paragraph “The GTK subelement contains the GTK, ...” as follows:***

The GTK subelement contains the GTK, which is encrypted (see procedures in 13.8.5 (FT authentication sequence: contents of fourth message)) or is not encrypted (see procedures in 12.16.6 ((Re)Association Request/Response Frame Encryption)) and is defined in Figure 9-439 (GTK subelement format).

***Modify Figure 9-439 (GTK subelement format) as follows:***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key Info | Key Length | RSC | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 1 | 8 | ~~24–40~~ variable |
| * GTK subelement format | | | | | | |

***Modify the paragraph “The Wrapped Key field contains the wrapped GTK being distributed.” as follows:***

The Wrapped Key/Key field contains the wrapped GTK being distributed if the frame containing(#2065) the FTE is not encrypted or(#2115) the GTK being distributed if the frame containing(#2065) the FTE is encrypted.

***Modify Figure 9-441 (IGTK subelement format) as follows:***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | IPN | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | ~~24-40~~variable |
| * IGTK subelement format | | | | | | |

***Modify the paragraph “The Wrapped Key field contains the wrapped IGTK being distributed.” as follows:***

The Wrapped Key/Key field contains the wrapped IGTK being distributed if the frame containing(#2065) the FTE is not encrypted or (#2115)the IGTK being distributed if the frame containing(#2065) the FTE is encrypted.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | BIPN | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | ~~24-40~~variable |
| * BIGTK subelement format | | | | | | |

***Modify Figure 9-443 (BIGTK subelement format) as follows:***

***Modify the paragraph “The Wrapped Key field contains the wrapped BIGTK being distributed.” as follows:***

The Wrapped Key/Key field contains the wrapped BIGTK being distributed if the frame containing(#2065) the FTE is not encrypted or (#2115)the BIGTK being distributed if the frame containing(#2065) the FTE is encrypted.

***Modify Figure 9-444 (WIGTK subelement format) as follows:***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | WIPN | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | ~~24~~variable |
| * WIGTK subelement format | | | | | | |

***Modify the paragraph “The Wrapped Key field contains the wrapped WIGTK being distributed.” as follows:***

The Wrapped Key/Key field contains the wrapped WIGTK being distributed if the frame containing(#2065) the FTE is not encrypted or (#2115)the WIGTK being distributed if the frame containing(#2065) the FTE is encrypted.

***Modify the paragraph “The MLO GTK subelement contains the GTK for a link, ...” as follows:***

The MLO GTK subelement contains the GTK for a link, which is encrypted (see procedures in 13.8.5 (FT authentication sequence: contents of fourth message)) or is not encrypted (see procedures in 12.16.6 ((Re)Association Request/Response Frame Encryption))and is defined in Figure 9-444a (MLO GTK subelement format).

***Modify Figure 9-444a (MLO GTK subelement format) as follows:***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key Info | Link ID Info | Key Length | RSC | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 1 | 1 | 8 | ~~24–40~~variable |
| * MLO GTK subelement format | | | | | | | |

***Modify the paragraph “The definitions of the Key Info, Key Length, RSC, and Wrapped Key fields are the same as in the GTK subelement.” as follows:***

The definitions of the Key Info, Key Length, RSC, and Wrapped Key/Key fields are the same as in the GTK subelement.

***Modify Figure 9-444b (MLO IGTK subelement format) as follows:***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | IPN | Link ID Info | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | 1 | ~~24–40~~variable |
| * MLO IGTK subelement format | | | | | | | |

***Modify the paragraph “The definitions of the Key ID, IPN, Key Length, and Wrapped Key fields are the same as in the IGTK subelement.” as follows:***

The definitions of the Key ID, IPN, Key Length, and Wrapped Key/Key fields are the same as in the IGTK subelement.

***Modify Figure 9-444c (MLO BIGTK subelement format) as follows:***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | BIPN | Link ID Info | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | 1 | ~~24–40~~variable |
| * MLO BIGTK subelement format | | | | | | | |

***Modify the paragraph “The definitions of the Key ID, BIPN, Key Length, and Wrapped Key fields are the same as in the BIGTK subelement.” as follows:***

The definitions of the Key ID, BIPN, Key Length, and Wrapped Key/Key fields are the same as in the BIGTK subelement.

***Insert the following at the end of 9.4.2.46 (FTE):***

The PGTK subelement contains the PGTK, used to anonymize fields that are common for all STAs assigned to an EPP group. The PGTK subelement format is shown in Figure 9-442d (PGTK subelement format).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | PGTK Switch Time | Key Length | Wrapped Key/Key |
| Octets: | 1 | 1 | 8 | 1 | variable |
| * PGTK subelement format | | | | | |

The PGTK Switch Time field is as defined in Figure 12-50i (PGTK KDE format).

The Key Length field is the length of the PGTK in octets, not including any padding (see 13.8.5 (FT authentication sequence: contents of fourth message)).

The Wrapped Key/Key field contains the wrapped PGTK being distributed if the frame containing(#2065) the FTE is not encrypted or (#2115)the PGTK being distributed if the frame containing(#2065) the FTE is encrypted.

The Identity Key subelement contains the Identity Key, used to encrypt the identity of the AP MLD in the Privacy Beacons. The Identity Key subelement format is shown in Figure 9-442e (Identity Key subelement format).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Wrapped Key/Key | Key Lifetime |
| Octets: | 1 | 1 | 16 | 3 |
| * Identity Key subelement format | | | | |

The Wrapped Key/Keyfield contains the wrapped identity key being distributed if the frame containing(#2065) the FTE is not encrypted or (#2115)the identity key being distributed if the frame containing(#2065) the FTE is encrypted.

The Key Lifetime contains the Lifetime of the Key in units of the seconds. Value 0 indicates that the key has no lifetime.

**TGbi Editor: *Instruction: Modify 9.3.3.11 as follows***

* Authentication frame format
* revme D7.0 up to order 27, 11bh D6.0 no addition, 11be D7.0 up to order 28, 11bk D5.0 no addition, 11bf D8.0 no addition

***Modify Table 9-70 (Authentication frame body) as follows (not all lines shown):***

|  |  |  |
| --- | --- | --- |
| * Authentication frame body (continued) | | |
| Order | Information | Notes |
| … |  |  |
| 9 | Confirm | An unsigned integer encoded as described in 12.4.7.4 (Encoding and decoding of SAE Confirm messages). This is present only in certain Authentication frames as defined in Presence of fields and elements in Authentication frames. |
| 9a | Encapsulation Length | This field indicates the number of octets in the Encapsulation field as described in 9.4.1.82 (Encapsulation Length field). This is present only in certain Authentication frames as defined in Table 9-71 (Presence of fields and elements in Authentication frames). |
| 9b | Encapsulation | This field is used to carry an EAPOL PDU as described in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames). This is present only when the Encapsulation Length field is nonzero. |
| ... |  |  |
| 17 | ~~FILS~~ Nonce | The ~~FILS~~ Nonce element is present only in certain(#2100) ~~FILS~~ Authentication frames as defined in Table 9-71 (Presence of fields and elements in Authentication frames). |
| ... |  |  |
| 25 | PASN Parameters | A PASN Parameters element is present only in certain Authentication frames  as defined in Table 9-71 (Presence of fields and elements in Authentication frames). |
| ... |  |  |
| 29 | Diffie-Hellman Parameter | A Diffie-Hellman Parameter element is present only in certain Authentication frames as defined in Table 9-71 (Presence of fields and elements in Authentication frames). |

**TGbi Editor: *Instruction: Modify 6.5.5 as follows***

* Authenticate
* MLME-AUTHENTICATE.request
* Semantics of the service primitive

***Modify MLME-AUTHENTICATE.request and the table as follows (not all lines shown):***

The primitive parameters are as follows:

MLME-AUTHENTICATE.request(

...,

Content of 802.1X Authentication frame,

Content of EPPKE Authentication frame,

VendorSpecificInfo

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| ..... | ..... | ..... | ..... |
| AuthenticationType | Enumeration | OPEN\_SYSTEM, SHARED\_KEY, FAST\_BSS\_TRANSITION, SAE, FILS\_SHARED\_KEY WITHOUT\_PFS, FILS\_SHARED KEY\_WITH\_PFS, FILS\_PUBLIC\_KEY, 802\_1X, EPPKE | Specifies the type of authentication algorithm to use during the authentication process. |
| ..... | ..... | ...... | ....... |
| Content of 802.1X Authentication frame | Sequence of elements and fields | As defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames), 12.16.8.2 (IEEE 802.1X),  9.4.1.82 (Encapsulation Length field), 9.4.1.83 (Encapsulation field), 9.4.2.295 (AKM Suite Selector element), 9.4.2.23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.188 (FILS Nonce element), 9.4.2.312 (Diffie-Hellman Parameter element). | The set of elements and fields to be included in 802.1X Authentication frames. Present if AuthenticationType indicates 802\_1X and dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated is true, otherwise not present. |
| Content of EPPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2 (EPPKE frame construction and processing), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.47 (Timeout Interval element) | The set of elements and fields to be included in EPPKE Authentication frames. Present if AuthenticationType indicates EPPKE and dot11EPPKEActivated is true, otherwise not present. |
| VendorSpecificInfo | A set of elements | As defined in 9.4.2.24 (Vendor Specific element) | Zero or more elements. |

* MLME-AUTHENTICATE.confirm
* Semantics of the service primitive

***Modify MLME-AUTHENTICATE.confirm and the table as follows (not all lines shown):***

The primitive parameters are as follows:

MLME-AUTHENTICATE.confirm(

....

Content of 802.1X Authentication frame,

Content of EPPKE Authentication frame,

VendorSpecificInfo

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| ..... | ..... | ..... | ..... |
| AuthenticationType | Enumeration | OPEN\_SYSTEM, SHARED\_KEY FAST\_BSS\_TRANSITION, SAE, FILS\_SHARED KEY\_WITHOUT\_PFS, FILS\_SHARED\_KEY\_WITH\_PFS, FILS\_PUBLIC\_KEY, PASN, 802\_1X, EPPKE | Specifies the type of authentication algorithm that was used during the authentication process. This value matches the AuthenticationType parameter specified in the corresponding MLME-AUTHENTICATE.request primitive. |
| ..... | ..... | ..... | ..... |
| Content of 802.1X Authentication frame | Sequence of elements and fields | As defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames), 12.16.8.2 (IEEE 802.1X),  9.4.1.82 (Encapsulation Length field), 9.4.1.83 (Encapsulation field), 9.4.2.295 (AKM Suite Selector element), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.188 (FILS Nonce element), 9.4.2.312 (Diffie-Hellman Parameter element). | The set of elements and fields received in 802.1X Authentication frames. Present if AuthenticationType indicates 802\_1X and dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated is true, otherwise not present. |
| Content of EPPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2 (EPPKE frame construction and processing), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.47 (Timeout Interval element) | The set of elements and fields received in EPPKE Authentication frames. Present if AuthenticationType indicates EPPKE and dot11EPPKEActivated is true, otherwise not present. |
| VendorSpecificInfo | A set of elements | As defined in 9.4.2.24 (Vendor Specific element) | Zero or more elements. |

* MLME-AUTHENTICATE.indication
* Semantics of the service primitive

***Modify MLME-AUTHENTICATE.indication and the table as follows (not all lines shown):***

The primitive parameters are as follows:

MLME-AUTHENTICATE.indication(

....

Content of 802.1X Authentication frame,

Content of EPPKE Authentication frame,

VendorSpecificInfo

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| ..... | ..... | ..... | ..... |
| AuthenticationType | Enumeration | OPEN\_SYSTEM, SHARED\_KEY, FAST\_BSS\_ TRANSITION, SAE, FILS\_SHARED\_KEY\_WITHOUT\_PFS, FILS\_SHARED\_KEY\_WITH\_PFS, FILS\_PUBLIC\_KEY, PASN, 802\_1X, EPPKE | Specifies the type of authentication algorithm that was used during the authentication process. |
| ..... | ..... | ..... | ..... |
| Content of 802.1X Authentication frame | Sequence of elements and fields | As defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames), 12.16.8.2 (IEEE 802.1X),  9.4.1.82 (Encapsulation Length field), 9.4.1.83 (Encapsulation field), 9.4.2.295 (AKM Suite Selector element), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.188 (FILS Nonce element), 9.4.2.312 (Diffie-Hellman Parameter element). | The set of elements and fields received in 802.1X Authentication frames. Present if AuthenticationType indicates 802\_1X and dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated is true, otherwise not present. |
| Content of EPPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2 (EPPKE frame construction and processing), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.47 (Timeout Interval element) | The set of elements and fields received in EPPKE Authentication frames. Present if AuthenticationType indicates EPPKE and dot11EPPKEActivated is true, otherwise not present. |
| VendorSpecificInfo | A set of elements | As defined in 9.4.2.24 (Vendor Specific element) | Zero or more elements. |

* MLME-AUTHENTICATE.response
* Semantics of the service primitive

***Modify MLME-AUTHENTICATE.response and the table as follows (not all lines shown):***

The primitive parameters are as follows:

MLME-AUTHENTICATE.response(

....

Content of 802.1X Authentication frame,

Content of EPPKE Authentication frame,

VendorSpecificInfo  
)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| ..... | ..... | ..... | ..... |
| Content of 802.1X Authentication frame | Sequence of elements and fields | As defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames), 12.16.8.2 (IEEE 802.1X),  9.4.1.82 (Encapsulation Length field), 9.4.1.83 (Encapsulation field), 9.4.2.295 (AKM Suite Selector element), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.188 (FILS Nonce element), 9.4.2.312 (Diffie-Hellman Parameter element). | The set of elements and fields to be included in 802.1X Authentication frames. Present if AuthenticationType indicates 802\_1X and dot11EPPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated is true, otherwise not present. |
| Content of EPPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2 (EPPKE frame construction and processing), 9.4.2. 23 (RSNE)(#2164), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.47 (Timeout Interval element) | The set of elements and fields to be included in EPPKE Authentication frames. Present if AuthenticationType indicates EPPKE and dot11EPPKEActivated is true, otherwise not present. |
| VendorSpecificInfo | A set of elements | As defined in 9.4.2.24 (Vendor Specific element) | Zero or more elements. |

**TGbi Editor: *Instruction: Modify 12.16.8 as follows***

* Key derivation with Authentication frame exchange

This subclause defines rules to derive a PTK with(#2175) temporal key (TK) through Authentication frame exchange to encrypt the Frame Body field of the (Re)Association Request/Response frame.

**TGbi Editor: *Instruction: Modify 12.16.8.2 as follows***

* IEEE 802.1X

If an originator or a responder defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames) sets the (Re)Association Frame Encryption Support field in the RSNXE to 1, then the originator or the responder supports the additional rules defined in this subclause when performing IEEE 802.1X Authentication frame exchange.

An originator that sets the (Re)Association Frame Encryption Support field in the RSNXE to 1 receives an RSNXE from the responder with the (Re)Association Frame Encryption Support field set to 1, and intends to continue association after authentication shall do the following in the first Authentication frame:

* Include a Nonce element to indicate SNonce.
* Include an RSNE to indicate the AKM and the pairwise cipher suite. The Version field shall be set to 1. The Pairwise Cipher Suite Count field shall be set to 1. The AKM Suite Count field shall be set to 1. The PMKID count field and the PMKID List field is set corresponding to PMKSA identifiers if exists. All other fields shall be as specified in 9.4.2.5 (TIM element) and 12.6.3 (RSNA policy selection in an infrastructure BSS).
* Not include an AKM Suite Selector element.
* Include an RSNXE.
* Include a Diffie-Hellman Parameter element.
* Select a finite cyclic group in the Diffie-Hellman Parameter element from the dot11RSNAConfigDLCGroupTable that is at least of the security strength provided by the AKM and cipher suites.
* With the chosen finite cyclic group, generate an ephemeral (random) private key, use the selected group’s scalar operation (see 12.4.4.1 (General)) with the private key to generate its ephemeral public key, and indicate the ephemeral public key in the Diffie-Hellman Parameter element.

Otherwise, an originator that sets dot11EPPReAssociationFrameEncryptionSupportActivated to false or does not receive the RSNXE from the responder with the (Re)Association Frame Encryption Support field set to 1 shall not include a Diffie-Hellman Parameter element nor an RSNE nor an RSNXE nor a Nonce element in the first Authentication frame for IEEE 802.1X authentication.

For the purpose of interoperability, an authenticator and a supplicant shall support group 19, an ECC group defined over a 256-bit prime order field.

A responder that sets the (Re)Association Frame Encryption Support field in the RSNXE to 1 receives the first Authentication frame with a Nonce element, RSNE, RSNXE, and a Diffie-Hellman Parameter element shall:

* Verify that the AKM indicated in the RSNE is supported. Otherwise, the responder shall reject the first message with status code set to STATUS\_INVALID\_AKMP.
* Verify that the pairwise cipher indicated in the RSNE is supported. Otherwise, the responder shall reject the first message with status code set to STATUS\_INVALID\_PAIRWISE\_CIPHER.
* Validate that the finite cyclic group indicated in the Diffie-Hellman Parameter element in the first Authentication frame is supported (present in dot11RSNAConfigDLCGroupTable). Otherwise, the responder shall reject the first message with status code set to UNSUPPORTED\_FINITE\_CYCLIC\_GROUP.
* Verify the public key indicated in the Diffie-Hellman Parameter element in the first message as specified in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the responder shall reject the first Authentication frame with status code set to INVALID\_PUBLIC\_KEY.
* Verify that a PMKSA named via a PMKID in the RSNE exists for the specified AKM in the RSNE if one or more PMKIDs are included.
* If a PMKSA is identified, the responder shall use PMKSA caching, shall not process the EAPOL PDU in the first Authentication frame, and shall not include the EAPOL PDU in the second authentication frame.
* If no PMKSA is identified, continue the IEEE 802.1X authentication.

If the first Authentication frame is not rejected, the responder shall:

* Store the indicated SNonce and generate an ephemeral (random) private key with the chosen finite cyclic group and use the selected group’s scalar operation with the private key to generate its ephemeral public key.
* Perform the group's scalar-op (see 12.4.4.1 (General)) with the originator’s ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.
* Use PMKSA caching if a PMKSA is identified via a PMKID in the RSNE in the first Authentication frame and before sending the second Authentication frame:
* Derive PTK with the identified PMKSA and DHss as defined in 12.7.1.3 (Pairwise key hierarchy).
* Irretrievably delete the shared secret, DHss, upon completion of PTK generation.

The responder shall do the following in the second Authentication frame:

* Include an RSNE to indicate the AKM and pairwise cipher indicated in the first Authentication frame.
* If a PMKSA is identified via a PMKID in the RSNE in the first Authentication frame, the responder shall include the PMKID corresponding to the PMKSA in the RSNE.
* Otherwise, the responder shall not include any PMKID in the RSNE.
* Not include an AKM Suite Selector element.
* Include a Diffie-Hellman Parameter element.
* Indicate the chosen finite cyclic group in the Diffie-Hellman Parameter element, which is the same as the finite cyclic group in the Diffie-Hellman Parameter element of the first Authentication frame.
* Indicate its ephemeral public key in the Diffie-Hellman Parameter element.
* Include a Nonce element to indicate ANonce.
* If a PMKSA is identified via a PMKID in the RSNE in the first Authentication frame, include a MIC element where the MIC field of the MIC element is set to the first MMM octets of: (#2175)

HMAC-HASH (KCK, AA || SPA || AP RSNE || AP RSNXE || Frame Data) Equation (XXX)

Where

HASH is the hash algorithm from the key derivation type (see Table 9-190 (AKM suite

selectors)) for each AKM

KCK is the key confirmation key for the PTKSA

AA is the authenticator address

SPA is the supplicant address

AP RSNE is the RSN element sent by the AP (e.g. RSNE in the Beacon frames or Probe Response frames) that sends the second Authentication frame

AP RSNXE is the RSNXE sent by the AP (e.g. RSNE in the Beacon frames or Probe Response frames) that sends the second

Authentication frame

Frame Data is the body of the second Authentication frame including the MIC element with the

octets in the MIC field of the MIC element set to 0

MMM is half of the output length in octets for the hash function used, that is, 16 or 24 octets for SHA-256 and SHA-384, respectively

Otherwise, a responder that sets dot11EPPReAssociationFrameEncryptionSupportActivated to false or does not receive the RSNXE in the first Authentication frame with the (Re)Association Frame Encryption Support field set to 1 shall not include a Diffie-Hellman Parameter element nor a Nonce element nor an RSNE in the second Authentication frame for IEEE 802.1X authentication.

After receiving the second Authentication frame with the status code set to SUCCESS, an originator shall:

* Validate that there is a Diffie-Hellman Parameter element and an RSNE included in the second Authentication frame and there is no AKM Suite Selector element in the second Authentication frame if the originator included a Diffie-Hellman Parameter element in the first Authentication frame. If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* Validate that there is no Diffie-Hellman Parameter element and no RSNE included in the second Authentication frame if the originator did not include a Diffie-Hellman Parameter element in the first Authentication frame. If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* Validate that the finite cyclic group indicated in the Diffie-Hellman Parameter element in the second Authentication frame is the same as the finite cyclic group indicated in the Diffie-Hellman Parameter element in the first Authentication frame if the originator included a Diffie-Hellman Parameter element in the first Authentication frame. Validate that the pairwise cipher suite and the AKM indicated in the second Authentication frame are the same as the pairwise cipher suite and the AKM indicated in the first Authentication frame if the originator includes a Diffie-Hellman Parameter element in the first Authentication frame. If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* Verify the public key indicated in the Diffie-Hellman Parameter element in the second Authentication frame as specified in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the originator shall discard the frame and terminate further protocol processing.
* Validate that the Encapsulation Length field is set to 0 and validate that the PMKID included in the second Authentication frame matches one of the PMKID(s) indicated in the first Authentication frame if the originator includes one or more PMKIDs in the first Authentication frame, and the second Authentication frame includes a PMKID. If verification succeeds, the originator shall use PMKSA caching with the PMKSA identified by the PMKID indicated in the second Authentication frame and shall not continue the IEEE 802.1X Authentication frame exchange. If verification fails, the originator shall discard the frame and terminate further protocol processing.
* Validate that there is no PMKID included in the second Authentication frame if the originator does not include any PMKID in the first Authentication frame. If verification fails, the originator shall discard the frame and terminate further protocol processing.
* Store the indicated ANonce, perform the group's scalar-op (see 12.4.4.1 (General)) with the originator’s ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss, if the second Authentication frame is not discarded.
* Derive the PTK with the identified PMKSA and DHss as defined in 12.7.1.3 (Pairwise key hierarchy) if a PMKSA is identified. Irretrievably delete the shared secret, DHss, upon completion of PTK generation. Compute the MIC as specified in equation XXX and verify it to be the same as the MIC provided in the MIC element. If validation fails, the originator shall discard the frame. (#2175)

If a PMKSA is not identified through PMKSA caching, before sending the Authentication frame carrying EAP Success, a responder shall:

* Derive the PTK with DHss as defined in 12.7.1.3 (Pairwise key hierarchy).
* Irretrievably delete the shared secret, DHss, upon completion of PTK generation.

If a PMKSA is not identified through PMKSA caching, after receiving the Authentication frame carrying EAP Success, an originator shall:

* Derive the PTK with DHss as defined in 12.7.1.3 (Pairwise key hierarchy).
* Irretrievably delete the shared secret, DHss, upon completion of PTK generation.

**TGbi Editor: *Instruction: Modify 9.3.3.11 as follows***

* Authentication frame format

|  |  |  |  |
| --- | --- | --- | --- |
| * Presence of fields and elements in Authentication frames (continued) | | | |
| Authentication algorithm | Authentication transaction sequence number | Status code | Presence of fields and elements  indicated as conditional in Table 9-70 (Authentication frame body) |
| .... |  |  |  |
| IEEE 802.1X authentication | 1 | Reserved | The Encapsulation Length field is present.  The Encapsulation field is present only when the Encapsulation Length field is nonzero.  The AKM Suite Selector element is optionally present as defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames).  The RSNE is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The RSNXE is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The Nonce element is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The Diffie-Hellman Parameter element is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The MDE is optionally present as defined in 12.16.8.3 (FT initial mobility domain association). |
| IEEE 802.1X authentication | 2 | SUCCESS | The Encapsulation Length field is present.  The Encapsulation field is present only when the Encapsulation Length field is nonzero.  The AKM Suite Selector element is optionally present as defined in 12.16.5 (IEEE 802.1X authentication utilizing Authentication frames).  The RSNE is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The Nonce element is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The Diffie-Hellman Parameter element is optionally present as defined in 12.16.8.2 (IEEE 802.1X).  The MDE is optionally present as defined in 12.16.8.3 (FT initial mobility domain association).  The FTE is optionally present as defined in 12.16.8.3 (FT initial mobility domain association).  The MIC element is optionally present as defined in 12.16.8.2 (IEEE 802.1X). (#2175) |
| IEEE 802.1X authentication | 2 | Not SUCCESS | The Encapsulation Length field is present.  The Encapsulation field is present only when the Encapsulation Length field is nonzero. |
| IEEE 802.1X authentication | > =3 | Status | The Encapsulation Length field is present.  The Encapsulation field is present only when the Encapsulation Length field is nonzero. |

**TGbi Editor: *Instruction: Modify 10.71.7 as follows***

**10.71.7 Frame anonymization and AID**

(…existing texts…)

Before the end of all the epochs indicated in the Number of Epochs field, the CPE AP MLD shall generate a new list of AID values and send a new AID Assignment Request frame with the new AID List element to the associated CPE non-AP MLD.(#2267)

The CPE AP MLD may generate a new AID list and send a new AID Assignment Requestframe with the new AID List element to the associated CPE non-AP MLD. (#2267)

(…existing texts…)

**TGbi Editor: *Instruction: Modify 10.71.8 as follows***

**10.71.8 BSS privacy enhancements operations**

(…existing texts…)

The associated BPE non-AP MLDs(#2268) and BPE AP MLD operate in a single EPP group. At the beginning of each epoch, the BPE non-AP STA addresses and SN spaces and PNs of the individual frames are anonymized in all links according to CPE anonymization, see 10.71.3 (Establishing CPE MAC header anonymization parameter sets). The BPE MLD affiliated AP addresses, the Timestamp field of the Privacy Beacons and the group frames are anonymized according to BPE anonymization, see 10.71.4 (Establishing BPE MAC header anonymization parameter sets). The AIDs used by the associated BPE non-AP MLDs(#2268) are assigned by the AP MLD, see 10.71.7 (Frame anonymization and AID).

**TGbi Editor: *Instruction: Modify 12.2.7 as follows***

* Requirements for management frame protection

***Change the first paragraph as follows:***

The robust Management frames are Disassociation, Deauthentication, robust Action, ~~and~~ robust Action No Ack frames, and if encryption of (Re)Association Request/Response frames is negotiated(#2335) (see 12.16.6 ((Re)Association Request/Response Frame Encryption)), Association Request frame, Association Response frame, Reassociation Request frame, and Reassociation Response frame.

**TGbi Editor: *Instruction: Modify 12.6.1.2.2 as follows***

* Security association in an ESS

(…existing texts…)

***Change item d) of the second paragraph (not all shown) as follows:***

* The last step is key management. The authentication process, whether SAE authentication utilizing Authentication frames, or FILS authentication utilizing Authentication frame, or IEEE 802.1X authentication utilizing Authentication frames, or IEEE 802.1X authentication utilizing Data frames post association, or the OWE exchange utilizing association frames creates cryptographic keys shared between the cryptographic endpoints—the AP and STA, or the IEEE 802.1X AS and the STA, when using SAE/FILS/OWE or IEEE Std 802.1X, respectively. When using IEEE Std 802.1X, the AS transfers these keys to the AP, and if encryption of (Re)Association Request/Response frames is not negotiated, the AP and STA uses one of the key confirmation handshakes, e.g., the 4-way handshake or FT 4-way handshake, to complete security association establishment. When using SAE authentication or OWE there is no AS and therefore no key transfer; if encryption of (Re)Association Request/Response frames is not negotiated, the 4-way handshake is performed directly between the AP and STA. The key confirmation handshake indicates when the link has been secured by the keys and is ready to allow normal data traffic and protected robust Management frames. When FILS authentication is performed or if encryption of (Re)Association Request/Response frames is negotiated, the key confirmation is performed using association frames. Hence, no additional handshake is necessary.(#2335)

(…existing texts…)

**TGbi Editor: *Instruction: Modify 12.6.17 as follows***

* Protection of robust Management frames

***Insert the following after “note 6 - The STA is not sent*** ...***” as shown below***

If management frame protection is negotiated for the link and encryption of (Re)Association Request/Response frames is negotiated(#2335), a STA shall not transmit any of the following, and shall discard all of the following:

* An unprotected Association Request frame, Association Response frame, Reassociation Request frame, or Reassociation Response frame.

**TGbi Editor: *Instruction: Modify 12.16.6 as follows***

* (Re)Association Request/Response Frame Encryption

This subclause defines rules to encrypt the Frame Body field of the (Re)Association Request/Response frame and to include a DS MAC Address element in the encrypted (Re)Association Request frame.

An EPP STA that sets the (Re)Association Frame Encryption Support field in the RSNXE to 1 shall set the MFPC subfield in the RSN Capabilities field in the RSNE to 1.

* Non-MLO procedure

Encryption of (Re)Association Request/Response frames is negotiated if both an EPP non-AP STA and an EPP AP set the (Re)Association Frame Encryption Support field in the RSNXE to 1.(#2335)

An EPP non-AP STA that negotiates encryption of (Re)Association Request/Response frames shall indicate a pairwise cipher, establish a PTKSA, and derive a temporal key (TK) through Authentication frame exchange with an EPP AP (see 12.16.8 (Key derivation with Authentication frame exchange) and 12.16.9 (Enhanced Privacy Protection Key Exchange)).(#2335)

An EPP non-AP STA shall randomize its STA MAC address during a BSS transition if the BSS transition procedure uses an encrypted (Re)Association Request frame to carry the DS MAC Address element.

After a pairwise cipher is indicated by the EPP non-AP STA and a temporal key (TK) is derived during Authentication frame exchange between the EPP non-AP STA and an EPP AP, the EPP non-AP STA shall encrypt the (Re)Association Request frame transmitted to the EPP AP using the TK and the pairwise cipher indicated in the Authentication frame exchange.

If the FT initial mobility domain association is used, then the EPP non-AP STA shall include the MDE and the FTE in the (Re)Association Request frame. MDE and the FTE shall be the same as the ones in the second Authentication frame as described in 12.16.8.3 (FT initial mobility domain association).

If the FT initial mobility domain association is used, then the PMKR1Name shall be included in the RSNE in the (Re)Association Request frame. The PMKR1Name shall be as calculated by the S1KH according to the procedures of 12.7.1.6.4 (PMK-R1).

For the RSNE included in the (Re)Association Request frame, other than the Length field, the PMKID Count field and the PMKID List field, shall be identical to those in the RSNE present in the first Authentication frame.

If the FT protocol is used, then the EPP non-AP STA shall not calculate the MIC for the MIC field of the FTE in the Reassociation Request frame. The length of the MIC field of the FTE in the Reassociation Request frame shall be 0 (i.e., the MIC Length subfield of the MIC Control field of the FTE is set to 3). The Element Count subfield of the MIC Control field of the FTE shall be set to 0.

If the FILS authentication protocol is used, then the EPP non-AP STA shall not encrypt the (Re)Association Request frame using the AEAD algorithm as described in 12.11.2.6.2 (Re)Association Request for FILS key confirmation.

If dot11DSMACAddressActivated is true and the EPP AP sets the DS MAC Address Support field in the RSNXE to 1, the EPP non-AP STA shall include the DS MAC Address element in the (Re)Association Request frame to indicate the DS MAC address to be used by the EPP AP for the mapping to the DS.

The EPP non-AP STA may randomize the DS MAC address. To construct a random DS MAC address, the EPP non-AP STA shall select the randomized DS MAC address according to IEEE Std 802-2014 and IEEE Std 802c-2017. If dot11DSMACAddressActivated is true, the EPP non-AP STA shall use the same DS MAC address for the duration of its connection across an ESS.

NOTE 2—Detection and remediation of possible DS MAC address collisions are outside the scope of this standard.

The EPP AP shall decrypt the (Re)Association Request frame received from the EPP non-AP STA using the TK and the pairwise cipher indicated in the Authentication frame exchange. If there is no output from the decryption algorithm because of unsuccessful MIC check (see 12.5.2.4.2 (CCM recipient processing) and 12.5.4.4.2 (GCM recipient processing)), then the EPP AP shall discard the frame.

If the FT initial mobility domain association is used, then the FTE and the MDE in the (Re)Association Request frame are checked to be the same as those provided in the second Authentication frame as defined in 12.16.8.3 (FT initial mobility domain association). If the MDE check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_MDE. If the FTE check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_FTE.

If the FT initial mobility domain association is used, then the PMKR1Name in the RSNE in the (Re)Association Request frame is checked to be included and calculated according to the procedures of 12.7.1.6.4 (PMK-R1). If the check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_PMKID.

The EPP AP shall verify that the RSNE fields other than the Length field, the PMKID Count field and the PMKID List field in the (Re)Association Request frame is identical to the RSNE fields included in the first Authentication frame. The EPP AP shall also verify that the RSNXE in the (Re)Association Request frame is identical to the RSNXE included in the first Authentication frame. If the validation fails, the EPP AP shall reject the association.

If the FT protocol is not used and the FT initial mobility domain association is not used, the RSNE included in the (Re)Association Response frame shall be identical to the corresponding RSNE fields in the Beacon or Probe Response frames from the EPP AP. (#2335)

If the FT initial mobility domain association is used, then EPP AP shall include the MDE, the FTE, TIE[ReassociationDeadline], and TIE[KeyLifetime] in the (Re)Association Response frame. The MDE and the FTE shall be the same as the ones in the second Authentication frame as described in 12.16.8.3 (FT initial mobility domain association). The reassociation deadline timeout is set to the minimum of dot11FTReassociationDeadline and the key lifetime in the TIE[ReassociationDeadline], and the PTK lifetime in the TIE[KeyLifetime].

If the FT initial mobility domain association is used, then the PMKR1Name shall be included in the RSNE in the (Re)Association Response frame. The PMKR1Name shall be as calculated by the R1KH according to the procedures of 12.7.1.6.4 (PMK-R1). The RSNE fields other than the Length field, the PMKID Count field and the PMKID List field shall be identical to the corresponding RSNE fields in the Beacon or Probe Response frames (#2281) from the EPP AP. (#2335)

If the FT protocol is used, then the EPP AP shall not wrap the Key field of the subelements in the FTE in the Reassociation Response frame and shall not calculate the MIC for the MIC field of the FTE in the Reassociation Response frame. The length of the MIC field of the FTE in the Reassociation Response frame shall be 0 (i.e., the MIC Length subfield of the MIC Control field of the FTE is set to 3). The Element Count subfield of the MIC Control field of the FTE shall be set to 0.

If the FILS authentication protocol is used, then the EPP AP shall not encrypt the (Re)Association Response frame using the AEAD algorithm as described in 12.11.2.6.3 ((Re)Association Response for FILS key confirmation).

The EPP AP shall encrypt the (Re)Association Response frame transmitted to the EPP non-AP STA in response to the (Re)Association Request frame using the TK and the pairwise cipher indicated in the Authentication frame exchange.

If the FILS authentication protocol and the FT protocol are not used, the EPP AP shall include a Key Delivery element in the (Re)Association Response frame.

If a Key Delivery element is included in the (Re)Association Response frame, the EPP AP shall construct the Key Delivery element indicating the current GTK PN in the RSC subfield, with the GTK KDE, with the IGTK KDE if management frame protection is enabled, with the BIGTK KDE if beacon protection is enabled, and with the WIGTK KDE if WUR frame protection is enabled.

The EPP non-AP STA shall decrypt the (Re)Association Response frame received from the EPP AP using the TK and the pairwise cipher indicated in the Authentication frame exchange. If there is no output from the decryption algorithm because of unsuccessful MIC check (see 12.5.2.4.2 (CCM recipient processing) and 12.5.4.4.2 (GCM recipient processing)), then the EPP non-AP STA shall discard the frame.

If the FT protocol is not used, the FT initial mobility domain association is not used, and in the (Re)Association Response frame the RSNE fields are not identical to the corresponding RSNE fields in the Beacon or Probe Response frames (#2281) from the EPP AP, the EPP non-AP STA shall discard the response.

If the FT protocol is not used, the EPP non-AP STA shall verify that the RSNXE included in the (Re)Association Response frame is identical to the RSNXE included in the Beacon or Probe Response frames (#2281) from the EPP AP. If those frames did not include the RSNXE or if the RSNXEs are not identical, the EPP non-AP STA shall discard the response.

If the FT initial mobility domain association is used, then the FTE and the MDE in the (Re)Association Response frame are checked to be the same as those provided in the second Authentication frame as defined in 12.16.8.3 (FT initial mobility domain association). If the check fails, the EPP non-AP STA shall discard the response.

If the FT initial mobility domain association is used, then the PMKR1Name in the RSNE in the (Re)Association Response frame is checked to be included and identical to the value that is sent in the Association Request frame. If the check fails, the EPP non-AP STA shall discard the response.

If the FT initial mobility domain association is used and in the (Re)Association Response frame the RSNE fields other than the Length field, the PMKID Count field and the PMKID List field are not identical to the corresponding RSNE fields in the Beacon or Probe Response frames (#2281) from the EPP AP, the EPP non-AP STA shall discard the response.

If IEEE 802.1X Authentication utilizing Authentication frame is used, and the RSN capabilities fields of the RSNE received in the (Re)Association Response frame is not identical to the RSN capabilities fields of the RSNE received in the second Authentication frame, the EPP non-AP STA shall discard the response.

On a successful (re)association,

* The EPP non-AP STA shall process the Key Delivery element in the (Re)Association Response frame if present.
* The EPP non-AP STA shall install the GTK and GTK RSC, and IGTK and IGTK RSC if management frame protection is enabled, and BIGTK and BIGTK RSC if present in the Key Delivery element and dot11BeaconProtectionEnabled is true, and WIGTK and WIGTK RSC if present in the Key Delivery element and dot11RSNAWURFrameProtectionActivated is true.
* The EPP AP and the EPP non-AP STA shall transition to State 4 (as defined in 11.3 (STA authentication and association)).
* If the DS MAC Address element is included in the (Re)Association Request frame, the EPP non-AP STA shall use the indicated DS MAC address rather than the MAC address of the EPP non-AP STA for the EPP non-AP STA to the EPP AP mapping to the DS.
* If the DS MAC Address element is included in the (Re)Association Request frame, the EPP AP shall process the DS MAC Address element and use the indicated DS MAC address rather than the MAC address of the EPP non-AP STA to establish the EPP non-AP STA to the EPP AP mapping to the DS.

NOTE 1—If the DS MAC Address element is included in the (Re)Association Request frame, the source address or destination address parameters of the MAC service tuples (see 5.2.4.2 (Semantics of the service primitive)) for the EPP non-AP STA are set to the DS MAC address, which is the identity of the non-AP STA known by the DS.

On a failed (re)association, the established PTKSA shall be irretrievably deleted.

* MLO procedure

Encryption of (Re)Association Request/Response frames is negotiated if both an EPP non-AP MLD and an EPP AP MLD set the (Re)Association Frame Encryption Support field in the RSNXE to 1.(#2335)

A non-AP MLD that negotiates encryption of (Re)Association Request/Response frames shall indicate a pairwise cipher, establish a PTKSA, and derive a temporal key (TK) through Authentication frame exchange with an EPP AP MLD (see 12.16.8 (Key derivation with Authentication frame exchange) and 12.16.9 (Enhanced Privacy Protection Key Exchange)). (#2335)

NOTE 1—For MLO, all STAs affiliated with an MLD set the RSNXE to the same value.

An EPP non-AP MLD shall randomize the STA MAC addresses of its affiliated STAs and its MLD MAC address during a BSS transition if the BSS transition procedure uses an encrypted (Re)Association Request frame to carry the DS MAC Address element.

After a pairwise cipher is indicated by the EPP non-AP MLD and a TK is derived during Authentication frame exchange between the EPP non-AP MLD and an EPP AP MLD, the EPP non-AP MLD shall encrypt the (Re)Association Request frame transmitted to the EPP AP MLD using the TK and the pairwise cipher indicated in the Authentication frame exchange.

If the FT initial mobility domain association is used, then the EPP non-AP MLD shall include the MDE and the FTE in the (Re)Association Request frame. The MDE and the FTE shall be the same as the ones in the second Authentication frame as defined in 12.16.8.3 (FT initial mobility domain association).

If the FT initial mobility domain association is used, then the PMKR1Name shall be included in the RSNE in the (Re)Association Request frame. The PMKR1Name shall be as calculated by the S1KH according to the procedures of 12.7.1.6.4 (PMK-R1).

For the RSNE included in the (Re)Association Request frame, other than the Length field, the PMKID Count field and the PMKID List field, shall be identical to the RSNE fields in the first Authentication frame.

The (Re)Association Request frame shall:

* Have the Address 1 field equal to the Address 1 field of the Authentication frame used by the non-AP MLD to establish the PTKSA.
* Have the Address 2 field equal to the Address 2 field of the Authentication frame used by the non-AP MLD to establish the PTKSA.
* Include the DS MAC Address element in the (Re)Association Request frame to indicate the DS MAC address to be used by the EPP AP MLD for the mapping to the DS if dot11DSMACAddressActivated is true and the APs affiliated with the EPP AP MLD set the DS MAC Address Support field in the RSNXE to 1.

If the FT protocol is used, then the EPP non-AP MLD shall not calculate the MIC for the MIC field of the FTE in the Reassociation Request frame. The length of the MIC field of the FTE in the Reassociation Request frame shall be 0 (i.e., the MIC Length subfield of the MIC Control field of the FTE is set to 3). The Element Count subfield of the MIC Control field of the FTE shall be set to 0.

If the FILS authentication protocol is used, then the EPP non-AP MLD shall not encrypt the (Re)Association Request frame using the AEAD algorithm as described in 12.11.2.6.2 (Re)Association Request for FILS key confirmation.

An EPP non-AP MLD may randomize its DS MAC address. To construct a random DS MAC address, the EPP non-AP MLD shall select the randomized DS MAC address according to IEEE Std 802-2014 and IEEE Std 802c-2017. If dot11DSMACAddressActivated is true, the EPP non-AP MLD shall use the same DS MAC address for the duration of its connection across an ESS.

NOTE 2—Detection and remediation of possible DS MAC address collisions are outside the scope of this standard.

The EPP AP MLD shall decrypt the (Re)Association Request frame received from the EPP non-AP MLD using the TK and the pairwise cipher indicated in the Authentication frame exchange. If there is no output from the decryption algorithm because of unsuccessful MIC check (see 12.5.2.4.2 (CCM recipient processing) and 12.5.4.4.2 (GCM recipient processing)), then the EPP AP MLD shall discard the frame.

If the FT initial mobility domain association is used, then the FTE and the MDE in the (Re)Association Request frame are checked to be the same as those provided in the second Authentication frame as defined in 12.16.8.3 (FT initial mobility domain association). If the MDE check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_MDE. If the FTE check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_FTE.

If the FT initial mobility domain association is used, then the PMKR1Name in the RSNE in the (Re)Association Request frame is checked to be included and calculated according to the procedures of 12.7.1.6.4 (PMK-R1). If the check fails, the (Re)Association Request frame is rejected with status code STATUS\_INVALID\_PMKID.

The EPP AP MLD shall verify that the RSNE fields other than the Length field, the PMKID Count field and the PMKID List field in the (Re)Association Request frame is identical to the RSNE fields included in the first Authentication frame. The EPP AP MLD shall also verify that the RSNXE in the (Re)Association Request frame is identical to the RSNXE included in the first Authentication frame. If the validation fails, the EPP AP MLD shall reject the association.

If the FT protocol is not used and the FT initial mobility domain association is not used, the RSNE fields corresponding to each link included in the (Re)Association Response frame shall be identical to the corresponding RSNE fields of the link in the Beacon or Probe Response frames from the corresponding AP affiliated with the EPP AP MLD or in the multi-link probe response received from the EPP AP MLD. (#2335)

If the FT initial mobility domain association is used, then EPP AP MLD shall include the MDE, the FTE, TIE[ReassociationDeadline], and TIE[KeyLifetime] in the (Re)Association Response frame. The MDE and the FTE shall be the same as the ones in the second Authentication frame as described in 12.16.8.1a (FT initial mobility domain association). The reassociation deadline timeout is set to the minimum of dot11FTReassociationDeadline and the key lifetime in the TIE[ReassociationDeadline], and the PTK lifetime in the TIE[KeyLifetime].

If the FT initial mobility domain association is used, then the PMKR1Name shall be included in the RSNE in the (Re)Association Response frame. The PMKR1Name shall be as calculated by the R1KH according to the procedures of 12.7.1.6.4 (PMK-R1). The RSNE fields other than the Length field, the PMKID Count field and the PMKID List field corresponding to each link shall be identical to the corresponding RSNE fields of the link in the Beacon or Probe Response frames (#2281) from the corresponding AP affiliated with the EPP AP MLD or in the multi-link probe response received from the EPP AP MLD. (#2335)

If the FT protocol is used, then the EPP AP MLD shall not wrap the Key field of the subelements in the FTE in the Reassociation Response frame and shall not calculate the MIC for the MIC field of the FTE. The length of the MIC field shall be 0 (i.e., the MIC Length subfield of the MIC Control field of the FTE is set to 3). The Element Count subfield of the MIC Control field of the FTE shall be set to 0.

If the FILS authentication protocol is used, then the EPP AP MLD shall not encrypt the (Re)Association Response frame using the AEAD algorithm as described in 12.11.2.6.3 ((Re)Association Response for FILS key confirmation).

The EPP AP MLD shall encrypt the (Re)Association Response frame transmitted to the EPP non-AP MLD in response to the (Re)Association Request frame using the TK and the pairwise cipher indicated in the Authentication frame exchange.

If the FILS authentication protocol and the FT protocol are not used, the EPP AP MLD shall include a Key Delivery element in the (Re)Association Response frame.

If a Key Delivery element is included in the (Re)Association Response frame, the EPP AP MLD shall construct the Key Delivery element with the RSC field set to 0, with the MLO GTK KDE for each setup link, with the MLO IGTK KDE for each setup link if management frame protection is negotiated, with the MLO BIGTK KDE for each setup link if beacon protection is enabled, with the PGTK KDE if the Group EPP Epoch Supported field in the RSNXE is set to 1 by both the APs affiliated with the AP MLD and the non-AP MLD, and with the Identity Key KDE if the AP MLD is a BPE AP MLD.

The EPP non-AP MLD shall decrypt the (Re)Association Response frame received from the EPP AP MLD using the TK and the pairwise cipher indicated in the Authentication frame exchange. If there is no output from the decryption algorithm because of unsuccessful MIC check (see 12.5.2.4.2 (CCM recipient processing) and 12.5.4.4.2 (GCM recipient processing)), the EPP non-AP MLD shall discard the frame.

If the FT protocol is not used, the FT initial mobility domain association is not used, and in the (Re)Association Response frame the RSNE fields corresponding to each link are not identical to the corresponding RSNE fields of the link in the Beacon or Probe Response frames (#2281) from the corresponding AP affiliated with the EPP AP MLD or in the multi-link probe response received from the EPP AP MLD, the EPP non-AP MLD shall discard the response.

If the FT protocol is not used, the EPP non-AP MLD shall verify that the RSNXE corresponding to each link in the (Re)Association Response frame is identical to the corresponding RSNXE of the link in the Beacon or Probe Response frames (#2281) from the corresponding AP affiliated with the EPP AP MLD or in the multi-link probe response received from the EPP AP MLD. If those frames did not include the RSNXE or if the RSNXEs are not identical, the EPP non-AP MLD shall discard the response.

If the FT initial mobility domain association is used, then the FTE and the MDE in the (Re)Association Response frame are checked to be the same as those provided in the second Authentication frame as defined in 12.16.8.3 (FT initial mobility domain association). If the check fails, the EPP non-AP MLD shall discard the response.

If the FT initial mobility domain association is used, then the PMKR1Name in the RSNE in the (Re)Association Response frame is checked to be included and identical to the value that is sent in the (Re)Association Request frame. If the check fails, the EPP non-AP MLD shall discard the response.

If the FT initial mobility domain association is used and in the (Re)Association Response frame the RSNE fields other than the Length field, the PMKID Count field and the PMKID List field corresponding to each link are not identical to the corresponding RSNE fields of the link in the Beacon or Probe Response frames (#2281) from the corresponding AP affiliated with the EPP AP MLD or in the multi-link probe response received from the EPP AP MLD, the EPP non-AP MLD shall discard the response.

If IEEE 802.1X Authentication utilizing Authentication frame is used, and the RSN capabilities fields of the RSNE received in the (Re)Association Response frame is not identical to the RSN capabilities fields of the RSNE received in the second Authentication frame, the EPP non-AP MLD shall discard the response.

On successful (re)association,

* The EPP non-AP MLD shall process the Key Delivery element in the (Re)Association Response frame if present.
* The EPP non-AP MLD shall install the GTK and GTK RSC, and IGTK and IGTK RSC if management frame protection is enabled, and BIGTK and BIGTK RSC if present in the Key Delivery element and dot11BeaconProtectionEnabled is true, and PGTK if the Group EPP Epoch Supported field in the RSNXE is set to 1 by both the APs affiliated with the AP MLD and the non-AP MLD, and identity key if the AP MLD is a BPE AP MLD.
* The EPP AP MLD and the EPP non-AP MLD shall transition to State 4 (as defined in 11.3 (STA authentication and association)).
* If the DS MAC Address element is included in the (Re)Association Request frame, the EPP non-AP MLD shall use the indicated DS MAC address rather than the MLD MAC address of the non-AP MLD for the EPP non-AP MLD to the EPP AP MLD mapping to the DS.
* If the DS MAC Address element is included in the (Re)Association Request frame, the EPP AP MLD shall process the DS MAC Address element and use the indicated DS MAC address rather than the MLD MAC address of the EPP non-AP MLD to establish the EPP non-AP MLD to the EPP AP MLD mapping to the DS.

NOTE 3—If the DS MAC Address element is included in the (Re)Association Request frame, the source address or destination address parameters of the MAC service tuples (see 5.2.4.2 (Semantics of the service primitive)) for the EPP non-AP MLD are set to the DS MAC address, which is the identity of the non-AP MLD known by the DS.

On failed (re)association, the established PTKSA shall be irretrievably deleted.

**TGbi Editor: *Instruction: Modify 4.5.4.2 as follows***

* Authentication

***Change the third, fourth, sixth paragraph as follows:***

IEEE Std 802.11 defines the following ~~five~~ IEEE 802.11 authentication methods: ~~Open System authentication, FT authentication, simultaneous authentication of equals (SAE), FILS authentication, and preassociation security negotiation (PASN) authentication.~~

* Open System authentication admits any STA to the DS.
* FT authentication relies on keys derived during the initial mobility domain association to authenticate the stations as defined in Clause 13 (Fast BSS transition).
* SAE authentication uses finite field cryptography to prove knowledge of a shared password.
* IEEE 802.1X authentication uses EAP to authenticate STAs and the AS with one another.
* FILS authentication allows for faster connection to the network for FILS non-AP STAs by providing authentication, association, and key confirmation information in an efficient number of frame exchanges (see 4.10.3.6 (AKM operations using FILS authentication)).
* PASN authentication allows for the protection of Management frames without association by establishing a PTKSA using authentication frames.
* EPPKE authentication allows for the protection of (Re)Association Request/Response frame by establishing a PTKSA using authentication frames.

The IEEE 802.11 authentication mechanism also allows definition of new authentication methods, or any combination of these authentication methods.

An RSNA might support one or more of the following authentication methods: SAE authentication, IEEE 802.1X authentication, FILS authentication, ~~or~~ PASN authentication, or EPPKE authentication. An RSNA also supports authentication based on IEEE Std 802.1X-2020, or preshared keys (PSKs) after Open System authentication. ~~IEEE 802.1X authentication utilizes the EAP to authenticate STAs and the AS with one another.~~ This standard does not specify an EAP method that is mandatory to implement for authentication based on IEEE Std 802.1X-2020(#2465). See 12.6.4 (RSNA policy selection in an IBSS) for a description of the IEEE 802.1X authentication and PSK usage within an IEEE 802.11 IBSS.

SAE authentication, IEEE 802.1X authentication, and Open System IEEE 802.11 authentication are used by STAs in an RSN for an infrastructure BSS. FILS authentication can be used by FILS STAs in an RSN for an infrastructure BSS. SAE authentication, Open System IEEE 802.11 authentication, or no IEEE 802.11 authentication is used in an RSN for an IBSS. SAE authentication is used for an MBSS. In an RSN for DMG BSS, Open System IEEE 802.11 authentication is not used (12.2.4 (RSNA establishment)).

***Change the last paragraph as follows:***

PASN authentication or EPPKE authentication is used in an RSN for an infrastructure BSS when it is based on a PMKSA established by another RSN authentication protocol. Otherwise, it does not guarantee mutual authentication, and can be used as a non-RSN protocol in an infrastructure BSS.

**TGbi Editor: *Instruction: Modify 9.3.3.5 as follows***

* Association Request frame format

***Insert new rows to Table 9-64 (Association Request frame body) in numeric order (not all lines shown):***

* revme D7.0 up to 60, 11bh D6.0 up to 61-62, 11be D7.0 63-65, 11bk D5.0 no addition, 11bf D8.0 66-69

|  |  |  |
| --- | --- | --- |
| * Association Request frame body (continued) | | |
| Order | Information | Notes |
| … |  |  |
| 11 | Mobility Domain | The MDE is present in an Association Request frame if dot11FastBSSTransitionActivated is true and if the frame is being sent to an AP that advertised its FT capability in the MDE in its Beacon or Probe Response frame (i.e., AP also has dot11FastBSSTransitionActivated equal to true).  dot11FastBSSTransitionActivated is true, encryption of the Association Request frame is used, and if the frame is being  sent to an AP that advertised its FT capability in the MDE in its  Beacon or Probe Response frame (i.e., AP also has  dot11FastBSSTransitionActivated equal to true). |
| … |  |  |
| 70 | DS MAC Address | The DS MAC Address element is present if the Association Request frame is encrypted, dot11DSMACAddressActivated is true, and the peer indicates support for DS MAC Address in the RSNXE; otherwise, it is not present. |
| 71 | EPP | The EPP element is optionally present if the Association Request frame is encrypted and dot11EPPGroupEpochActivated is true; otherwise, it is not present. |
|  |  | (#2469) |
| 72(#2469) | Fast BSS Transition | An FTE is present in an Association Request frame if  dot11FastBSSTransitionActivated is true, encryption of the Association Request frame is used, and  dot11RSNAAuthenticationSuiteSelected is equal to an AKM suite  selector value for which the Authentication type column indicates  FT authentication. See Table 9-190 (AKM suite selectors) (i.e.,  part of a fast BSS transition in an RSN). |
| 73(#2469) | Nonce | The Nonce element is optionally present as defined in 12.16.7.1 (PMKID privacy); otherwise, it is not present. |

**TGbi Editor: *Instruction: Modify 12.16.4.2 as follows***

**12.16.4.2 MLO procedure**

(…existing texts…)

If APs affiliated with an AP MLD set the EPP Capabilities And Operation Parameters Request/Response Support field in the RSNXE to 1 and the AP MLD receives through a setup link from an associated non-AP MLD an individually addressed EPP Capabilities And Operation Parameters Request frame with a Basic Multi-Link element, then the AP MLD shall respond with an individually addressedEPP Capabilities And Operation Parameters Response frame through an affiliated AP over an enabled(#2488) link to the non-AP MLD.

(…existing texts…)