IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolutions to CIDs related to Protected WUR frames | | | | |
| Date: 2019-05-10 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Yunsong Yang | Huawei Technologies | 10180 Telesis Court, STE 400, San Diego, CA 92121 | +1-858-754-3638 | yangyunsong@huawei.com |

Abstract

This contribution proposes resolutions and spec text changes to resolve 18 LB237 comments (with the following CIDs) related to the Protected WUR frames:

* 2318, 2333, 2334, 2335, 2336, 2337, 2341, 2350, 2418, 2419, 2421, 2522, 2555, 2576, 2578, 2579, 2823, 2824

Revisions:

* Rev 0: Initial version of the document.

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 TGba Draft. This introduction is not part of the adopted material.

***Editing instructions formatted like this are intended to be copied into the TGba Draft (i.e. they are instructions to the 802.11 editor on how to merge the text with the baseline documents).***

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 2418 | 76.52 | Protected WUR frames are dependent on Management Frame Protection being negotiated between the WUR STA and the WUR AP | Change "A WUR AP may transmit..." to "When management frame protection is negotiated through the association of the WUR STA to the WUR AP, a WUR AP shall transmit..." | **Revised.**  Agree with the commenter that protected WUR frames are dependent on Management Frame Protection being negotiated. Following the style of management frame protection, add, under 30.9, a new second paragraph describing when WUR frame protection is enabled, a new third paragraph describing when WUR frame protection is negotiated, and a new sixth paragraph describing behaviors before the associated keys are installed. The original first and second paragraphs are simplified and become the fourth and fifth paragraphs, respectively.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2418. |
| 2419 | 76.76 | WUR frame protection can only occur when an RSN security association is established. | State that there is a requirement for a RSA assocaition to be established. | **Revised.**  Agree with the commenter that WUR frame protection can only occur when an RSNA is established. PTKSA is modified to support the protection of indicvidially addressed WUR frames and WIGTKSA is introduced to support the protection of broadcast and group addressed WUR frames. Add a new first paragraph under 30.9 to state that as a high level requirement. Detailed changes are made to 12.6.1 Security associations.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2419. |

**Discussion:**

CID 2418: Agree with the commenter that protected WUR frames are dependent on Management Frame Protection being negotiated. Following the style of management frame protection, add, under 30.9, a new second paragraph describing when WUR frame protection is enabled, a new third paragraph describing when WUR frame protection is negotiated, and a new sixth paragraph describing behaviors before the associated keys are installed. The original first and second paragraphs are simplified and become the fourth and fifth paragraphs, respectively.

CID 2419: Agree with the commenter that WUR frame protection can only occur when an RSNA is established. PTKSA is modified to support the protection of indicvidially addressed WUR frames and WIGTKSA (similar to IGTKSA) is introduced to support the protection of broadcast and group addressed WUR frames. Add a new first paragraph under 30.9 to state that as a high level requirement. Detailed changes are made to 12.6.1 Security associations.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 2341 | 59.31 | If the intention of clause 31.8 is to leverage existing IEEE 802.11 encapsulation mechanisms, these clauses should be moved to Clause 12. Placing clauses at this location makes it difficult to evaluate whether these mechanisms introduce security vulnerabilities. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Rejected.**  The commenter suggests to move certain text from 30.9 (31.8 in D1.0) to Clause 12. However, the commenter didn’t identify any technical issues in the text of the cited subcaluse, other than where to place those text. Both Clauses 12 and 30 are for normative text. There is no evidence that the current placements of those text under 30.9 cause any difficulty in evaluating the security mechanism. Besides, protected WUR frames don’t exactly use the BIP encapsulation. The MIC simply replaces the CRC in the FCS field (see 9.10.2.5). |

**Discussion:**

The commenter suggests to move certain text from 30.9 (31.8 in D1.0) to Clause 12. However, the commenter didn’t identify any technical issues in the text of the cited subcaluse, other than where to place those text. Both Clauses 12 and 30 are for normative text. There is no evidence that the current placements of those text under 30.9 cause any difficulty in evaluating the security mechanism. Besides, protected WUR frames don’t exactly use the BIP encapsulation. The MIC simply replaces the CRC in the FCS field (see 9.10.2.5).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 2350 | 32.00 | It would be good if the WUR protection capability of a WUR STA is verified during Message 2 and 3 of the 4-way handshake used to negotiate WUR TK (as is done for management frame protection - 802.11-2016-P1193 last paragraph). The "Protection Support" bit then is better shifted/copied to the RSN capabilities field of the RSNE. Otherwise, the WUR capabilities element may need to be carried in Message 2 and 3 of the 4-way handshake just for this one bit. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  Text are added in clause 12.6 and 30.9 such that WUR frame protection becomes negotiated only after the WUR Frame Protection Support bit from the WUR AP and WUR non-AP STA is verified in the 4-way handshake, FT 4-way handshake, or reassociation procedures.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2350. |
| 2576 | 42.53 | It would be good if the WUR protection capability of a WUR STA is verified during Message 2 and 3 of the 4-way handshake used to negotiate WUR TK (as is done for management frame protection - 802.11-2016-P1193 last paragraph). The "Protection Support" bit then is better shifted/copied to the RSN capabilities field of the RSNE. Otherwise, the WUR capabilities element may need to be carried in Message 2 and 3 of the 4-way handshake just for this one bit. | Either move or create a copy of the "Protection Support" bit in the RSN capabilities field of the RSNE. | **Revised.**  Text are added in clause 12.6 and 30.9 such that WUR frame protection becomes negotiated only after the WUR Frame Protection Support bit from the WUR AP and WUR non-AP STA is verified in the 4-way handshake, FT 4-way handshake, or reassociation procedures.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2376. |

**Discussion:**

Agree with the commenters that the cited bit should be verified during the 4-way handshake. However, there is only one reserved bit left in the RSN Capabilities field in the RSNE. On the other hand, including the entire WUR Capabilities element in the 4-way handshake is inefficient. Currently, TGm is evaluating two possible ways of verifying a TWT protection bit, as well as large number of future security related capabilities bits, during the 4-way handshake. One is to include the Extended Capabilites element during the 4-way handshake. The other is to include a new RSN Extension element during the 4-way handshake. Once TGm finalizes the element to use, it is trivial to include the cited bit in that element in 11ba draft for verification purpose. The current text describes that bit as in the <Extended Capabililites element or RSN Extension element>, pending the decision in TGm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CID** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 2318 | 61.31 | Given that the WUR connection between the STA and the AP are a separate communications connection for a separate purpose (WUR), WUR should use a separately derived key. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  Details of the new WTK (a new pairwise key derived as part of PTK for protecting individually addressed WUR Wake-up frames) and WIGTK (a new group key similar to IGTK for protecting broadcast and group addressed WUR Wake-up frames) are added in clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13. |
| 2333 | 59.60 | The separate WUR TK is not specified in this draft nor in the baseline document. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  Details of the WTK (a new pairwise key derived as part of PTK for protecting individually addressed WUR Wake-up frames) are added in clauses 3.2, 3.4, 4.10, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 3.2, 3.4, 4.10, 12, and 13. |
| 2334 | 59.59 | The separate WUR IGTK is not specified in this draft nor in the baseline document. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  Details of the WIGTK (a new group key similar to IGTK for protecting broadcast and group addressed WUR Wake-up frames) are added in clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13. |
| 2335 | 59. 58 | It should be clarified when exactly does the negotiation of integrity keys (WUR IGTK) and (pairwise WUR TK) takes place? Is it right after Association, or does it only happen after a successful WUR mode setup? Adding a figure of the handshaking used to exchange WUR keys would be very helpful. | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  The WTK and WIGTK are derived during 4-way or FT 4-way handshake. Details are added in clauses 4.10, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 4.10 12, and 13. |
| ,2336 | 59. 58 | It is not clear how WUR IGTK and WUR TK are negotiated. Does that require a separate handshake? | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  The WTK and WIGTK are derived during 4-way or FT 4-way handshake. Details are added in clauses 4.10, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 4.10 12, and 13. |
| 2337 | 59. 58 | Similar to management frame protection, WUR frame protection should also be added to the RSNA establishment procedure | Picking up on comments made in the previous letter ballot on D1.0, the TG did not properbly address the issue raised in the comment, nor does the TG provide an indication that the text commented on has been deleted and hence the comment does not apply. (Note, page and line and sublause number refer to D1.0). In fact, as stated in the TGba minutes (11-19/226r0), the intend of the task group was to "Move to resolve CIDs that have no approved resolution as rejected with a reason read "TGba is unable to reach consensus on a resolution" in the interest of releasing draft 2.0". Also, the statement ""TGba is unable to reach consensus on a resolution" was added to the motion text there was one person speaking against the motion." was only added to the motion after objection to the original motion trying to reject comments in bulk with the reason of releasing a new LB.  The TG is asked to give the original comment due consideration and debade the proposed comment resolution as included in 11-18/1794r10. The referenced document includes an actionable comment resolution. | **Revised.**  In subclause 12.2.4 (RSNA establishment), add a new sub-bullet to the end of both bullets a) and b) that reads: “If the STAs negotiate WUR frame protection, the SME programs the WTK and WTPN into the MAC for protection of individually addressed WUR Wake-up frames, and programs the WIGTK and WIPN into the MAC for the protection of broadcast and group addressed WUR Wake-up frames.”  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2337. |
| 2421 | 76.52 | A BIP key is derived and bound to group-addressed communicaions in the BSS. A PTK is bound to a security associiation between a STA and an AP in a BSS. Using the same keys opens new possible attacks on BSS security. To protect WUR frames, new keys need to be derived. They can be derived in a similar manner to PTK and BIP, but they shoudl be separate keys. | Provide new key derivations so that WUR protection is not using keying material for BSS operations. | **Revised.**  Details of the new WTK (a new pairwise key derived as part of PTK for protecting individually addressed WUR Wake-up frames) and WIGTK (a new group key similar to IGTK for protecting broadcast and group addressed WUR Wake-up frames) are added in clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 3.2, 3.4, 4.10, 6.3, 9.4.2, 11.3, 12, and 13. |
| 2522 | 77.08 | For WUR IGTK, WUR AP may only update some STAs to the new WUR IGTK with new key ID, and some STAs still use old WUR IGTK with old key ID. As a reuslt, the definition of current Key ID is different for different STAs. | Based on the existing usage of key ID, 6 or 7 can be used for WUR IGTK. The usage of existing key ID is described in the following phrase "0-3 shall be used with WEP, TKIP, CCMP, and GCMP; 4-5 with BIP; and 6-4095 are reserved". Have an indication in broadcast WUR frame to indicate current key ID. Most likely, the indication will be in misc field. Note that the indication can be just one bit. For example, if Key ID 6 or 7 is used for WUR IGTK, then 0 can mean 6, and 1 can mean 7. | **Revised.**  Key ID values of 8-9 are added for WIGTK in clauses 6.3.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2522. |
| 2555 | 77.08 | I wonder if we need more description for the following phrase "that is negotiated as defined in 12.7.7 (Group key handshake)". Specifically, to use 12.7.7, we need a key ID for the WUR IGTK based on the texts in 12.7.7. To have a key ID, we need to define WUR IGTK KDE in table 12-7. | Define WUR IGTK KDE in table 12-7 probably following the format of Figure 12-42--IGTK KDE format. | **Revised.**  New WIGTK KDE is added in Table 12-7. Format and text description of it, similar to the ones of IGTK, are added to the end of clause 12.7.2.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2555. |
| 2578 | 77.07 | It should be clarified when exactly does the negotiation of integrity keys (WUR IGTK) and (pairwise WUR TK) takes place? Is it right after Association, or does it only happen after a successful WUR mode setup? Adding a figure of the handshaking used to exchange WUR keys would be very helpful. | as in comment | **Revised.**  The WTK and WIGTK are derived during 4-way or FT 4-way handshake. Details are added in clauses 4.10, 12, and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 4.10 12, and 13. |
| 2579 | 77.07 | Similar to management frame protection, WUR frame protection should also be added to the RSNA establishment procedure | Add text in each relevant sections of 12.2.4 (RSNA establishment) to specify the steps taken by a STA to program the WUR TK and WUR IGTK used to protect WUR frames: "If the STAs negotiate protection of WUR frames, the SME programs the WUR TK and pairwise cipher suite into the WUR MAC for protection of individually addressed WUR frames. It also installs the WUR IGTK and IPN for protection of broadcast and group addressed WUR frames. | **Revised.** In subclause 12.2.4 (RSNA establishment), add a new sub-bullet to the end of both bullets a) and b) that reads: “If the STAs negotiate WUR frame protection, the SME programs the WTK and WTPN into the MAC for protection of individually addressed WUR Wake-up frames, and programs the WIGTK and WIPN into the MAC for the protection of broadcast and group addressed WUR Wake-up frames.”  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2579. |
| 2823 | 77.08 | Text to amend 12.7.7 for the WUR IGTK is missing. | Add amendment text in 12.7.7 that specifies WUR IGTK. | **Revised.**  Details of the WIGTK is added in clauses 12 and 13.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under clauses 12 and 13. |
| 2824 | 77.10 | Text to amend 12.7.6 for the WUR TK is missing. | Add amendment text in 12.7.7 that specifies WUR TK. | **Revised.**  Details of the WTK is added in 12.7.1.3, 12.7.1.6.5, and 12.12.2.5.3.  TGba Editor: make changes as shown in doc. 11-19-0761r0 under all headings that include CID #2824. |

**Discussion:**

This group of comments essentially requests new security keys be defined for WUR, or points out missing details about these new keys, namely the WTK and WIGTK, as to security association, key derviation, distribution, procedures and operations associated with them.

As a high level summary, a WTK is derived as a part of the PTK during 4-way handshake, by extending the key length, and is used for protecting individually addressed WUR Wake-up frames. A WIGTK is generated by the Authenticator and distributed to Supplicants in a similar way as the IGTK today (So, text describing the WIGTK largely follows the style of simualr text for the IGTK), and is used for protecting broadcast and group addressed WUR Wake-up frame.

The requested details are added in clause 4.10, 6.3, 9.4.2, 11.3, 12, and 13. New definitions and abbreviations are added in 3.2 and 3.4.

The spec text begins from the next page.

***TGba editor: insert the following definitions in subclause 3.2, maintaining alphabetical order*** (#2318, #2333, #2334, #2421)

**wake-up radio (WUR) integrity group temporal key (WIGTK):** A random value, assigned by a WUR access point (AP), that is used to protect broadcast and group addressed WUR frames from that AP.

**Wake-up radio (WUR) temporal key (WTK):** A temporal key used to protect individually addressed WUR Wake-up frames.

***TGba editor: change subclause 3.4 in P802.11ba D2.1, as follows:*** (#2318, #2333, #2334, #2421)

…

WIGTK wake-up radio integrity group temporal key

WIGTKSAwake-up radio integrity group temporal key security association

WIPN wake-up radio IGTK packet number

WTK wake-up radio temporal key

WTPN wake-up radio TK packet number

***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 4, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421, #2333, #2335, #2336, #2578)

**4.5 Overview of the services**

**4.5.4 Access control and data confidentiality services**

**4.5.4.3 Deauthentication**

***Instruct the editor to change the 4th paragraph of 4.5.4.3 Deauthentication as follows:***

In an RSNA, deauthentication also deletes any related pairwise transient key security association (PTKSA), group temporal key security association (GTKSA), TPK security association (TPKSA), integrity group temporal key security association (IGTKSA)), beacon integrity group temporal key security association (BIGTKSA), wake-up radio integrity group temporal key security association (WIGTKSA), mesh GTKSA, and mesh TKSA that exist in the STA and closes the associated IEEE 802.1X Controlled Port, if used for this association. If pairwise master key security association (PMKSA) caching is not enabled, deauthentication also deletes the PMKSA or mesh PMKSA.

**4.10 IEEE Std 802.11 and IEEE Std 802.1X-2010**

**4.10.3 Infrastructure functional model overview**

**4.10.3.2 AKM operations with AS**

***Instruct the editor to change 4.10.3.2 AKM operations with AS as follows:***

…

A 4-way handshake or FT 4-way handshake utilizing EAPOL-Key frames is initiated by the Authenticator to do the following:

* Confirm that a live peer holds the PMK.
* Confirm that the PMK is current.
* In the case of fast BSS transition, derive PMK-R0s and PMK-R1s.
* Derive a fresh pairwise transient key (PTK) from the PMK or, in the case of fast BSS transition, from the PMK-R1, the derived PTK including the WTK if WUR frame protection is negotiated.
* Install the pairwise encryption and integrity keys.
* Transport the group temporal key (GTK) and GTK sequence number from Authenticator to Supplicant and install the GTK and GTK sequence number in the STA and, if not already installed, in the AP.
* If management frame protection is negotiated, transport the IGTK and the IGTK packet number (IPN) from the Authenticator to the Supplicant and install these values in the STA and, if not already installed, in the AP.
* If beacon protection is enabled, transport the BIGTK and the BIGTK packet number (BIPN) from the Authenticator to the Supplicant and install these values in the STA and, if not already installed, in the AP.
* If WUR frame protection is negotiated, transport the WIGTK and the WIGTK packet number (WIPN) from the Authenticator to the Supplicant and install these values in the STA and, if not already installed, in the AP.
* Verify that the RSN capabilities negotiated are valid as defined in 9.4.2.24.4 (RSN capabilities).
* Confirm the cipher suite selection.

Installing the PTK, and where applicable the GTK and, if management frame protection is negotiated, the IGTK, causes the MAC to encrypt and decrypt all subsequent MSDUs irrespective of their path through the controlled or uncontrolled ports. Installing the PTK when WUR frame protection is negotiated also causes the MAC to integrity protect subsequent individually addressed WUR Wake-up frames at the AP or to validate subsequent individually addressed WUR Wake-up frames at the non-AP STA. Installing the BIGTK when beacon protection is enabled causes the MAC to integrity protect at the AP or validate subsequent Beacon frames at the non-AP STA. Installing the WIGTK when WUR frame protection is negotiated causes the MAC to integrity protect subsequent broadcast or group addressed WUR Wake-up frames at the AP or to validate subsequent broadcast or group addressed WUR Wake-up frames at the non-AP STA.

…

When management frame protection is negotiated, the Authenticator also uses the group key handshake with all associated STAs to change the IGTK, BIGTK if beacon protection is enabled, and WIGTK if WUR frame protection is negotiated. The Authenticator encrypts the GTK, IGTK, BIGTK, and WIGTK values in the EAPOL-Key frame as described in 12.7 (Keys and key distribution).

**4.10.3.3 AKM operations with a password or PSK**

***Instruct the editor to change 4.10.3.3 AKM operations with a password or PSK as follows:***

…

* The GTK and GTK sequence number are sent from the Authenticator to the Supplicant just as in the AS case. See Figure 4-32 (Establishing pairwise and group keys) and Figure 4-33 (Delivery of subsequent group keys).
* If WUR frame protection is negotiated, the WIGTK and WIPN are sent from the Authenticator to the Supplicant just as in the AS case. See Figure 4-32 (Establishing pairwise and group keys) and Figure 4-33 (Delivery of subsequent group keys).

**4.10.3.4 Alternate operations with PSK**

***Instruct the editor to change the last bullet of 4.10.3.4 Alternate operations with PSK as follows:***

…

* If management frame protection is negotiated, the IGTK and IGTK packet number are sent from the Authenticator to the Supplicant just as in the AS case. If beacon protection is enabled, the BIGTK and BIPN are sent from the Authenticator to the Supplicant just as in the AS case. If WUR frame protection is negotiated, the WIGTK and WIPN are sent from the Authenticator to the Supplicant just as in the AS case. See Figure 4-32 (Establishing pairwise and group keys) and Figure 4-33 (Delivery of subsequent group keys).

***TGba Editor: in P802.11ba draft, add the following instruction and Figure 4-32, after replacing “IGTK and BIGTK” with “IGTK, BIGTK, and WIGTK”, replacing “IGTK, BIGTK” with “IGTK, BIGTK, WIGTK”, and replacing “and BIGTK” with “BIGTK, and WIGTK” in the figure to the following effects:***

***…***

***Derive PTK***

***if needed***

***Generate GTK, IGTK, BIGTK, and WIGTK]***

***...***

***Encrypted (GTK, IGTK, BIGTK)***

***…***

***Install PTK, GTK, IGTK, BIGTK, and WIGTK [two boxes - Supplicant and Authenticator]*** (#2318, #2334, #2421)

**4.10.3.3 AKM operations with a password or PSK**

***Instruct the editor to replace Figure 4-32 Establishing pairwise and group keys with the figure as follows***



***TGba Editor: in P802.11ba draft, add the following instruction and Figure 4-33, after replacing “IGTK, and BIGTK” with “IGTK, BIGTK, and WIGTK”, replacing “IGTK, BIGTK” with “IGTK, BIGTK, WIGTK”, replacing “Encrypted BIGTK” with “Encrypted BIGTK, Encrypted WIGTK” in the figure to the following effects:***

***…***

***Generate GTK, IGTK, BIGTK, WIGTK***

***Encrypt GTK, IGTK, BIGTK, WIGTK with KEK***

***…***

***Message 1: EAPOL-Key (Encrypted GTK, Encrypted IGTK, Encrypted BIGTK, Encrypted WIGTK, Group, MIC)***

***....***

***Install GTK, IGTK, BIGTK, and WIGTK*** (#2318, #2334, #2421)

***Instruct the editor to replace Figure 4-33 Delivery of subsequence group keys with the figure as follows:***



***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 6, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421)

**6.3.19 SetKeys**

**6.3.19.1 MLME-SETKEYS.request**

**6.3.19.1.2 Semantics of the service primitive**

***Instruct the editor to change rows “Key ID”, “Key Type”and “Receive Sequence Count” in the SetKeyDescriptor*** ***table of 6.3.19.1.2 as follows:*** (#2522)

|  |  |  |  |
| --- | --- | --- | --- |
| Key ID | Integer | 0–3 shall be used  with WEP, TKIP,  CCMP, and  GCMP;  4–5 with BIP for IGTK; 6–7 with BIP for BIGTK; 8-9 with BIP for WIGTK; 10 with BIP for WTK; and 11–4095 are  Reserved(#2522) | Key identifier |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK, BIGTK, WIGTK | Defines whether this key is a group key, pairwise key, PeerKey, Integrity Group key, beacon integrity group temporal key, or wake-up radio integrity group temporal key. |

|  |  |  |  |
| --- | --- | --- | --- |
| Receive Sequence Count | 8 octets | N/A | Value to which the RSC(s) is initialized.  This parameter is valid only when the Key Type is Group, IGTK, BIGTK, or WIGTK. |

**6.3.19.1.4 Effect of receipt**

***Instruct the editor to change 6.3.19.1.4 as follows:***

When the Key Type is Group, IGTK, BIGTK, or WIGTK, and the key matches the corresponding existing GTK, IGTK, BIGTK, or WIGTK, installed as a result of exiting WNM sleep mode (see 11.2.3.16.1 (WNM sleep mode capability)), if any, or matches the existing GTK or IGTK installed as a result of receipt of EAPOL-Key frames (see 12.7.7.4 (Group key handshake implementation considerations)) receipt of this primitive shall have no effect. Otherwise, receipt of this primitive causes the MAC to apply the keys as follows, subject to the MLME-SETPROTECTION.request primitive:

...

* When the Key, Address, Key Type, and Key ID parameters identify a new key to be set, the MAC initializes the transmitter TSC/PN/IPN/BIPN/WIPN/WTPN counter to 0. When the Key, Address, Key Type, and Key ID parameters identify an existing key, the MAC shall not change the current transmitter TSC/PN/IPN/BIPN/WIPN/WTPN counter or the receiver replay counter values associated with that key.

**6.3.20 DeleteKeys**

**6.3.20.1 MLME-DELETEKEYS.request**

**6.3.20.1.2 Semantics of the service primitive**

***Instruct the editor to change rows “Key ID” and “Key Type” in the DeleteKeyDescriptor table of 6.3.20.1.2 as follows:*** (#2522)

|  |  |  |  |
| --- | --- | --- | --- |
| Key ID | Integer | 0–3 shall be used  with WEP, TKIP,  CCMP, and  GCMP;  4–5 with BIP for IGTK; 6–7 with BIP for BIGTK; 8-9 with BIP for WIGTK; 10 with BIP for WTK; and 11–4095 are  Reserved(#2522) | Key identifier |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK, BIGTK, WIGTK | Defines whether this key is a group key, pairwise key, PeerKey, Integrity Group key, beacon integrity group temporal key, or wake-up radio integrity group temporal key. |

**6.3.23 SetProtection**

**6.3.23.1 MLME-SETPROTECTION.request**

**6.3.23.1.2 Semantics of the service primitive**

***Instruct the editor to change row “Key Type” in the ProtectKeyDescriptor table of 6.3.23.1.2 as follows:***

|  |  |  |  |
| --- | --- | --- | --- |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK, BIGTK, WIGTK | Defines whether this key is a group key, pairwise key, PeerKey, Integrity Group key, beacon integrity group temporal key, or wake-up radio integrity group temporal key. |

**6.3.94 PN event report**

**6.3.94.2 MLME-PN-EXHAUSTION.indication**

**6.3.94.2.2 Semantics of the service primitive**

***Instruct the editor to change row “Key Type” in the table in 6.3.94.2.2 (semantics of service for PN exhaustion) as follows:***

|  |  |  |  |
| --- | --- | --- | --- |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK, BIGTK, WIGTK | Defines whether this key is a group key, pairwise key, PeerKey, Integrity Group key, beacon integrity group temporal key, or wake-up radio integrity group temporal key. |

**6.3.94.3 MLME-PN-WARNING.indication**

**6.3.94.3.2 Semantics of the service primitive**

***Instruct the editor to change row “Key Type” in the table in 6.3.94.3.2 (semantics of service primitive for PN warning) as follows:***

|  |  |  |  |
| --- | --- | --- | --- |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK, BIGTK, WIGTK | Defines whether this key is a group key, pairwise key, PeerKey, Integrity Group key, beacon integrity group temporal key, or wake-up radio integrity group temporal key. |

***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 9, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421)

**9.4.2.24.2 Cipher suites**

***Instruct the editor to change 9.4.2.24.2 Cipher suites as follows:***

…

In an RSNA with management frame protection enabled, the cipher suite selector 00-0F-AC:6 (BIP-CMAC-128) is the default group cipher suite for Management frames when the Group Management Cipher Suite field is not included in the RSNE.

In an RSNA with WUR frame protection enabled, the cipher suite used for protecting individually addressed and broadcast and group addressed WUR Wake-up frames is BIP-CMAC-128 and is not explicitly indicated in the RSNE.

…

Cipher suite usage  indicates the circumstances under which each cipher suite is used.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * Cipher suite usage | | | | |
| Cipher suite selector | GTK | PTK | IGTK or BIGTK(#2116) | WIGTK |
| Use group cipher suite | No | Yes | No | No |
| WEP-40 | Yes | No | No | No |
| WEP-104 | Yes | No | No | No |
| TKIP | Yes | Yes | No | No |
| CCMP-128 | Yes | Yes | No | No |
| BIP-CMAC-128 | No | Yes only for WTK; No otherwise | Yes | Yes |
| GCMP-128 | Yes | Yes | No | No |
| GCMP-256 | Yes | Yes | No | No |
| CCMP-256 | Yes | Yes | No | No |
| BIP-GMAC-128 | No | No | Yes | No |
| BIP-GMAC-256 | No | No | Yes | No |
| BIP-CMAC-256 | No | No | Yes | No |

**9.4.2.47 Fast BSS Transition element (FTE)**

***Instruct the editor to insert a new row in Table 9-181 Subelement IDs in 9.4.2.47 Fast BSS Transition element (FTE) as follows, and adjust reserved values:***

|  |  |
| --- | --- |
| 7 | WIGTK |
| 8–255 | Reserved |

***Instruct the editor to add paragraphs at the end of 9.4.2.47 Fast BSS Transition element (FTE) as follows:***

The WIGTK subelement contains the WIGTK, used for protecting broadcast and group addressed WUR Wake-up frames. The WIGTK subelement format is shown in Figure 9-xxx (WIGTK subelement format).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Subelement ID | Length | Key ID | WIPN | Key Length | Wrapped Key |
| Octets: | 1 | 1 | 2 | 6 | 1 | 24 |

Figure 9-xxx WIGTK subelement format

The Key ID field indicates the value of the WIGTK identifier.

The WIPN field indicates the receive sequence counter for the WIGTK being installed, to allow a STA to identify replayed broadcast or group addressed WUR Wake-up frames.

The Key Length field is the length of the WIGTK in octets, not including any padding (see 13.8.5 (FT

authentication sequence: contents of fourth message)).

The Wrapped Key field contains the wrapped WIGTK being distributed. The length of the resulting AES-Key-wrapped WIGTK in the Wrapped Key field is Key Length + 8 octets.

***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 11, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421)

**11. MLME**

**11.3 STA authentication and association**

**11.3.4 Authentication and deauthentication**

**11.3.4.4 Deauthentication—originating STA**

***Instruct the editor to change the e bullet in 11.3.4.4 Deauthentication - originating STA, as follows:***

e) The SME, upon receipt of an MLME-DEAUTHENTICATE.confirm primitive, shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the indicated STA by using the MLME-DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by generating an MLME-SETPROTECTION.request(None) primitive.

**11.3.4.5 Deauthentication—destination STA**

***Instruct the editor to change the b bullet in 11.3.4.5 Deauthentication - destination STA - as follows:***

b) Upon receiving an MLME-DEAUTHENTICATE.indication primitive, the SME shall

1) Delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the originating STA by using the MLME-DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by generating an MLMESETPROTECTION.request(None) primitive.

**11.3.5 Association, reassociation, and disassociation**

**11.3.5.2 Non-AP and non-PCP STA association initiation procedures**

***Instruct the editor to change the first paragraph of 11.3.5.2 Non-AP and non-PCP STA association initiation procedures as follows:***

The SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the AP or PCP by using MLME-DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) before invoking MLME-ASSOCIATE.request primitive.

**11.3.5.3 AP or PCP association receipt procedures**

***Instruct the editor to change the l bullet in 11.3.5.3 AP or PCP STA association receipt procedures - as follows***

* If the ResultCode in the MLME-ASSOCIATE.response primitive is SUCCESS, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the STA by using the MLME-DELETEKEYS.request primitive (see 11.5.18 (RSNA security association termination)).

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

***Instruct the editor to change the first parapraph of 11.3.5.4 Non-AP or non-PCP STA reassociation initiation procedures - as follows:***

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

**11.3.5.5 AP or PCP reassociation receipt procedures**

***Instruct the editor to change the j bullet in 11.3.5.5 AP or PCP STA reassociation receipt procedures as follows:***

* If the ResultCode in the MLME-REASSOCIATE.response primitive is SUCCESS and the reassociation is not part of a fast BSS transition, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the STA by using the MLME-DELETEKEYS.request primitive (see 11.5.18 (RSNA security association termination)).

**11.3.5.6 Non-AP and non-PCP STA disassociation initiation procedures**

***Instruct the editor to change the d bullet in 11.3.5.6 Non-AP and non-PCP STA disassociation initiation procedures as follows:***

* Upon receiving an MLME-DISASSOCIATE.confirm primitive, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the AP or PCP by using the MLME‑DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by invoking an MLME-SETPROTECTION.request(None) primitive. In the case of an MM-SME coordinated STA, the MLME shall perform this for each STA whose address was included in the MMS parameter of the MLME-ASSOCIATE.request or MLME-REASSOCIATE.request primitive that established the association.

**11.3.5.7 Non-AP and non-PCP STA disassociation receipt procedure**

***Instruct the editor to change the c bullet in 11.3.5.7 Non-AP and non-PCP STA disassociation receipt procedures as follows:***

* Upon receiving the MLME-DISASSOCIATE.indication primitive, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the AP or PCP by using the MLME‑DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by invoking an MLME-SETPROTECTION.request(None) primitive. The MM-SME shall perform this process for each STA whose address was included in the MMS parameter of the MLME-ASSOCIATE.request or MLME-REASSOCIATE.request primitive that established the association.

**11.3.5.8 AP or PCP disassociation initiation procedure**

***Instruct the editor to change the d bullet in 11.3.5.8 as follows:***

* Upon receiving an MLME-DISASSOCIATE.confirm primitive, the SME shall delete any PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and temporal keys held for communication with the STA by using the MLME‑DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by invoking an MLME-SETPROTECTION.request(None) primitive. The MM-SME shall perform this process for each STA whose address was included in the MMS parameter of the MLME-ASSOCIATE.request or MLME-REASSOCIATE.request primitive that established the association.

**11.3.5.9 AP or PCP disassociation receipt procedure**

***Instruct the editor to change the c bullet in 11.3.5.9 as follows:***

* Upon receiving an MLME-DISASSOCIATE.indication primitive the SME shall delete any PTKSA, GTKSA, IGTKSA BIGTKSA, WIGTKSA and temporal keys held for communication with the STA by using the MLME‑DELETEKEYS.request primitive (see 12.6.18 (RSNA security association termination)) and by invoking an MLME-SETPROTECTION.request(None) primitive. The MM-SME shall perform this process for each STA whose address was included in the MMS parameter of the MLME-ASSOCIATE.request or MLME-REASSOCIATE.request primitive that established the association.

***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 12, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421, #2333, #2335, #2336, #2578, #2823)

**12. Security**

**12.2 Framework**

**12.2.4 RSNA establishment**

***Instruct the editor to insert a new bullet in a) after the 8) bullet and in b) after 7) bullet in 12.2.4 RSNA establishment as follows:***

An SME establishes an RSNA in one of six ways:

* If an RSNA uses authentication negotiated over IEEE Std 802.1X or FILS authentication in an infrastructure BSS, an RSNA capable STA’s SME establishes an RSNA as follows:
* …If the STAs negotiate management frame protection, the SME programs the TK and pairwise cipher suite into the MAC for protection of individually addressed robust Management frames. It also installs the IGTK and IPN for protection of group addressed robust Management frames.

8) If beacon protection is enabled, the SME programs the BIGTK and BIPN into the MAC for the protection of Beacon frames.

9) If the STAs negotiate WUR frame protection, the SME programs the WTK and WTPN into the MAC for protection of individually addressed WUR Wake-up frames, and programs the WIGTK and WIPN into the MAC for the protection of broadcast and group addressed WUR Wake-up frames. (#2337, #2579)

…

* If an RSNA is based on a PSK or password in an infrastructure BSS, the SME establishes an RSNA as follows:

…

* If the STAs negotiate management frame protection, the STA programs the TK and pairwise cipher suite into the MAC for protection of individually addressed robust Management frames. It also installs the IGTK and IPN for protection of group addressed robust Management frames.

7) If beacon protection is enabled, the SME programs the BIGTK and BIPN into the MAC for the protection of Beacon frames.

8) If the STAs negotiate WUR frame protection, the SME programs the WTK and WTPN into the MAC for protection of individually addressed WUR Wake-up frames, and programs the WIGTK and WIPN into the MAC for the protection of broadcast and group addressed WUR Wake-up frames. (#2337, #2579)

**12.5.4 Broadcast/multicast integrity protocol (BIP)**

**12.5.4.1 BIP overview**

***Instruct the editor to change 12.5.4.1 BIP overview, as follows:***

…

BIP uses the IGTK or BIGTK to compute the MMPDU MIC, uses the WTK to compute the MIC for protecting individually addressed WUR Wake-up frames, and uses the WITGK to compute the MIC for protecting broadcast or group addressed WUR Wake-up frames. The authenticator shall distribute one new IGTK and IGTK PN (IPN) whenever it distributes a new GTK. The IGTK is identified by the MAC address of the transmitting STA plus an IGTK identifier that is encoded in the MME Key ID field. If beacon protection is enabled, the authenticator may distribute one new BIGTK and BIPN when it distributes a new GTK. The BIGTK is identified by the MAC address of the transmitting STA plus a BIGTK identifier that is encoded in the MME Key ID field. If WUR frame protection is negotiated, the authenticator may distribute one new WIGTK and WIPN when it distributes a new GTK. The WIGTK is identified by the MAC address of the transmitting STA plus the WIGTK identifier.

**12.6 RSNA security association management**

**12.6.1 Security associations**

**12.6.1.1 Security association definitions**

**12.6.1.1.1 General**

***Instruct the editor to add new bullet at the end of bullet list of the second paragraph in 12.6.1.1.1 General, as follows:***

— WIGTKSA: A result of a successful group key handshake, successful 4-way handshake, successful FT 4-way handshake, the Reassociation Response frame of the fast BSS transition protocol, or successful FILS authentication.

**12.6.1.1.6 PTKSA**

***Instruct the editor to change 12.6.1.1.6 PTKSA, as follows:***

…

The PTKSA consists of the following:

* PTK, where the PTK includes the WTK when WUR frame protection is negotiated
* Pairwise cipher suite selector, and when WUR frame protection is negotiated, the cipher suite selector 00-0F-AC:6 (BIP-CMAC-128) for individually addressed WUR Wake-up frames

***Instruct the editor to add a new subclause at the end of 12.6.1.1, as follows:***

**12.6.1.1.12 WIGTKSA**

When WUR frame protection is enabled, a WUR non-AP STA’s SME creates a WIGTKSA when it receives a WIGTK in a valid message 3 of the 4-way handshake or FT 4-way handshake, the Reassociation Response frame of the fast BSS transition protocol with a status code indicating success, a valid message 1 of the group key handshake, or the (Re)Association Response frame of FILS authentication with a status code indicating success. The Authenticator’s SME creates a WIGTKSA when it establishes or changes the WIGTK with all WUR non-AP STAs to which it has a valid PTKSA. A WIGTKSA has the same lifetime as the BSS, unless superseded.

A WIGTKSA consists of the following:

— Direction vector (whether the WIGTK is used for transmit or receive)

— The cipher suite selector 00-0F-AC:6 (BIP-CMAC-128) for broadcast and group addressed WUR Wake-up frames

— Key ID

— WIGTK

— Authenticator MAC address

**12.6.1.3 Security association life cycle**

**12.6.1.3.2 Security association in an ESS**

***Instruct the editor to change the last paragraph in 12.6.1.3.2 Security association in an ESS, as follows:***

The MLME-DELETEKEYS.request primitive deletes the temporal key(s) established for the security association so that they cannot be used to protect subsequent IEEE 802.11 traffic. An SME uses this primitive when it deletes a PTKSA, GTKSA, IGTKSA, BIGTKSA, or WIGTKSA.

**12.6.2 RSNA selection**

***Instruct the editor to change the first paragraph in 12.6.2 RSNA selection, as follows:***

A STA prepared to establish RSNAs shall advertise its capabilities by including the RSNE in Beacon, Information Response, and Probe Response frames and may also include the RSNE in DMG Beacon and Announce frames. The included RSNE shall specify all of the authentication and cipher suites enabled by the STA’s policy. A STA shall not advertise any authentication or cipher suite that is not enabled. If WUR frame protection is enabled, a WUR AP shall advertise such capability by setting to 1 the WUR Frame Protection Support field in the <Extended Capabilities element or RSN Extension element> in its Beacon and Probe Response frames. (#2350, #2576)

**12.6.3 RSNA policy selection in an infrastructure BSS**

***Instruct the editor to change the first paragraph in 12.6.3 RSNA policy selection in an infrastructure BSS, as follows:***

RSNA policy selection in an infrastructure BSS utilizes the normal IEEE 802.11 association procedure. RSNA policy selection is performed by the associating STA. The STA does this by including an RSNE, and if WUR frame protection is enabled, an <Extended Capabilities element or RSN Extension element> with the WUR Frame Protection Support field set to 1 in its (Re)Association Requests. (#2350, #2576)

**12.6.14 RSNA key management in an infrastructure BSS**

***Instruct the editor to change 12.6.14 RSNA key management in an infrastructure BSS, as follows:***

…

A second key exchange, the group key handshake, is also defined. It distributes a subsequent GTK, and if WUR frame protection is negotiated, a WIGTK. The AP’s Authenticator can use the group key handshake to update the GTK, and if WUR frame protection is negotiated, the WIGTK at the Supplicant. The group key handshake uses the EAPOL-Key frames for this exchange. When it completes, the Supplicant can use the MLME-SETKEYS.request primitive to configure the GTK, and if WUR frame protection is negotiated, the WIGTK into the IEEE 802.11 MAC.

**12.6.18 RSNA security association termination**

***Instruct the editor to change the second paragraph in 12.6.18 RSNA security association termination as follows:***

…

it deletes some security associations. In the case of an ESS, the non-AP STA’s SME shall delete the PTKSA, GTKSA, IGTKSA, BIGTKSA, WIGTKSA and any TPKSA, and the AP’s SME shall delete the PTKSA. In the case of an IBSS, the SME shall delete the PTKSA and the receive GTKSA and IGTKSA.

**12.6.21 RSNA rekeying**

***Instruct the editor to insert new paragraphs to the end of 12.6.21 RSNA rekeying, as follows:***

When a WIGTKSA is deleted, an originating STA may create a new WIGTKSA by using 4-way handshake or group key handshake.

**12.7 Keys and key distribution**

**12.7.1 Key hierarchy**

**12.7.1.1 General**

***Instruct the editor to change the first paragraph in 12.7.1.1 General as follows:***

RSNA defines the following key hierarchies:

* Pairwise key hierarchy, to protect individually addressed traffic, where the PTK includes a WTK if WUR frame protection is negotiated, to provide integrity protection for individually addressed WUR Wake-up frames, and excludes the WTK otherwise
* GTK, a hierarchy consisting of a single key to protect group addressed traffic

NOTE—Pairwise key support with enhanced data cryptographic encapsulation mechanisms allows a receiving STA to detect MAC address spoofing and data forgery. The RSNA architecture binds the transmit and receive addresses to the pairwise key. If an attacker creates an MPDU with the spoofed TA, then the decapsulation procedure at the receiver generates an error. GTKs do not have this property.

* Integrity GTK (IGTK), a hierarchy consisting of a single key to provide integrity protection for group addressed robust Management frames
* BIGTK, a hierarchy consisting of a single key to provide for integrity protection for Beacon frames

e) WIGTK, a hierarchy consisting of a single key to provide integrity protection for broadcast and group addressed WUR Wake-up frames

* Pairwise key hierarchy

***Instruct the editor to change the second paragraph in 12.7.13 Pairwise key hierarchy, as follows:***(#2824)

Except when preauthentication or FILS authentication(11ai) is used, the pairwise key hierarchy utilizes PRF-384, PRF-512, or PRF-704 to derive session-specific keys from a PMK, as depicted in Figure 12-30 (Pairwise key hierarchy). When using AKM suite selector 00-0F-AC:12, the length of the PMK, PMK\_bits, shall be 384 bits. With all other AKM suite selectors, the length of the PMK, PMK\_bits, shall be 256 bits. The pairwise key hierarchy takes a PMK and generates a PTK. The PTK is partitioned into KCK, KEK, and a temporal key, which is used by the MAC to protect individually addressed communication between the Authenticator’s and Supplicant’s respective STAs, and if WUR frame protection is negotiated, a WTK, which is used by the MAC of the WUR AP to protect and by the MAC of the WUR non-AP STA to validate individually addressed WUR Wake-up frames. PTKs are used between a single Supplicant and a single Authenticator.

…

* The PTK shall be derived from the PMK by

PTK = PRF-Length(PMK, “Pairwise key expansion”, Min(AA,SPA) || Max(AA,SPA) || Min(ANonce,SNonce) || Max(ANonce,SNonce))

where Length = KCK\_bits + KEK\_bits + TK\_bits + WTK\_bits, if WUR frame protection is negotiated; otherwise, Length = KCK\_bits + KEK\_bits + TK\_bits. The values of KCK\_bits and KEK\_bits are AKM suite dependent and are listed in Table 12-8 (Integrity and key-wrap algorithms). The value of TK\_bits is cipher-suite dependent and is defined in Table 12-5 (Cipher suite key lengths). The value of WTK\_bits is 128. The Min and Max operations for IEEE 802 addresses are with the address converted to a positive integer treating the first transmitted octet as the most significant octet of the integer. The nonces are encoded as specified in 9.2.2 (Conventions).

NOTE 4(11ai)—The Authenticator and Supplicant normally derive a PTK only once per association. A Supplicant or an Authenticator use the 4-way handshake or FILS authentication(11ai) to derive a new PTK. Both the Authenticator and Supplicant create a new nonce value for each 4-way handshake or FILS authentication(11ai) instance.

* The KCK shall be computed as the first KCK\_bits bits (bits 0 to KCK\_bits–1) of the PTK:

KCK= L(PTK, 0, KCK\_bits)

The KCK is used by IEEE Std 802.1X-2010 to provided data origin authenticity in the 4-way handshake and group key handshake messages.

* The KEK shall be computed as the next KEK\_bits bits of the PTK:

KEK = L(PTK, KCK\_bits, KEK\_bits)

The KEK is used by the EAPOL-Key frames to provide data confidentiality in the 4-way handshake and group key handshake messages.

* The temporal key (TK) shall be computed as the next TK\_bits bits of the PTK:

TK = L(PTK, KCK\_bits+KEK\_bits, TK\_bits)

* If WUR frame protection is negotiated, the WTK shall be computed as the next WTK\_bits bits of the PTK:

WTK = L(PTK, KCK\_bits+KEK\_bits+TK\_bits, WTK\_bits)

Otherwise, the WTK is not derived.

**12.7.1.6.5 PTK**

***Instruct the editor to change 12.7.1.6.5 PTK, as follows:*** (#2824)

…

* Length is the total number of bits to derive, i.e., number of bits of the PTK. The length is dependent on the negotiated cipher suites and AKM suites as defined by Table 12-5 (Cipher suite key lengths) in 12.7.2 (EAPOL-Key frames) and Table 12-8 (Integrity and key-wrap algorithms) in 12.7.3 (EAPOL-Key frame construction and processing), and whether WUR frame protection is negotiated.

Except when WUR frame protection is negotiated, each PTK has five component keys, KCK, KEK, a temporal key, KCK2, and KEK2 derived as follows:

…

The KEK2 is used to provide data confidentiality for certain fields in the FT authentication sequence, as defined in 13.8 (FT authentication sequence).

When WUR frame protection is negotiated, each PTK has six component keys, KCK, KEK, a temporal key, KCK2, KEK2, and a WTK derived as follows:

The KCK, KEK, temporal key, KCK2, and KEK2 shall be computed in the same way as when WUR frame protection is not negotiated.

The WTK shall be computed as the next WTK\_bits of the PTK:

WTK = L(PTK, KCK\_bits+KEK\_bits+TK\_bits+KCK2\_bits+KEK2\_bits, WTK\_bits)

The WTK is used to protect individually addressed WUR Wake-up frames, as defined in 30.9 (Protected WUR frames).

***Instruct the editor to add a new subclause at the end of 12.7.1 as follows:***

**12.7.1.8 Wake-up radio integrity group temporal key (WIGTK) hierarchy**

The Authenticator shall select the WIGTK as a random value each time it is generated.

The Authenticator may update the WIGTK for any reason, including:

a) The disassociation or deauthentication of a WUR non-AP STA.

b) An event within the SME that triggers a group key handshake.

The WIGTK is configured via the MLME-SETKEYS.request primitive; see 6.3.19 (SetKeys). The WIGTK configuration is described in the EAPOL-Key state machines; see 12.7.9 (RSNA Supplicant key management state machine) and 12.7.10 (RSNA Authenticator key management state machine).

The WIPN is used to provide replay protection.

**12.7.2 EAPOL-Key frames**

***Instruct the editor to change Table 12-7—KDE in 12.7.2 as follows:*** (#2555)

|  |  |  |
| --- | --- | --- |
| * KDE | | |
| OUI | Data type | Meaning |
| 00-0F-AC | 0 | Reserved |
| ... ... | ... ... | ... ... |
| 00-0F-AC | 14 | BIGTK KDE |
| 00-0F-AC | 15 | WIGTK KDE(#2555) |
| 00-0F-AC | 16–255 | Reserved |
| Other OUI or CID | Any | Vendor specific |

***Instruct the editor to add a paragraph and figure at the end of 12.7.2*** ***EAPOL-Key frames as follows:*** (#2555)

The format of the WIGTK KDE is shown in Figure 12-xxx (WIGTK KDE format).

The WIPN corresponds to the WIPN value that was used for computing the MIC in the last protected broadcast or group addressed WUR Wake-up frame and it is used by the receiver as the initial value for the BIP replay counter for the WIGTK.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Key ID | WIPN | WIGTK |
| Octets: | 2 | 6 | (Length – 12) |

**Figure 12-xxx WIGTK KDE format**(#2555)

**12.7.4 EAPOL-Key frame notation**

***Instruct the editor to change 12.7.4 EAPOL-Key frame notation as follows:***

…

BIGTK[Q] is the BIGTK, with key identifier field set to Q

BIPN is the current BIGTK replay counter value provided by the BIGTK KDE

WIGTK[R] is the WIGTK, with key identifier field set to R

WIPN is the current WIGTK replay counter value provided by the WIGTK KDE

**12.7.6 4-way handshake**

**12.7.6.1 General**

***Instruct the editor to change the second paragraph in 12.7.6.1 General as follows:***

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

**12.7.6.4 4-way handshake message 3**

***Instruct the editor to change 12.7.6.4 - 4-way handshake message 3 as follows:***

...

* For PTK generation for the current operating band, the AP’s Beacon/Probe Response frame’s RSNE for the current operating band, and, optionally, a second RSNE that is the Authenticator’s pairwise cipher suite assignment for the current operating band, and, if a group cipher has been negotiated, the GTK and the GTK’s key identifier (see 12.7.2 (EAPOL-K) for the current operating band, and if management frame protection is negotiated, the IGTK KDE, and if beacon protection is enabled, the BIGTK KDE, and if WUR frame protection is negotiated, the WIGTK KDE, and when this message 3 is part of a fast BSS transition initial mobility domain association or an association started through the FT protocol, the PMKR1Name calculated according to the procedures of 12.7.1.6.4 (PMK-R1). in the PMKID field of the RSNE and the FTE with the same contents as in the (Re)Association Response frame, the MDE with the same contents as in the (Re)Association Response frame, the reassociation deadline timeout set to the minimum of dot11FTReassociationDeadline and the key lifetime in the TIE[ReassociationDeadline], and the PTK lifetime in the TIE[KeyLifetime]; or

**12.7.7 Group key handshake**

**12.7.7.1 General**

***Instruct the editor to change 12.7.7.1 General as follows:***

The Authenticator uses the Group key handshake to send a new GTK and, if management frame protection is negotiated, a new IGTK, and if beacon protection is enabled, a new BIGTK, and if WUR frame protection is negotiated, a new WIGTK to the Supplicant.

The Authenticator may initiate the exchange when a Supplicant is disassociated or deauthenticated.

Message 1: Authenticator Supplicant:

EAPOL-Key(1,1,1,0,G,0,Key RSC,0, MIC, {GTK[N], IGTK[M], BIGTK[Q], WIGTK(R)})

Message 2: Supplicant Authenticator: EAPOL-Key(1,1,0,0,G,0,0,0,MIC,{})

Here, the following assumptions apply:

— Key RSC denotes the last TSC or PN sent using the GTK.

— GTK[N] denotes the GTK with its key identifier as defined in 12.7.2 (EAPOL-Key frames) using the KEK defined in 12.7.1.3 (Pairwise key hierarchy) and associated IV.

— IGTK[M], when present, denotes the IGTK with its key identifier as defined in 12.7.2 (EAPOL-Key frames) using the KEK defined in 12.7.1.3 (Pairwise key hierarchy) and associated IV.

— BIGTK[Q], when present, denotes the BIGTK with its key identifier as defined in 12.7.2 (EAPOL-Key frames) using the KEK defined in 12.7.1.3 (Pairwise key hierarchy) and associated IV.

— WIGTK[R], when present, denotes the WIGTK with its key identifier as defined in 12.7.2 (EAPOL-Key frames) using the KEK defined in 12.7.1.3 (Pairwise key hierarchy) and associated IV.

**12.7.7.2 Group key handshake message 1**

***Instruct the editor to change 12.7.7.2 Group key handshake message 1 as follows:***

…

— When present, BIGTK, BIGTK’s key identifier, and BIPN (see 12.7.2 (EAPOL-Key frames))

— When present, WIGTK, WIGTK’s key identifier, and WIPN (see 12.7.2 (EAPOL-Key frames))

— OCI KDE when dot11RSNAOperatingChannelValidationActivated on the Authenticator

The Authenticator sends message 1 to the Supplicant.

On reception of message 1, the Supplicant:

…

d) Uses the MLME-SETKEYS.request primitive to configure the temporal GTK, the IGTK when present, the BIGTK if beacon protection is enabled, and the WIGTK if WUR frame protection is negotiated, into its IEEE 802.11 MAC.

**12.7.7.4 Group key handshake implementation considerations**

***Instruct the editor to change the second paragraph in 12.7.7.4 as follows:***

To prevent key reinstallation attacks, the Supplicant shall maintain a copy of the most recent GTK key, most recent IGTK key, most recent BIGTK, and most recent WIGTK, installed as a result of receipt of EAPOL-Key frames. The Supplicant shall not install a GTK, an IGTK, a BIGTK or a WIGTK when the key to be set matches either of these two keys (see 6.3.19 (SetKeys)).

**12.7.9.4 Supplicant state machine procedures**

***Instruct the editor to change 12.7.9.4 Supplicant state machine procedures as follows:***

…

When processing 4-way handshake message 3, the GTK, IGTK, BIGTK if present, and WIGTK if present are decrypted from the EAPOL-Key frame and installed. The PTK shall be installed before the GTK and IGTK.

**12.8 Mapping EAPOL keys to IEEE 802.11 keys**

***Instruct the editor to add two new subclauses at the end of 12.8 as follows:***

**12.8.11 Mapping WTK to BIP key**

See 12.7.1.3 (Pairwise key hierarchy) for the definition of the WTK. A STA shall use bits 0-127 of the WTK as the AES-128-CMAC key for protecting individually addressed WUR Wake-up frames.

**12.8.12 Mapping WIGTK to BIP key**

See 12.7.1.8 (Wake-up radio integrity group temporal key hierarchy) for the definition of the WIGTK. A STA shall use bits 0-127 of the WIGTK as the AES-128-CMAC key for protecting broadcast and group addressed WUR Wake-up frames.

**12.12 Authentication for FILS**

**12.12.2 FILS authentication protocol**

**12.12.2.1 General**

***Instruct the editor to change the last paragraph in 12.12.2.1 as follows:***

To prevent key reinstallation attacks, the non-AP STA shall maintain a copy of the most recent GTK, most recent IGTK, most recent BIGTK, and most recent WIGTK installed as part of the FILS authentication protocol as if they were installed as a result of receipt of EAPOL-Key frames (see 12.7.7.4 (Group key handshake implementation considerations)) and shall refuse to update a GTK, IGTK, BIGTK or WIGTK when the key to be set matches either one of these two keys (see 6.3.19 (SetKeys)).

* PTKSA Key derivation with FILS authentication

***Instruct the editor to change 12.12.2.5.3 12.12.2.5.3 PTKSA Key derivation with FILS authentication as follows:*** (#2824)

For PTKSA key generation, the inputs to the PRF are the PMK of the PMKSA, a constant label, and a concatenation of the STA’s MAC address, the AP’s BSSID, the STA’s nonce, and the AP’s nonce. When the negotiated AKM is 00-0F-AC:14 or 00-0F-AC:16, the length of KEK shall be 256 bits, and the length of the ICK shall be 256 bits. When the negotiated AKM is 00-0F-AC:15 or 00-0F-AC:17, the length of the KEK shall be 512 bits, and the length of ICK shall be 384 bits. When the negotiated AKM is 00-0F-AC:16, FILS-FT is 256 bits; when the negotiated AKM is 00-0F-AC:17, FILS-FT is 384 bits; otherwise, FILS-FT is not derived; when WUR frame protection is negotiated, the WTK is 128 bits; otherwise, the WTK is not derived. The total amount of bits extracted from the KDF shall therefore be 640+TK bits, 1124+TK bits, or 1408+TK bits depending on the negotiated AKM when WUR frame protected is negotiated, otherwise, shall be 512+TK bits, 896+TK bits, or 1280+TK bits depending on the negotiated AKM, where TK\_bits are determined from Table 12-4:

(#114)FILS-Key-Data = PRF-X(PMK, “FILS PTK Derivation”, SPA || AA || SNonce || ANonce [ || DHss ])

ICK = L(FILS-Key-Data, 0, ICK\_bits)

KEK = L(FILS-Key-Data, ICK\_bits, KEK\_bits)

TK = L(FILS-Key-Data, ICK\_bits + KEK\_bits, TK\_bits)

When doing FT initial mobility domain association using FILS authentication,

FILS-FT = L(FILS-Key-Data, ICK\_bits + KEK\_bits + TK\_bits, FILS-FT\_bits)

When WUR frame protection is negotiated while doing FT initial mobility domain association using FILS authentication,

WTK = L(FILS-Key-Data, ICK\_bits + KEK\_bits + TK\_bits + FILS-FT\_bits, WTK\_bits)

When WUR frame protection is negotiated while not doing FT initial mobility domain association using FILS authentication,

WTK = L(FILS-Key-Data, ICK\_bits + KEK\_bits + TK\_bits, WTK\_bits)

where

ICK\_bits is the length of ICK in bits

KEK\_bits is the length of KEK in bits

FILS-FT\_bits is the length of FILS-FT in bits when doing FT initial mobility domain association using FILS authentication

WTK\_bits is the length of WTK in bits and is equal to 128

X is 640+TK bits, 1124+TK bits, or 1408+TK bits from Table 12-5 (Cipher suite key lengths) depending on the negotiated AKM when WUR frame protection is negotiated; otherwise, is 512+TK\_bits, 768+TK bits, 896+TK bits, or 1280+TK bits from Table 12-5 (Cipher suite key lengths) depending on the negotiated AKM

**12.12.2.6 Key confirmation with FILS authentication**

**12.12.2.6.3 (Re)Association Response for FILS key confirmation**

***Instruct the editor to change the second paragraph and the last paragraph in 12.12.2.6.3 (Re)Association Response for FILS confirmation, as follows:***

…

The AP constructs a Key Delivery element indicating the current GTK and Key RSC, the current IGTK and IPN if management frame protection is enabled, the current BIGTK and BIPN if beacon protection is enabled, and the current WIGTK and WIPN if WUR frame protection is enabled. The GTK is carried in a GTK KDE with Tx subfield equal to 0. The IGTK and IPN are carried in an IGTK KDE. The BIGTK and BIPN are carried in a BIGTK KDE. The WIGTK and WIPN are carried in a WIGTK KDE. The AP puts this element into the (Re)Association Response frame.

…

Upon successful completion of the FILS authentication procedure, the STA shall process the Key Delivery element in the (Re)Association Response frame. The STA installs the GTK and key RSC, and IGTK and IPN if management frame protection is enabled, and BIGTK and BIPN if present in the key delivery element and dot11BeaconProtectionEnabled is true, and WIGTK and WIPN if present in the key delivery element and dot11RSNAWURFrameProtectionActivated is true.

***TGba Editor: in P802.11ba draft, add the following instruction and Figure 12-30, after inserting, under the PTK block and on the right of the TK block, a new block containing the following text, center-aligned:***

**“WUR Temporal Key**

**L(PTK, KCK\_bits+KEK\_bits+TK\_bits, WTK\_bits)**

**(WTK)”** (#2318, #2334, #2421)

**12.7.1.3 Pairwise key hierarchy**

***Instruct the editor to replace Figure 12-30 Pairwise key hierarchy with the figure as follows***



***TGba editor: in P802.11ba draft, add the following changes to various subclauses of Clause 13, including the respective instructions. The baseline text used is REVmd D2.2.*** (#2318, #2334, #2421, #2333, #2335, #2336, #2578, #2823)

**13. Fast BSS transition**

**13.2 Key holders**

**13.2.2 Authenticator key holders**

***Instruct the editor to change 13.2.2 Authenticator key holders as follows:***

…

The R1KH shall meet the following requirements:

— The R1KH-ID shall be set to a MAC address of the physical entity that stores the PMK-R1 and uses it to generate the PTK. That same MAC address shall be used to advertise the PMK-R1 identity to the STA and the R0KH.

— The R1KH shall derive and distribute the GTK and IGTK to all connected STAs.

— If beacon protection is enabled, the R1KH shall derive and distribute the BIGTK and BIPN to all connected STAs.

— If WUR frame protection is enabled, the R1KH shall derive and distribute the WIGTK and WIPN to all WUR non-AP STAs, with which the R1KH has negotiated WUR frame protection.

**13.4 FT initial mobility domain association**

**13.4.2 FT initial mobility domain association in an RSN**

***Instruct the editor to change 13.4.2 FT initial mobility domain association in an RSN as follows:***

…

The R1KH and S1KH then perform an FT 4-way handshake. The EAPOL-Key frame notation is defined in 12.7.4 (EAPOL-Key frame notation).

R1KHS1KH: EAPOL-Key(0, 0, 1, 0, P, 0, 0, ANonce, 0, {})

S1KHR1KH: EAPOL-Key(0, 1, 0, 0, P, 0, 0, SNonce, MIC, {RSNE[PMKR1Name], MDE, GTK[N], IGTK[M], FTE, TIE[ReassociationDeadline], TIE[KeyLifetime], RSNE[PMKR1Name], MDE, FTE})

R1KHS1KH: EAPOL-Key(1, 1, 1, 1, P, 0, 0, ANonce, MIC, {RSNE[PMKR1Name], MDE, GTK[N], IGTK[M], BIGTK[Q], WIGTK[R], FTE, TIE[ReassociationDeadline], TIE[KeyLifetime]})

**13.5 FT protocol**

**13.5.1 Overview**

***Instruct the editor to change the last paragraph in 13.5.1 as follows:***

To prevent key reinstallation attacks, the non-AP STA shall maintain a copy of the most recent GTK, IGTK, BIGTK when present, and WIGTK when present, installed as part of the FT protocol as if they were installed as a result of receipt of EAPOL-Key frames (see 12.7.7.4 (Group key handshake implementation considerations)) and shall refuse to update a GTK, an IGTK, a BIGTK or a WIGTK when the key to be set matches either one of these two keys (see 6.3.19 (SetKeys)).

**13.6 FT resource request protocol**

**13.6.2 Over-the-air fast BSS transition with resource request**

***Instruct the editor to change 13.6.2 Over-the-air fast BSS transition with resource request - as follows:***

…

In an RSN, on successful completion of the FT authentication exchange of the FT resource request protocol, the PTKSA has been established and proven live. The key replay counter shall be initialized to 0, and the subsequent EAPOL-Key frames (e.g., GTK, IGTK, BIGTK and WIGTK updates) shall use the key replay counter to detect and discard replays. The PTKSA shall be deleted by the target AP if it does not receive a Reassociation Request frame from the FTO within the reassociation deadline timeout value.

**13.6.3 Over-the-DS fast BSS transition with resource request**

***Instruct the editor to change 13.6.3 Over-the-DS fast BSS transition with resource request as follows:***

…

In an RSN, on successful completion of the FT Confirm/Acknowledgment frame exchange, the PTKSA has been established and proven live. The key replay counter shall be initialized to 0, and the subsequent EAPOL-Key frames (e.g., GTK, IGTK, BIGTK and WIGTK updates) shall use the key replay counter to detect and discard replays.

**13.7 FT reassociation**

**13.7.1 FT reassociation in an RSN**

***Instruct the editor to change the second paragraph in 13.7.1 FT reassociation in an RSN as follows:***

The FTO shall perform a reassociation directly with the target AP via the following exchange:

FTOTarget AP: Reassociation Request(RSNE[PMKR1Name], MDE, FTE[MIC, ANonce, SNonce, R1KH-ID, R0KH-ID], RIC-Request)

Target APFTO: Reassociation Response(RSNE[PMKR1Name], MDE, FTE[MIC, ANonce, SNonce, R1KH-ID, R0KH-ID, GTK[N], IGTK[M], BIGTK[Q], WIGTK[R]], RICResponse)

**13.8 FT authentication sequence**

**13.8.5 FT authentication sequence: contents of fourth message**

***Instruct the editor to change 13.8.5 FT authentication sequence: contents of fourth message as follows:***

…

* When this message of the authentication sequence appears in a Reassociation Response frame, the Optional Parameter(s) field in the FTE may include the GTK, IGTK, BIGTK and WIGTK subelements. If a GTK, an IGTK, a BIGTK or a WIGTK are included, the Key field of the subelement shall be encrypted using KEK (when the negotiated AKM is 00-0F-AC:3, 00-0F-AC:4, 00-0F-AC:9, or 00-0F-AC:13) or KEK2 (when the negotiated AKM is 00-0F-AC:16 or 00-0F-AC:17) and the NIST AES key wrap algorithm. The Key field shall be padded before encrypting if the key length is less than 16 octets or if it is not a multiple of 8. The padding consists of appending a single octet 0xdd followed by zero or more 0x00 octets. When processing a received message, the receiver shall ignore this trailing padding. Addition of padding does not change the value of the Key Length field. Note that the length of the encrypted Key field can be determined from the length of the GTK, IGTK, BIGTK or WIGTK subelement.

***TGba Editor: Change Subclause 30.9 in TGba Draft 2.1 as follows:*** (#2418, #2419)

* Protected WUR frames

WUR frame protection cannot be applied until the PTKSA (see 12.6.1.1.6 PTKSA) and WIGTKSA (see 12.6.1.1.12 WIGTKSA) have been established. (#2419)

WUR frame protection is enabled when dot11RSNAWURFrameProtectionActivated is true, and is disabled otherwise. When WUR frame protection is enabled at a WUR AP, the WUR AP shall advertise such capability by setting to 1 the WUR Frame Protection field of the <Extended Capabilities element or RSN Extension element> in its Beacon and Probe Response frames. When WUR frame protection is enabled at a WUR non-AP STA, the WUR non-AP STA shall indicate such capability by setting to 1 the WUR Frame Protection field of the <Extended Capabilities element or RSN Extension element> in its (Re)Association Request frames. (#2418, #2350, #2576)

WUR frame protection is negotiated between the WUR AP and the WUR non-AP STA when management frame protection is negotiated, both parties set the WUR Frame Protection Support field to 1 in their respective <Extended Capabilties elements or RSN Extension elements> in the (re)association procedure, and both parties confirm the WUR Frame Protection Support bit set to 1 in the 4-way handshake, FT 4-way handshake, FT fast BSS transition protocol, or the (Re)Association Request and (Re)Association Response frames of FILS authentication. Otherwise, WUR frame protection is not negotiated between the WUR AP and the WUR non-AP STA. (#2418, #2350, #2576)

If WUR frame protection is negotiated between a WUR AP and a WUR non-AP STA, the WUR AP may transmit a protected individually addressed WUR Wake-up frame or a protected broadcast or group addressed WUR Wake-up frames to the WUR non-AP STA; otherwise the AP shall not transmit a protected WUR Wake-up frame to the WUR non-AP STA. (#2418)

The WUR AP may transmit a protected WUR Wake-up frame addressed to more than one WUR non-AP STAs if WUR frame protection is negotiated with and the WIGTK has been transported to all the WUR non-AP STAs being addressed to. (#2418)

The WUR AP shall not transmit a protected individually addressed WUR Wake-up frame to the WUR non-AP STA until the WTK is installed at the WUR non-AP STA, and should not transmit a protected broadcast or group addressed WUR Wake-up frame until the WIGTK is installed at all WUR non-AP STAs, with which the WUR AP has negotiated WUR frame protection. The WUR non-AP STA shall discard any protected WUR frames received before the WTK and WIGTK are installed. (#2418)

***TGba Editor: Change C.3 in TGba Draft 2.1 as follows:*** (#2418)

**C.3 MIB detail**

***Change dot11SattionConfigEntry as follows:***

Dot11StationConfigEntry::= SEQUENCE

{

…,

dot11LocallyAdministeredMACConfig Unsigned32,

dot11WUROptionImplemented TruthValue,

dot11WURBeaconPeriod Unsigned32,

dot11WURFDMAChannelSwitchImplemented TruthValue,

dot11WURDiscoveryImplemented TruthValue,

dot11WURNeighborDiscoveryImplemented TruthValue,

dot11WURDiscoveryPeriod Unsigned32,

dot11RSNAWURFrameProtectionActivated Truth Value(#2418) }

***Insert the following after the dot11LocallyAdministeredMACConfig OBJECT-TYPE in the Dot11StationConfig TABLE and maintain the order based on the assigned value:***

…

dot11WURDiscoveryPeriod OBJECT-TYPE

SYNTAX Unsigned32(1..65535)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable. It is written by an external management entity. Changes take effect for the next MLME-START.request primitive. For WUR STAs, this attribute specifies the number of TUs that a station uses for scheduling WUR Discovery frame transmissions. This value is transmitted in Beacon or Probe Response frames."

::= { dot11StationConfigEntry 193}

dot11RSNAWURFrameProtectionActivated OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable. It is written by an external management entity. Changes take effect for the next MLME-START.request primitive or MLME-JOIN.request primitive. This attribute, when true, indicates that the station, if a WUR AP, is capable of transmitting protected WUR frames, or if a WUR non-AP STA, is capable of receiving protected WUR frames. Otherwise, the capability is disabled."

DEFVAL { false }

::= { dot11StationConfigEntry <ANA>}(#2418)