IEEE P802.11
Wireless LANs

|  |
| --- |
| EAPoL-Key Notation  |
| Date: 2022-10-10 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Michael Montemurro | Huawei |  |  | montemurro.michael@gmail.com  |

Background

This contribution deals with updating EAPoL-Key notation.

### Discussion:

The updated EAPoL-key notation, particularly for the Key Data field requires every possible Key Data combination of elements to be shown. Following this update, the notation needs to document every combination of possible values for Key Data rather than to just show which KDEs are mandatory/optional.

This document was posted to address previous comments in the CC and WG LB rounds.

Notation:

* The Key Data field value is denoted by {…}
* Optional key data elements are denoted by […, ]

Other issues:

* There are two PMKID entries in the list of key data KDEs/elements
	+ From Jouni: That duplicated PMKID line (and also the entry following it) seems to be editing issue in REVmd/D3.0. CID 2541 was resolved with editing instructions asking the two entries to be added to the list at REVmd/D2.0 P2626 L35 which is the "Here, the following assumptions apply" list in 12.7.6.1. These two lines were added incorrectly into 12.7.4 (REVmd/D2.0 P2625 L32). Those two entries should be moved to the correct list to fix this.

### Proposed Resolution:

Revised. Update the EAPoL-key notation for Key Data to make it less cumbersome and more extensible in <this>

***Update the following text in this clause as follows:***

* EAPOL-Key PDU notation(#1836)

The following notation is used throughout the remainder of 12.7 (Keys and key distribution) and 13.4 (FT initial mobility domain association) to represent (#1836)EAPOL-Key PDUs:

EAPOL-Key(S, M, A, I, K, Reserved, (#1406)RSC, ANonce/SNonce, MIC, {Key Data})

where

S (#1082)is the Secure bit of the Key Information field

M means the MIC is available in message. (#216)This should be set in all messages except message 1 of a 4-way handshake. This is the (#1829)Key MIC Present bit of the Key Information field. (#1831)When using an AEAD cipher, this (#1829)Key MIC Present bit is set to 0 regardless of the M parameter value.

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

I is the Install bit: indicates whether to install (1) or not install (0) for the pairwise key. This is the Install bit of the Key Information field(#216).

K is the key type: P (Pairwise), G (Group); this is the Key Type bit of the Key Information field

Reserved is reserved

(#1406)RSC is the RSC; this is the (#1406)RSC field

ANonce/SNonce is the Authenticator or Supplicant nonce, respectively. (#216)This is the Key Nonce field.

MIC is the integrity check, which is generated using the (#3744)PTK-KCK. This is the Key MIC field. (#1831)When using an AEAD cipher, (#1825)this parameter is ignored, and no Key MIC field is included in the EAPOL-Key PDU(#216).

{Key Data} is a sequence of zero or more elements and KDEs, concatenated and contained in the Key Data field, where

 RSNE is (#3493)the RSNE, described in 9.4.2.23 (RSNE)

 RSNE(KeyName) is the RSNE, with the PMKID List field set to KeyName

 GTK(N) is the GTK KDE, with the (#3493)Key ID field set to N (The key ID specifies which index is used for this GTK. (#3056)Indices 0 and 3 shall not be used for GTKs)

 FTE is the (#1776)FTE, described in 9.4.2.46 (FTE(#1776))

 MDE is the (#1776)MDE, described in 9.4.2.45 (MDE(#1776))

 TIE(IntervalType) is a (#1776)TIE of type IntervalType, as described in 9.4.2.47 (TIE(#1776)), containing e.g., for type KeyLifetime, the lifetime of the FT key hierarchy

 IGTK(M, IPN) is the (#3493)IGTK KDE, with the Key ID field set to M

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

 BIGTK(Q, BIPN) is the (#3493)BIGTK KDE, with the Key ID field set to Q

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

 (11ba)WIGTK(R, WIPN) is the (#3493)WIGTK KDE, with the Key ID field set to R

 (11ba)WIPN is the current WIGTK replay counter value provided by the WIGTK KDE

 PMKID is (#3493)the PMKID KDE and is the PMK identifier used during the 4-way handshake for PMK identification

 OCI KDE is (#3493)the OCI KDE

 RSNXE is (#3493)the RSNXE, described in 9.4.2.240 (RSNXE(#1776))

 PMKID (#3493)is the PMK identifier for the PMKSA selected by the Authenticator

***Update the following text in this clause as follows:***

* 4-way handshake
* General

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

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

Message 1: Authenticator ® Supplicant: EAPOL-Key(0,0,1,0,P,0,0,ANonce,0, {[PMKID]})

~~Message 2: Supplicant ® Authenticator: EAPOL-Key(0,1,0,0,P,0,0,SNonce,MIC,{RSNE} or {RSNE, OCI KDE} or {RSNE, RSNXE} or {RSNE, OCI KDE, RSNXE})~~

Message 2: Supplicant ® Authenticator: EAPOL-Key(0,1,0,0,P,0,0,SNonce,MIC, {RSNE, [, OCI] [, RSNXE]})

~~Message 3: Authenticator®Supplicant:
EAPOL-Key(1,1,1,1,P,0,KeyRSC,ANonce,MIC,{RSNE,GTK[N]} or
{RSNE, GTK[N], OCI KDE} or {RSNE, GTK[N], RSNXE} or
{RSNE, GTK[N], OCI KDE, RSNXE})~~

Message 3: Authenticator®Supplicant:
EAPOL-Key(1,1,1,1,P,0,KeyRSC,ANonce,MIC,{ RSNE, GTK(N)[, IGTK(M, IPN)] [, BIGTK(Q, BIPN)] [, OCI] [, RSNXE][, WIGTK(R, WIPN)]})

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

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

The following apply:

* EAPOL-Key(´) denotes an EAPOL-Key frame conveying the specified argument list, using the notation introduced in 12.7.4 (EAPOL-Key frame notation).
* ANonce is a nonce that the Authenticator contributes for PTK generation. ANonce has the same value in message 1 and message 3.
* SNonce is a nonce from the Supplicant for PTK generation.
* P means the pairwise bit is set.
* The MIC is computed over the body of the EAPOL-Key frame (with the Key MIC field first zeroed before the computation) using the KCK defined in 12.7.1.3 (Pairwise key hierarchy) for PTK generation.
* RSNE represents the appropriate RSNEs.
* GTK[N] represents the GTK with its key identifier.
* OCI KDE contains the current operating channel information for the operating channel in which the EAPOL-Key frame is sent. OCI KDE is present when dot11RSNAOperatingChannelValidationActivated is true on the Supplicant in Message 2 and Authenticator in Message 3. Otherwise it is absent.
* RSNXE, when included in message 2, contains the RSNXE that the Supplicant sent in its (Re)Association Request frame, and when included in message 3, contains the RSNXE that the Authenticator sent in its Beacon or Probe Response frame. RSNXE is present in message 2 if this element is present in the (Re)Association Request frame that the Supplicant sent, and is present in  message 3 if this element is present in the Beacon or Probe Response frame that the Authenticator  sent.
* PMKID identifies the PMKSA selected by the Authenticator
* “[, a]” identifies that element “a” is conditionally present in {Key Data}
* Group key handshake
* General

***Update the following text in this clause 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 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]})~~

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

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

* FT initial mobility domain association
* Overview

The FT initial mobility domain association is the first (re)association in the mobility domain, where the SME of the STA enables its future use of the FT procedures.

FT initial mobility domain association is typically the first association within the ESS. In addition to Association Request and Response frames, Reassociation Request and Response frames are supported in the initial mobility domain association to enable both FT and non-FT APs to be present in a single ESS.

* FT initial mobility domain association in an RSN

***Update the following text in this clause 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, FTE, RSNXE})~~

S1KH®R1KH: EAPOL-Key(0, 1, 0, 0, P, 0, 0, SNonce, MIC, {RSNE(PMKR1Name), MDE, FTE [, RSNXE]})

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

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

S1KH®R1KH: EAPOL-Key(1, 1, 0, 0, P, 0, 0, 0, MIC, {})

The message sequence is described in 12.7.6 (4-way handshake).