IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolving Some Security CIDs | | | | |
| Date: 2017-09-08 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Dan Harkins | HPE | 3333 Scott boulevard  Santa Clara, California  United States of America | Yes I have a phone | Yes, I have email too |
|  |  |  |  |  |

Abstract

This submission addresses CIDs 92, 139, 224, 225, and 245.

CID 92

Comment: “REVmc extended P802.11ad to allow SAE and FT authentication to be used in DMG. However, those changes did not cover all places where the original 802.11ad restriction were added. This work should be completed to make the standard unambiguous on SAE, FT, and now also FILS authentication being allowed in DMG even when Open System authentication is not used there.”

Proposed change: “Replace "SAE authentication and Open System IEEE 802.11 authentication are used by non-DMG STAs in an RSN for an infrastructure BSS" with "SAE authentication and Open System IEEE 802.11 authentication are used by STAs in an RSN for an infrastructure BSS."

Discussion: Seems reasonable.

Resolution: Accept.

CID 139

Comment: “The Cipher Suite Selector is not needed in SETKEYS.request, because you always know exactly what suite you will use.”

Proposed change: “Delete the row at 348.25 and the references to Cipher Suite Selector in 6.3.19.1.4.”

Discussion: the Key ID and key length should be able to unambiguously tell the MAC what the specific cipher is.

Resolution: like to accept but “delete…the references to Cipher Suite Selector” is a bit ambiguous for the editor so let’s do Counter and instruct the editor to modify the draft to incorporate the following changes:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Valid Range** | **Description** |
| Key | Bit string | N/A | The temporal key value |
| Length | Integer | N/A | The number of bits in the Key to be used |
| Key ID | Integer | 0-3 shall be used with WEP, TKIP, CCMP, and GCMP, 4-5 with BIP, and 6-4095 are reserved | Key identifier |
| Key Type | Integer | Group, Pairwise, PeerKey, IGTK | Defines whether this key is a group key, pairwise key, PeeerKey, or Integrity Group key. |
| Address | MACAddress | Any valid individual MAC address | This parameter is valid only when the Key Type value is Parwise, when the Key Type value is Group and the STA is in IBSS, or when the Key Type value is PeerKey |
| Receive Sequence Count | 8 octets | N/A | Value to which the RSC(s) is initialized. |

**6.3.19.1.4 Effect of receipt**

Receipt of this primitive causes the MAC to apply the keys as follows, subject to the MLMESETPROTECTION. request primitive:

* The MAC uses the key information (as defined by the Key Type, Key ID, and Address) for the transmission of subsequent frames to which the key applies (as defined by the Key Type, Key ID and Address elements).
* The MAC installs the key with the associated Key ID such that received frames for that cipher, of the appropriate type, and containing the matching Key ID are processed using that key and its associated state information, subject to validation based on the Receive Sequence Count.

CID 224

Comment: “It says "The TPKSA shall be deleted by the TDLS responder STA if it does not receive a valid TPK handshake message 3 from the TDLS Initiator STA within dot11TDLSResponseTimeout." But per 12.6.1.2 "The TPKSA results from a successful completion of the TPK handshake." so at this point there is no TPKSA to delete.”

Proposed change: “Change the para to "The TDLS responder STA shall abandon the TPK handshake if it does not receive a valid TPK handshake message 3 from the TDLS Initiator STA within dot11TDLSResponseTimeout."

Discussion: The security association is not a hypothetical entity, it consists of state and memory allocated and set to certain values. If one “shall abandon”, or completely give up on, a handshake what becomes of this state and memory that was just about to become a full-fledged TPKSA if only message 3 was received in time? It has to be deleted which is what the sentence says. But it’s not quite yet a TPKSA as the comment notes. Need a way to refer to this stuff.

Resolution: Let’s call it a “nascent TPKSA” and instruct the editor to incorporate the following changes into the draft:

**12.7.9.4.4 TPK handshake message 3**

The nascent TPKSA shall be deleted by the TDLS responder STA if it does not receive a valid TPK handshake message 3 from the TDLS Initiator STA within dot11TDLSResponseTimeout.

CID 225

Comment: “It says ‘, and delete existing TPK handshake key state for this sequence’, but there is no definition of ‘handshake key state’. The deletion of state is covered by ‘shall abandon the TPK handshake identified by the <ANonce, SNonce> combination’ anyway.”

Proposed change: “Change ‘The TDLS responder STA shall discard the message, the TDLS responder STA shall abandon the TPK handshake identified by the <ANonce, SNonce> combination, and delete existing TPK handshake key state for this sequence’ to ‘The TDLS responder STA shall discard the message, and abandon the TPK handshake identified by the <ANonce, SNonce> combination’"

Discussion: Abandoning a handshake doesn’t say what to do with the state you’ve created doing all the handshaking up to the point of abandonment. This is similar to CID 224 (see aboe).

Resolution: let’s call it a “nascent TPKSA” and instruct the editor to incorporate the following changes into the draft:

**12.7.9.4.4 TPK handshake message 3**

The TDLS responder STA shall discard the message, the TDLS responder STA shall abandon the TPK handshake identified by the <ANonce, SNonce> combination, and delete the nascent TPKSA for this sequence if any of the following checks fail:

The contents of the RSNE are not the same as that sent by the TDLS responder STA in message 2

The Timeout Interval element is not the same as that sent in message 2

The BSSID from the Link Identifier element is not the same as that sent in message 2

 On successful processing of message 3, the TPK handshake is considered successful.

CID 245

Comment: “’The Authenticator shall select the IGTK as a random value each time it is generated.’ -- how is this random value arrived at?”

Proposed change: “Add wording as in 12.7.1.4”

Discussion: (this discussion assumes the comment is on 12.7.1.5) The procedure given in 12.7.1.5 is an example of one way to generate a GTK using a random GMK and a Gnonce contribution to produce the GTK. There is no reason to replicate this example in 12.7.1.5. The text says the IGTK is a random value. The comment is “how is this random value arrived at?” The answer is, like every other way a random value is arrived at, appendix J.5 gives a few nice suggestions on how to do it.

Resolution: Reject.

**References:**