IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolution of Some Security Commits from Letter Ballot 193 | | | | |
| Date: 2013-03-18 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Dan Harkins | Aruba Networks | 1322 Crossman ave, Sunnyvale, CA | +1 408 227 4500 | dharkins at arubanetworks dot com |
|  |  |  |  |  |

Abstract

1071, 1073, 1074, 1075, 1092, 1093, 1095, 1096, 1097, 1098, 1328

***Instruct the editor to modify section 11.1.5 as indicated:***

**11.1.5 RSNA PeerKey Support**

The PeerKey protocol provides mutual authentication, session identification, and data confidentiality for a station-to-station connection. A PeerKey association, composed of an SMKSA and an STKSA, shall be allowed only within the context of an existing RSNA by both peers with a common AP, or between peer Aps that implement the AP PeerKey protool. Both the initiator STA and the peer STA shall ensure that dot11RSNAActivated is true before initiating the STSL master key (SMK) Handshake and STSL transient key (STK) Handshake and establishing their respective security associations.

***Instruct the editor to modify section 11.4.2.1.2 as indicated:***

**11.4.2.1.2 TKIP cryptographic encapsulation**

1. If needed, the MSDU with MIC is fragmented into one or more MPDUs. TKIP assigns a monotonically increasing TSC value to each MPDU, taking care that all the MPDUs generated from the same MSDU have the same value of extended IV (see 11.4.2.2 (TKIP MPDU formats)).

***Insruct editor to modify section 11.4.3.4.4 as indicated:***

**11.4.3.4.4 PN and replay detection**

1. If dot11RSNAProtectedManagementFramesActivated is true, the recipient shall maintain a single replay counter for received individually addressed robust Management frames that are not affordedthe QMF service and shall use the PN from the received frame to detect replays. If dot11QMFActivated is also true, the recipient shall maintain an additional replay counter for each ACI for received individually addressed Robust Management frames that is afforded the QMF service. The QMF receiver shall use the ACI encoded in the Sequence Number field of the received frame to select the replay counter to use for the received frame, and shall use the PN from the received frame to detect replays. A replayed frame occurs when the PN from the frame is less than or equal to the current value of the management frame replay counter that corresponds to the ACI of the frame. The transmitter shall preserve the order of protected robust Management frames that are transmitted to the same DA without the QMF service. When the QMF service is used, the transmitter shall not reorder robust IQMFs within an AC when the frames are transmitted to the same RA.

***Instruct the editor to modify section 11.4.4.4 as indicated:***

**11.4.4.4 BIP replay protection**

The MME Sequence Number field represents a sequence number whose length is 6 octets.

When management frame protection is negotiated, the receiver shall maintain a 48-bit replay counter for each IGTK. The receiver shall set the receive replay counter to the value of the IPN in the IGTK key data encapsulation (KDE) (see 11.6.2) provided by the Authenticator in either the 4-Way Handshake, FT 4-Way Handshake, FT Handshake, or Group Key Handshake. The transmitter may reinitialize the sequence counter when the IGTK is refreshed. See 11.4.4.5 (BIP transmission) and 11.4.4.6 (BIP reception) for per packet BIP processing.

***Instruct the editor to modify section 11.5.1.1.11 as indicated:***

**11.5.1.1.11 SMKSA**

An SMKSA is the result of a successful SMK Handshake by the initiator STA (described in 11.6.8 (PeerKey Handshake) and 11.10 (AP PeerKey support)). It is derived from parameters provided by the STAs and AP. This security association is bidirectional between the initiator and the peer STA. In other words, both parties use the information in the security association for both sending and receiving. The SMKSA is created as a result of a successful SMK Handshake (see 11.6.8 (PeerKey Handshake) and 11.10 (AP PeerKey support)).

***Instruct the editor to modify section 11.5.1.1.12 as indicated:***

**11.5.1.1.12 STKSA**

The STKSA is a result of successful completion of the 4-Way STK Handshake. This security association is bidirectional between the initiator and the peer STAs. The STKSA is used to create session keys to protect this STSL. STKSAs are cached for the life of the SMKSA or until the STSL ends, whichever comes first. There shall be only one STKSA with the same initiator STA and peer MAC addresses at any one time. STKSA is created as a result of PeerKey Handshake (see 11.6.8 (PeerKey Handshake)) or the AP PeerKey protocol (see 11.10 (AP PeerKey support)).

***Instruct the editor to modify section 11.5.9.2 as indicated:***

**11.5.9.2 Preauthentication and RSNA key management**

Preauthentication uses the IEEE Std 802.1X protocol and state machines with EtherType 88-C7, rather than the EtherType 88-8E. Only IEEE Std 802.1X frame types EAP-Packet and EAPOL-Start are valid for preauthentication.

A Supplicant may initiate preauthentication when it has completed the 4-Way Handshake and configured the required temporal keys. To effect preauthentication, the Supplicant sends an EAPOL-Start frame with the DA being the BSSID of a targeted AP and the RA being the BSSID of the AP with which it is associated.

***Instruct the editor to modify section 11.10.2 as indicated:***

**11.10.2 AP PeerKey protocol**

Entropy of the shared secret shall then be extracted using function H to produce *keyseed* using Equation (11-3).

*keyseed* = H(<0>32, k) (11-3)

Where <0>32 indicates thirty-two (32) octets of the value zero (0). The PMK shall be derived using the key derivation function (KDF) from 11.6.1.7.2 (Key derivation function (KDF)) using Equation (11-4).

PMK = KDF-256(keyseed, ìAP Peerkey Protocolî,

0x00 || Max(LOCAL-MAC, PEER-MAC) || Min(LOCAL-MAC, PEER-MAC) ) (11-4)

where

0x00 is a single octet with a value of zero

LOCAL-MAC is the APís BSSID

PEER-MAC is the peer APís BSSID

The Min and Max operations for IEEE Std 802 addresses are with the address converted to a positive integer treating the first transmitted octet as the most significant octet of the integer. Keyseed shall be irretrievably destroyed after the PMK is generated.

***Instruct the editor to modify section 13.5.1 as indicated:***

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

**13.5.1 Overview**

The authenticated mesh peering exchange (AMPE) establishes an authenticated mesh peering between the

mesh STAs, under the assumption that mesh PMKSA has already been established before the initiation of

the protocol. An authenticated mesh peering includes a mesh peering, corresponding mesh TKSA, and the

two mesh STAs mesh GTKSAs.

The AMPE is also used to establish an authenticated peering between two Aps that support the AP Peerkey protocol (as defined in 11.10) under the assumption that a PMK has already been established before the inititiation of the AMPE exchange.

The AMPE uses Mesh Peering Management frames. Parameters are exchanged via the RSNE, the

Authenticated Mesh Peering Exchange element, and the MIC element.

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

management.

***Instruct the editor to modify section 13.5.2.1 as indicated:***

**13.5.2.1 Instance Pairwise Cipher Suite selection**

Pairwise cipher suite selectors WEP-40, WEP-104, and TKIP shall not be used as the pairwise cipher suite when dot11MeshSecurityActivated, dot11ProtectedTXOPNegotiationActivating, or dot11ProtectedQLoadReportActivated is enabled.

If the pairwise cipher suite has not been selected, STAs shall attempt to reach the agreement on the

pairwise cipher suite using the following procedure in four steps:

***Instruct the editor to modify section 13.5.7 as indicated:***

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

The MTK is used to protect communications between two peer STAs. The local STA and peer

STA derive an MTK per peering instance and may rekey the MTK using AMPE.

**References:**