IEEE P802.11
Wireless LANs

|  |
| --- |
| MAC Address Change Scrambler Reset |
| Date: 2018-12-26 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Assaf Kasher | Qualcomm |  |  | akasher@qti.qualcomm.com |
|  |  |  |  |  |

Abstract

This contribution addresses an issue with scrambler reset when the MAC address changes as proposed by TGaq and inserted into TGmd.

Discussion:

Clause 17.3.5.5 now has the following text:

If dot11MACPrivacyActivated is true, the initial state of the scrambler shall be reset when the STA’s

MAC address is changed.

There are a few issues with this text:

1. How does the PHY know when the MAC address has changed.
2. The scrambler shift register is supposed to be set a non-zero pseudo-random value. What does resetting it mean[[1]](#footnote-1). Setting the scrambler to all ones (all zeros is forbidden) is akin to yelling “I am the device that changed MAC address”
3. The scrambler is sometimes used to carry information. See the paragraph above this text. If the scrambler is reset, this information is lost.

Proposed Resolution:

1. Add a TXVECTOR parameter of SCRMABLER\_RESET. The scrambler reset should be based on this parameter.
2. Rather than reset the scrambler, set the scrambler value to a MIB variable: dot11ScramblerResetValue. This MIB variable should not have a default value, and it should be instantiation dependent. This means that each STA should have a value to which it resets the scrambler, independent of other STAs.

***Editor: Add the following parameter to the TX vector parameter:***

|  |  |  |
| --- | --- | --- |
| SCRAMBLER\_RESET | PHY-TXSTART.request (TXVECTOR) | Indicates that the scrambler shall be reset before the start of the PPDUEnumerated Type:RESET\_SCRAMBLER: The scrambler should be reset NO\_SCRAMBLER\_RESET: The scrambler should not be reset. |

***Editor: modify the text in P2906L60 (17.3.5.5) as follows:***

If SCRAMBLER\_RESET is set to RESET\_SCRAMBLER and dot11MACPrivacyActivated is true, the initial state of the scrambler shall be set to dot11ScramblerResetValue before changes based on CH\_BANDWIDTH\_IN\_NON\_HT define above are applied. ~~reset when the STA’s MAC address is changed.~~

***Editor: Modify the following line at P3856L25 (Dot11HTStationConfigEntry):***

dot11FortyMHzOptionImplemented TruthValue,

 dot11ScramblerResetValue BIT STRING}

***Editor: Add the following MIB variable specification at P3859L65 (end of dot11HTStationConfig):***

dot11ScramblerResetValue OBJECT-TYPE

SYNTAX OCTET STRING(Size(1))

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable. It is written by the SME or external management entity.

This is the value which the PHY uses to initialize the scrambler when instructed to do so using setting the TXVECTOR parameter SCRAMBLER\_RESET to RESET\_SCRAMBLER. It has no default value and shall be instantiation dependent"

DEFVAL { none }

::= { dot11HTStationConfigEntry 17 }

***Editor: Add the following text at the end 12.2.10 (P2518)***

When the MAC address is changed, the STA shall set the TXVECTOR parameter SCAMBLER\_RESET to RESET\_SCRAMBLER on the next transmitted PPDU.

**References:**

1. The idea behind the reset is clear. Most implementations will keep the scrambler shift register value from the end of the last transmitted packet to be used in the beginning of the next packet. This is random enough for purposes for which the scrambler is used, but it can tell an eavesdropper that this packet is from a specific STA. The question remains, what does reset mean. [↑](#footnote-ref-1)