IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolutions for some comments on 11me/D2.0 (LB270) | | | | |
| Date: 2022-12-05 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Mark RISON | Samsung Cambridge Solution Centre | SJH, CB4 0DS, U.K. | +44 1223 434600 | at samsung (a global commercial entity) I'm the letter emme then dot rison |

Abstract

This submission proposes resolutions for various CIDs on 11me/D2.0. Green indicates material agreed to in the group, yellow material to be discussed, red material rejected by the group and cyan material not to be overlooked. The “Final”/“No Markup” view should be selected in Word (this means Word comments can be disregarded by the Editor).

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3510  Mark RISON  9.4.2.17 | The Supported Channels element stuff is a mess.  Some bits of the spec suggest this element only applies to DFS operation  (e.g. Clause 6), others that that it doesn't apply if you support ECSA  (e.g. Clause 9). 9.2.4.17 even suggests it only applies to 11a!  But 9.6.7.16/12.2/12.3 indicates it's also used for TDLS. 11.8.2/8  suggest it's about letting the AP decide whether to allow a STA to  associate, based on the channels it supports, and if so what channels  to use when changing channels. 11.20.6 says that for TDLS you include  both that a Supported Operating Classes element, but it doesn't describe  how you determine which OC in the SOCe a given channel in the SCe  applies to | At the start of the referenced subclause add "The Supported Channels element means completely different things in different contexts." |

Discussion:

Here are the key instances of “Supported Channels”:

**4.5.5.3 DFS**

[…]

The DFS service provides for the following:

— Association of STAs with an AP based on the STAs’ supported channels.

**4.5.8 Radio measurement service**

The Radio measurement service provides the following:

— The ability to request and report radio measurements in supported channels.

— The ability to perform radio measurements in supported channels.

**6.5.7.2.2 Semantics of the service primitive [ditto reassoc]**

The primitive parameters are as follows:

MLME-ASSOCIATE.request(

[…]

Supported Channels

The list of channels in which the STA is capable of operating.

Present if DFS functionality is required, as specified in 11.8 (DFS procedures); otherwise not present.

=> DFS-only

**Table 9-62—Association Request frame body [ditto reassoc]**

The Supported Channels element is present if dot11SpectrumManagementRequired is true and dot11ExtendedChannelSwitchActivated is false.

=> suggests not present if ECSA supported; at least not required

**9.4.2.17 Supported Channels element**

The Supported Channels element contains a list of channel subbands (from those channels defined in 17.3.8.4.3 (Channelization)) in which a STA is capable of operating.

=> 11a-only

[…]

The use of the Supported Channels element is described in 11.8.2 (Association based on supported channels) and 11.8.8 (Selecting and advertising a new channel).

=> about letting the AP decide whether to allow a STA to associate based on the channels it supports, and what channel to switch to

**9.4.2.20.7 Beacon request**

[…]

For operating classes that identify the location of the primary channel, a Channel Number field set to 0 indicates a request to make iterative measurements for all supported channels in the operating class where the measurement is permitted on the channel and the channel is valid for the current regulatory domain.

For operating classes that encompass a primary channel but do not identify the location of the primary channel, a Channel Number field set to 0 indicates a request to make iterative measurements for all primary channel positions within all requested and supported channels where the measurement is permitted on the channel and the channel is valid for the current regulatory domain.

For operating classes that identify the location of the primary channel, a Channel Number field set to 255 indicates a request to make iterative measurements for all supported channels in the current operating class listed in the latest AP Channel Report received from the serving AP.

**Table 9-457—TDLS Discovery Response frame Action field format [ditto setup req]**

[…]

The Supported Channels element is present if the TDLS Channel Switching subfield is equal to 1.

The Supported Channels element is defined in 9.4.2.17 (Supported Channels element).

=> also used for TDLS

**Table 9-498—Information for TDLS Setup Response Action field**

[…]

The Supported Channels element is defined in 9.4.2.17 (Supported Channels element). It is present if the TDLS Channel Switching subfield is equal to 1 and the Status Code is SUCCESS, and not present otherwise.

=> also used for TDLS

**Table 9-519—Mesh Peering Open frame Action field format**

[…]

The Supported Channels element is present if dot11SpectrumManagementRequired is true and dot11ExtendedChannelSwitchActivated is false.

=> suggests not present if ECSA supported; at least not required

**10.39.12.2.1 Channel access rules**

[…]

An EDMG STA shall not transmit an EDMG PPDU to a peer EDMG STA over a channel that is not supported by the peer STA as indicated in the Supported Channels field [I think subelement is intended here] in the peer STA’s EDMG Capabilities element.

**11.8 DFS procedures**

**11.8.1 General [ditto DMG]**

[…]

Attribute dot11SpectrumManagementRequired shall be set to true (#1038)if regulatory authorities require DFS. It may also be set to true in other circumstances. The DFS procedures provide for the following:

— Associating STAs with an AP based on the STAs’ supported channels (see 11.8.2 (Association based on supported channels)).

=> about letting the AP decide whether to allow a STA to associate based on the channels it supports

**11.8.2.1 Association based on supported channels in a non-DMG BSS [similar DMG]**

A STA shall provide an AP with a list of the channels in which the STA can operate when associating or reassociating by including a Supported Channels element in its (Re)Association Request frames.

=> about letting the AP decide whether to allow a STA to associate based on the channels it supports

An AP may use the supported channels list for associated STAs as an input into an algorithm used to select a new channel for the BSS. The specification of this algorithm is beyond the scope of this standard.

=> about helping the AP choose which channel to switch to

**11.8.8.2 Selecting and advertising a new channel in a non-DMG infrastructure BSS [similar DMG]**

[…]

The decision to switch to a new operating channel in an infrastructure BSS shall be made only by the AP. An AP may make use of the information in Supported Channel [should be Channels, also in MBSS and DMG] elements and the results of measurements undertaken by the AP and other STAs in the BSS to assist the selection of the new channel. The algorithm to choose a new channel is beyond the scope of this standard. The AP shall attempt to select a new channel that is supported by all associated STAs.

=> about helping the AP choose which channel to switch to

**11.9.3.2 Selecting and advertising a new channel in an infrastructure BSS [similar MBSS]**

[…]

The decision to switch to a new operating channel and/or operating class in an infrastructure BSS is made by the AP when dot11DSERequired is false. An AP may make use of the information in the Supported Channels element, Supported Operating Classes element, and the results of measurements undertaken by the AP and other STAs in the BSS to assist the selection of the new channel and/or operating class.

=> about helping the AP choose which channel to switch to

**11.20.6.1 General**

The STA shall include a Supported Channels element and a Supported Operating Classes element in all TDLS Setup Request and TDLS Setup Response frames that have a TDLS Channel Switching subfield equal to 1. The STA shall include only channels in the Supported Channels element for which it can adhere to the local power constraint.

=> how do you determine which OC in the SOCe a given channel in the SCe applies to?

**B.4.10 Spectrum management extensions**

SM3 Power Capability and Supported Channels elements in (Re)Association Request and Response frames

CFSM:M

=> so not if spectrum management not supported

My starting suggestion would be:

* the bits that suggest it's 11a-only or DFS-only or SM-only are just wrong
* the bits that suggest it's not used if ECSA is supported are misleading
* the combined use of SC and SOC in TDLS needs clarification (or just delete the SOC from the TDLS frames?)
* the EDMG typo should be fixed
* the Supported Channel element typos should be fixed

Proposed changes:

Make the following changes:

**6.5.7.2.2 Semantics of the service primitive**

The primitive parameters are as follows:

MLME-ASSOCIATE.request(

[…]

Supported Channels

The list of channels in which the STA is capable of operating.

Present if DFS functionality is required, as specified in 11.8 (DFS procedures); otherwise ~~not~~optionally present.

**6.5.8.2.2 Semantics of the service primitive**

The primitive parameters are as follows:

MLME-REASSOCIATE.request(

[…]

Supported Channels

The list of channels in which the STA is capable of operating.

Present if DFS functionality is required, as specified in 11.8 (DFS procedures); otherwise ~~not~~optionally present.

**Table 9-62—Association Request frame body**

The Supported Channels element is present if dot11SpectrumManagementRequired is true and dot11ExtendedChannelSwitchActivated is false.

The Supported Channels element is optionally present, otherwise.

**Table 9-64—Reassociation Request frame body**

The Supported Channels element is present if dot11SpectrumManagementRequired is true and dot11ExtendedChannelSwitchActivated is false.

The Supported Channels element is optionally present, otherwise.

**9.4.2.17 Supported Channels element**

The Supported Channels element contains a list of channel subbands ~~(from those channels defined in 17.3.8.4.3 (Channelization))~~ in which a STA is capable of operating.

[…]

~~The use of the Supported Channels element is described in 11.8.2 (Association based on supported channels) and 11.8.8 (Selecting and advertising a new channel).~~

**Table 9-519—Mesh Peering Open frame Action field format**

The Supported Channels element is present if dot11SpectrumManagementRequired is true and dot11ExtendedChannelSwitchActivated is false.

The Supported Channels element is optionally present, otherwise.

**10.39.12.2.1 Channel access rules**

An EDMG STA shall not transmit an EDMG PPDU to a peer EDMG STA over a channel that is not supported by the peer STA as indicated in the Supported Channels ~~field~~subelement in the peer STA’s EDMG Capabilities element.

**11.8.8.2 Selecting and advertising a new channel in a non-DMG infrastructure BSS**

The decision to switch to a new operating channel in an infrastructure BSS shall be made only by the AP. An AP may make use of the information in Supported Channels elements and the results of measurements undertaken by the AP and other STAs in the BSS to assist the selection of the new channel.

**11.8.8.4.1 General**

A mesh STA may make use of the information in Supported Channels elements, Supported Operating Classes

elements, and the results of measurements undertaken by the mesh STAs in the MBSS to assist the selection of

the new channel.

**11.8.8.6 Selecting and advertising a new channel in a DMG BSS**

The decision to switch to a new operating channel in a DMG BSS shall be made only by an AP or PCP. An AP

or PCP may make use of the information in received Supported Channels elements and the results of

measurements undertaken by the AP or PCP and other STAs in the BSS to assist the selection of the new

channel.

**11.20.6.1 General**

The STA shall include a Supported Channels element and a Supported Operating Classes element in all TDLS Setup Request and TDLS Setup Response frames that have a TDLS Channel Switching subfield equal to 1. The Supported Channels element shall indicate the channels supported for the current operating class. The STA shall include only channels in the Supported Channels element for which it can adhere to the local power constraint.

**B.4.10 Spectrum management extensions**

SM3 Power Capability and Supported Channels elements in (Re)Association Request and Response frames

CFSM:M

NOT CFSM:O

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3510 in <this document>, which clarify the usage of the Supported Channels element.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3338  Mark RISON | Sometimes it's "The BSS Load element is optionally present if  dot11QosOptionImplemented and dot11QBSSLoadImplemented" and sometimes "is present". Also a DMG STA is a QoS STA | At 737.31 change "The BSS Load element is optionally present if  dot11QosOptionImplemented and dot11QBSSLoadImplemented are  both true; otherwise not present.(#1598-Ed1)" to "The BSS Load element is present if  dot11QBSSLoadImplemented is true; otherwise not present.(#1598-Ed1)" |

Discussion:

Solomon TRAININ has clarified that:

The general approach in the DMG is to have an alternative way to deliver elements. This element same like some others may be delivered in the Announce frame instead or in addition to the DMG beacon

[Note: it doesn’t in fact seem to be possible to deliver a BSS Load element in an Announce frame. Something for D3.0!]

So the element should remain optional in DMG beacons. But it’s mandatory in Probe Responses sent by a DMG AP (or indeed a PCP).

It is also the case that DMG APs are QoS STAs, so checking dot11QosOptionImplemented is otiose. Ditto for VHT APs.

Dave GOODALL has clarified that:

It is optional for S1G to support BSS Load. We would typically put the element in a Probe Response but it is optional to put it in an S1G Beacon.

So this is the same behaviour as for DMG.

Proposed changes:

Change at 697.25 (in Table 9-60—Beacon frame body) and 725.15 (in Table 9-67—Probe Response frame body):

The Extended BSS Load element is optionally present if

~~dot11QosOptionImplemented,~~ dot11QBSSLoadImplemented~~,~~ and

dot11VHTOptionImplemented are both true.

Change at 737.31 (in Table 9-73—DMG Beacon frame body):

The BSS Load element is optionally present if

~~dot11QosOptionImplemented and~~ dot11QBSSLoadImplemented ~~are~~

~~both~~is true; otherwise not present.

Change at 4903.3 (in C.3 MIB):

dot11QBSSLoadImplemented OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This is a capability variable.

Its value is determined by device capabilities.

This attribute is available only at a~~n~~ QoS AP or PCP. This attribute, when true, indicates that the AP or PCP ~~implementation is capable of generating and transmitting~~transmits the BSS ~~l~~Load element in ~~the~~ Beacon frames (non-DMG non-S1G AP only) and Probe Response frames, and optionally in DMG Beacon frames (DMG AP or PCP only) and S1G Beacon frames (S1G AP only); optionally transmits the Extended BSS Load element in Beacon and Probe Response frames (VHT AP only); and optionally transmits the HE BSS Load element in Beacon and Probe Response frames (HE AP only). These elements are not transmitted~~The capability is disabled~~, otherwise."

::= { dot11StationConfigEntry 36 }

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3338 in <this document>, which make changes in the suggested direction (including recognising VHT and DMG APs as necessarily QoS APs), but keep the BSS Load element optional in DMG beacons. The description of the dot11QBSSLoadImplemented attribute is aligned with Clause 9.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3541  Mark RISON  C.3 | Can read-create MIB attributes have a DEFVAL? What does read-create mean anyway? | Delete DEFVAL lines for MIB attributes that are read-create |

Discussion:

Here are examples:

dot11PPEThresholdsMappingStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The status column used for creating, modifying, and deleting instances of the columnar objects in the PPE thresholds mapping table."

DEFVAL { active }

::= { dot11PPEThresholdsMappingsEntry 6 }

dot11RMRqstDuration OBJECT-TYPE

SYNTAX Unsigned32

UNITS "TUs"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity when requesting a

measurement.

Changes take effect when dot11RMRqstRowStatus is set to Active.

This attribute indicates the preferred or mandatory measurement duration

for this Measurement Request. This attribute is ignored if dot11RMRqstType = LCI or Measurement Pause."

DEFVAL { 0 }

::= { dot11RMRequestEntry 12 }

RFC 2578 (Structure of Management Information Version 2 (SMIv2)) says:

The MAX-ACCESS clause, which must be present, defines whether it

makes "protocol sense" to read, write and/or create an instance of

the object, or to include its value in a notification. This is the

maximal level of access for the object. (This maximal level of

access is independent of any administrative authorization policy.)

The value "read-write" indicates that read and write access make

"protocol sense", but create does not. The value "read-create"

indicates that read, write and create access make "protocol sense".

The value "not-accessible" indicates an auxiliary object (see Section

7.7). The value "accessible-for-notify" indicates an object which is

accessible only via a notification (e.g., snmpTrapOID [5]).

These values are ordered, from least to greatest: "not-accessible",

"accessible-for-notify", "read-only", "read-write", "read-create".

If any columnar object in a conceptual row has "read-create" as its

maximal level of access, then no other columnar object of the same

conceptual row may have a maximal access of "read-write". (Note that

"read-create" is a superset of "read-write".)

and:

The DEFVAL clause, which need not be present, defines an acceptable

default value which may be used at the discretion of an agent when an

object instance is created. That is, the value is a "hint" to

implementors.

During conceptual row creation, if an instance of a columnar object

is not present as one of the operands in the correspondent management

protocol set operation, then the value of the DEFVAL clause, if

present, indicates an acceptable default value that an agent might

use (especially for a read-only object).

Note that with this definition of the DEFVAL clause, it is

appropriate to use it for any columnar object of a read-create table.

It is also permitted to use it for scalar objects dynamically created

by an agent, or for columnar objects of a read-write table

dynamically created by an agent.

and:

For newly-defined conceptual rows which allow the creation of new

object instances and/or the deletion of existing object instances,

there should be one columnar object with a SYNTAX clause value of

RowStatus (a textual convention defined in [[3](https://www.rfc-editor.org/rfc/rfc2578.html#ref-3)]) and a MAX-ACCESS

clause value of read-create. By convention, this is termed the

status column for the conceptual row.

Proposed resolution:

REJECTED

RFC 2578 specifies that read-create means that “read, write and create access make "protocol sense"” (and requires that read-write not be used for a table if read-create is used). It also specifies that a DEFVAL can be used “for any columnar object of a read-create table”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3290  Mark RISON | "ERP-DSSS modes" is not clear since ERP-DSSS is itself defined as a mode of a PHY. Ditto CCK and OFDM | Use better terminology |

Discussion:

Instances (note no instances of “ERP-CCK mode”):

18.1.3: For example, a BSS could operate in an ERP-OFDM-only mode, a mixed mode of ERP-OFDM and ERP-DSSS/CCK, or a mixed mode of ERP-DSSS/CCK and non-ERP. When options are enabled, combinations are also allowed. [Arguably is about the mode of the BSS]

18.3.2.4: The format, preamble, and headers for the ERP-OFDM PPDU are described in 17.3.2 (PPDU format) to 17.3.5 (DATA field). For the ERP-OFDM modes, the DATA field that contains the SERVICE field, the PSDU, the TAIL bits, and the PAD bits shall follow 17.3.5 (DATA field).

For ERP-OFDM modes, an ERP PPDU is (#14)immediately followed by a period of no transmission with a duration of aSignalExtension(#14)

18.4.7.3: The transmit spectral mask for the ERP-OFDM modes shall follow 17.3.9.3 (Transmit spectrum mask) and is shown in Figure 17-13 (Transmit spectrum mask for 20 MHz transmission) therein. The transmit spectral mask for the ERP-DSSS modes shall follow 16.3.7.4 (Transmit spectrum mask) and is shown in Figure 16-8 (Transmit spectrum mask) therein.

18.4.8.1: Subclause 18.4.8 (PHY receive specifications) describes the receive specifications for the PHY sublayer. The receive specification for the ERP-OFDM modes shall follow 17.3.10 (PHY receiver specifications) with the exception of the receiver maximum input level (17.3.10.5 (Receiver maximum input level)) and the adjacent channel rejection (17.3.10.3 (Adjacent channel rejection)). The receive specifications for the ERP-DSSS modes shall follow 16.3.8 (PHY receiver specifications) with the exception of the receiver maximum input level (16.3.8.3 (Receiver maximum input level)).

18.4.8.2: The PER of the ERP-OFDM modes shall be less than 10% at a PSDU length of 1000 octets for the input levels of Table 17-18 (Receiver performance requirements) of 17.3.10 (PHY receiver specifications). […] The PER for ERP-CCK shall be as specified in 16.3.8.2 (Receiver minimum input level sensitivity).

18.4.8.3: The adjacent channel rejection of the ERP-DSSS modes shall follow 16.3.8.4 (Receiver adjacent channel

rejection).

Proposed changes:

18.1.3: For example, a BSS could operate in an ERP-OFDM-only mode, a mixed mode of ERP-OFDM and ERP-DSSS/CCK, or a mixed mode of ERP-DSSS/CCK and non-ERP. When options are enabled, combinations are also allowed. *[No change]*

18.3.2.4: The format, preamble, and headers for the ERP-OFDM PPDU are described in 17.3.2 (PPDU format) to 17.3.5 (DATA field). For ERP-OFDM PPDUs, the DATA field that contains the SERVICE field, the PSDU, the TAIL bits, and the PAD bits shall follow 17.3.5 (DATA field).

An ERP-OFDM PPDU is (#14)immediately followed by a period of no transmission with a duration of aSignalExtension(#14)

18.4.7.3: The transmit spectral mask for ERP-OFDM PPDUs shall follow 17.3.9.3 (Transmit spectrum mask) and is shown in Figure 17-13 (Transmit spectrum mask for 20 MHz transmission) therein. The transmit spectral mask for ERP-DSSS PPDUs shall follow 16.3.7.4 (Transmit spectrum mask) and is shown in Figure 16-8 (Transmit spectrum mask) therein.

18.4.8.1: Subclause 18.4.8 (PHY receive specifications) describes the receive specifications for the PHY sublayer. The receive specification for ERP-OFDM PPDUs shall follow 17.3.10 (PHY receiver specifications) with the exception of the receiver maximum input level (17.3.10.5 (Receiver maximum input level)) and the adjacent channel rejection (17.3.10.3 (Adjacent channel rejection)). The receive specification for ERP-DSSS PPDUs shall follow 16.3.8 (PHY receiver specifications) with the exception of the receiver maximum input level (16.3.8.3 (Receiver maximum input level)).

18.4.8.2: The PER for ERP-OFDM PPDUs shall be less than 10% at a PSDU length of 1000 octets for the input levels of Table 17-18 (Receiver performance requirements) of 17.3.10 (PHY receiver specifications). […] The PER for ERP-CCK PPDUs shall be as specified in 16.3.8.2 (Receiver minimum input level sensitivity).

18.4.8.3: Adjacent channels at 2.4 GHz are defined to be at ± 25 MHz spacing. ***<para break>***

For ERP-OFDM PPDUs, the adjacent channel rejection shall be measured by setting the desired signal’s strength 3 dB above the rate-dependent sensitivity specified in Table 17-18 (Receiver performance requirements) of 17.3.10 (PHY receiver specifications) […]

The adjacent channel rejection for ERP-DSSS PPDUs shall follow 16.3.8.4 (Receiver adjacent channel

rejection).

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3290 in <this document>, which refer to ERP PPDUs rather than modes, and add flavours where missing.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3719  Mark RISON  C.3 | dot11RSNAConfigGroupRekeyPackets should be stated to apply only if dot11RSNAConfigGroupRekeyMethod is packetBased(3) or timepacketBased(4) | As it says in the comment |

Discussion:

We have:

dot11RSNAConfigGroupRekeyTime OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity.

Changes take effect as soon as practical in the implementation.

The time after which the (#1980)(#1509)GTK is (#1382)refreshed (by rekeying). The timer starts at the moment the GTK was set using the MLME-SETKEYS.request primitive."

DEFVAL { 86400 } -- once per day

::= { dot11RSNAConfigEntry 6 }

dot11RSNAConfigGroupRekeyPackets OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

UNITS "1000 packets"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity.

Changes take effect as soon as practical in the implementation.

A packet count after which the (#1980)(#1509)GTK is (#1382)refreshed (by rekeying). The packet counter starts at the moment the GTK was set using the MLME-SETKEYS.request primitive and it counts all packets encrypted using the current GTK."

::= { dot11RSNAConfigEntry 7 }

but these only apply for certain values of dot11RSNAConfigGroupRekeyMethod.

Proposed changes:

Change from 4952.15 as follows:

dot11RSNAConfigGroupRekeyTime OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

UNITS "seconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity.

Changes take effect as soon as practical in the implementation.

The time after which the (#1980)(#1509)GTK is (#1382)refreshed (by rekeying), if dot11RSNAConfigGroupRekeyMethod is timeBased(2) or timepacketBased(4). The timer starts at the moment the GTK was set using the MLME-SETKEYS.request primitive."

DEFVAL { 86400 } -- once per day

::= { dot11RSNAConfigEntry 6 }

dot11RSNAConfigGroupRekeyPackets OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

UNITS "1000 packets"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity.

Changes take effect as soon as practical in the implementation.

~~A~~The packet count after which the (#1980)(#1509)GTK is (#1382)refreshed (by rekeying), if dot11RSNAConfigGroupRekeyMethod is packetBased(3) or timepacketBased(4). The packet counter starts at the moment the GTK was set using the MLME-SETKEYS.request primitive and it counts all packets encrypted using th~~e current~~at GTK."

::= { dot11RSNAConfigEntry 7 }

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3719 in <this document>, which make the changes requested and also clarify the behaviour following rekeying.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3409  Mark RISON | CID 1927: add a NOTE to say that the SCRAMBLER\_INITIAL\_VALUE is equal to the value in the SERVICE field after scrambling | As it says in the comment |

Discussion:

I think Youhan KIM presented some figures to demonstrate that the initial value of the scrambler (which is what the transmitter cares about) is also the value in the SERVICE field after scrambling (which is what the receiver cares about) is. However, I can’t find these (not in 22/0990 for instance).

In any case, the point is that the receiver doesn’t care about the initial value of the scrambler at the transmitter, the receiver cares about the value in the SERVICE field after scrambling (because that’s what’s used for e.g. MU-RTS/CTS). In general this is made clear through wording like “(the first 7 bits received in the SERVICE field prior to descrambling)”, but not always.

Proposed changes:

Change from 3180.45 as follows:

During reception, an HE STA shall generate the RXVECTOR parameter SCRAMBLER\_INITIAL\_VALUE as the ~~integer representation of the first 7 bits of the scrambling sequence, with the first bit of the scrambling sequence being the LSB of the SCRAMBER\_INITIAL\_VALUE~~ value in the first 7 bits of the SERVICE field prior to descrambling.

Change “Service field” to “SERVICE field” at 559.36/37, 1739.44, 1793.23, 3579.10, 3658.43, 3659.20, 3687.39.

Delete the second “]” at 559.37.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3409 in <this document>, which clarify that on rx the value is the SERVICE field prior to descrambling, and fix the case of “SERVICE field”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3795  Xiangxin GU  B.4.4.2  4651.16 | not CFDMG:O or not CFDMG:M for CF-End frame? | as commented |

Discussion:

It is indeed the case that a M/O indication is missing:



CF-End can be used to truncate a TXOP under EDCA, but I am not aware of any situations in which CF-End transmission is mandatory, at least when disregarding deprecated/obsolete features. I find evidence that CF-End can be used with DMG, so I am not sure what the “not CFDMG” pertained to. However, FR16 is a reminder that CF-End does not apply to OCB operation.

Proposed resolution:

REVISED

At the referenced location change “not CFDMG” to “NOT CFOCB:O”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3321  Mark RISON  C.3 | The DESCRIPTION of dot11SPIdleTimeout is far too generic. This is a DMG service period thing only | As it says in the comment |

Discussion:

dot11SPIdleTimeout is only used in these subclauses:

* 10.39.6.7 Service period recovery (under 10.39.6 Channel access in scheduled DTI (under 10.39 DMG and CMMG channel access))
* 11.3.5.2 Non-AP and non-PCP STA association initiation procedures [in a DMG list item]
* 11.3.5.4 Non-AP and non-PCP STA reassociation initiation procedures [in a DMG list item]

However, only DMG is mentioned, not CMMG, in those subclauses, so this seems to be a DMG-only thing.

Proposed changes:

Change from 5283.40 as follows:

dot11SPIdleTimeout OBJECT-TYPE

SYNTAX Unsigned32 (1..100000)

UNITS "microseconds"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by the SME or an external management entity.

Changes take effect as soon as practical in the implementation.

This attribute~~e SPIdleTimeout subfield~~ indicates the time during which a DMG STA expects to receive a frame from its partner STA."

DEFVAL { 200 }

::= { dot11DMGOperationEntry 9 }

Proposed resolution:

REVISED

At 5283.50 change “The SPIdleTimeout subfield indicates time during which a STA” to “This attribute indicates the time during which a DMG STA”.

At 5283.34 change “The MinPPDuration subfield” to “This attribute”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3200  Mark RISON  C.3 | "The default is" should not appear in the DESCRIPTION, but in a DEFVAL | As it says in the comment |

Discussion:

As it says in the comment, we have DEFVAL for defaults.

Proposed changes:

REVISED

In dot11StationID at 4894.59 delete “Its syntax is MAC address, and the default value is the STA's assigned, unique MAC address.” and after the DESCRIPTION add “DEFVAL { the STA's MAC address }”.

In dot11RMMeasurementPilotPeriod at 4907.3 delete “The default period is 25% of dot11BeaconPeriod.” and after the DESCRIPTION add “DEFVAL { dot11BeaconPeriod / 4 }”.

In dot11RSNAConfigGroupRekeyMethod at 4951.55 delete “The default is time-based, once per day.”

In dot11EDMGOptionImplemented at 5238.61 delete “The default value of this attribute is false.”

In dot11QAPEDCATableMandatory at 5275.57 delete “The default value for this parameter is false.” and after the DESCRIPTION add “DEFVAL { false }”.

In dot11NonAPStationAuthMaxVoiceRate at 5378.41 delete “, which is the default value,”. Ditto for dot11NonAPStationAuthMaxVideoRate, dot11NonAPStationAuthMaxBestEffortRate, dot11NonAPStationAuthMaxBackgroundRate, dot11NonAPStationAuthMaxHCCAHEMMRate, dot11NonAPStationAuthHCCAHEMMDelay, dot11NonAPStationAuthMaxSourceMulticastRate.

What about things like

See Table 9-192

(Default EDCA Parameter Set element (#1660)parameters

if dot11OCBActivated is false and the STA is a non-sensor STA(#251)) and

Table 9-193 (Default EDCA parameter set for STA operation if dot11OCBActi-

vated is true).

?

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3493  Mark RISON | The interpretation of the Key ID field in KDEs is not specified | Add sentences of the form "The Key ID field contains ..." |
| CID 3494  Mark RISON | The interpretation of some of the fields in KDEs is not specified, e.g. GTK in GTK KDE, MAC Address in MAC Address KDE, | Add sentences of the form "The <blah> field contains ..." |

Discussion:

As these comments say, some KDE fields are not described. Ditto some TK subelement fields.

Also, “Key ID” should have lowercase “key” when not about the name of a field etc., and not at the start of a sentence etc.

Miscellaneous additional small horrors have come to light in the course of examining these issues.

Proposed changes:

At 1038.46 add a para “The Key ID field contains the GTK key ID.”

At 2905.6 add a para “The Key ID field contains the GTK key ID.”

At 2905.12 add a para “The GTK field contains the GTK.”

At 2905.22 add a para “The MAC Address field contains a MAC address.”

At 2905.31 add a para “The PMKID field contains a PMKID.”

At 2905.40 add a para “The Key Nonce field contains a key nonce.”

At 2905.41 change “The Key Lifetime value is expressed in seconds and uses big endian octet order.” to “The Key Lifetime field contains a key lifetime in seconds, in big endian octet order.”

At 2905.45 delete “(in seconds)”.

At 2905.51 change “The Error Type field is in big endian octet order.” to “The Error Type field contains an error type, in big endian octet order.”

At 2906.12 add three paras “The Key ID field contains the IGTK key ID.”, “The IPN field contains the IPN.” and “The IGTK field contains the IGTK.” Move the sentence “The IPN corresponds…” above to the end of the second new para.

At 2908.6 add a para “The Key ID field contains the BIGTK key ID.”

At 2908.10 add a para “The BIGTK field contains the BIGTK.”

At 2908.21 add a para “The Key ID field contains the WIGTK key ID.”

At 2908.26 add a para “The WIGTK field contains the WIGTK.”

At 1038.35 change “The GTK subelement Key Info subfield” to “The Key Info subfield”.

At 2908.16 change “(Length-12)” to “variable”.

At 2908.18 change “WIGTK KDE” to “WIGTK KDE format”.

At 1039.16 change “indicates the value of the BIP key identifier” to “contains the IGTK key ID”.

At 1040.1 change “indicates the value of the BIGTK identifier” to “contains the BIGTK key ID”.

At 1040.25 change “indicates the WIGTK identifier” to “contains the WIGTK key ID”.

At 1634.15 change “indicates the value of the BIP key identifier” to “contains the IGTK key ID”.

At 1634.41 change “indicates the value of the BIGTK identifier” to “contains the BIGTK key ID”.

Change “Key ID” to “key ID” at 492.5/6, 978.56/57, 1046.11(2x)/12, 1159.1, 1320.54, 1321.14 (rightmost), 1476.47 (rightmost), 1722.64, 2313.49, 2841.8/10, 2847.31 (rightmost, also delete “value”), 2859.49/58/59, 2871.8, 2876.60/61, 2881.43/44/45/48/50/51/53, 2907.18 (rightmost), 2917.14/15/18(2x)/63, 3034.41, 4572.17(2x, also delete “value”)/53/54(also delete “value”), 4573.13/14 (also delete “value”), 4575.50/56, 4576.2,

At 338.57/59 change “key identifier” to “PMK identifier”.

Change at 1258.53 as follows:

The Number of Public Key Identifiers subfield ~~lists~~contains the number of Public Key Identifier fields that are present in the Public Key Identifiers field in the FILS Indication element. When the Number of Public Key Identifiers subfield is 0, the Public Key Identifiers field is not present in the FILS Indication element. Each Public Key Identifier field is formatted per Figure 9-729 (Public Key Identifier field format). Up to seven Public Key Identifier~~s~~ fields may be carried in a FILS Indication element.

At 1258.24 change “Public Key Identifier” to “Public Key Identifiers”.

Change at 2911.16 as follows:

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

RSNE is the RSNE, described in 9.4.2.24 (RSNE)

RSNE[KeyName] is the RSNE, with the PMKID List field set to KeyName

GTK[N] is the GTK KDE, with the ~~k~~Key ID~~identifier~~ field set to N (The key ID~~identifier~~ specifies which index is used for this GTK. Index 0 shall not be used for GTKs, except in mixed environments, as described in 12.7.1 (Key hierarchy). (#1457)Index 3 shall not be used for GTKs)

FTE is the (#1776)FTE, described in 9.4.2.47 (FTE(#1776))

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

TIE[IntervalType] is a (#1776)TIE of type IntervalType, as described in 9.4.2.48 (TIE(#1776)), containing e.g., for type KeyLifetime, the lifetime of the FT key hierarchy

IGTK[M] is the IGTK KDE, with the ~~k~~Key ID~~identifier~~ field set to M

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

BIGTK[Q] is the BIGTK KDE, with the ~~k~~Key ID~~identifier~~ field set to Q

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

(11ba)WIGTK[R] is the WIGTK KDE, with the ~~k~~Key ID~~identifier~~ field set to R

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

PMKID is ~~of type~~the PMKID KDE and is the ~~key~~PMK identifier used during the 4-way ~~PTK~~ handshake for PMK identification

OCI KDE is ~~a~~the OCI KDE ~~containing operating channel information~~

RSNXE is the RSNXE, described in 9.4.2.241 (RSNXE(#1776))

PMKID is the PMK identifier for~~identifies~~ the PMKSA selected by the Authenticator

“{a} or {b}” means that exactly one of either {a} or {b} is present as the {Key Data}

Change “key identifier” to “key ID” at 2912.42, 2916.25/50, 2922.50/54/58/62, 2923.61/62, 2924.1/3.

Change “TK identifier” to “TK key ID” at 1752.46/49, 2845.53/57/60.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3493 and 3494 in <this document>, which make the changes requested by the commenter, and fix related issues.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3631  Mark RISON  12 | The setting of the Secure bit in M1 and M2 of a rekeying 4WH is not clear | Specify that it is indeterminate, and hence must be ignored by the receiver |
| CID 3596  Mark RISON  12.7.6.1 | It is not clear whether 12.7.6.1 General applies to rekeying, e.g. whether the Secure bit is 0 in M1 and M2 even in rekeying (when it has the old PTK and GTK). Ditto 12.7.2.b)7) Note that if the Secure bit is 1 in M2 rekeying then the Key Information field is the same for M2 and M4 | Clarify the rules for rekeying |

Discussion:

In 12.7.2 we have:

Secure (bit 9) indicates whether the Authenticator and the Supplicant share a PTKSA. It is set to 0 in messages 1 and 2 of the 4-way handshake if the Authenticator and the Supplicant do not share a PTKSA. Otherwise, it is set to 1.

NOTE 1—Some deployed Authenticator and Supplicant implementations set the Secure bit to 0 in messages 1 and 2 of the 4-way handshake that is used for PTK rekeying even when they already share a previously generated PTKSA. In the interests of interoperability, implementations might ignore the Secure bit in received frames.

NOTE 2—The Secure bit is set to 1 in messages 3 and 4 of the 4-way handshake.

I think we ended up with NOTE 1 as a compromise between saying “it shall be 1 when PTK rekeying (because a PTKSA exists at that point)” and “it shall be ignored when PTK rekeying (because some existing implementations set it to 0)”.

However, 12.7.6.1 suggests the Secure bit is always set to 0 in M1 and M2:

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

And in fact so does 12.7.6.2/3:

Message 1 uses the following values for each of the (#1836)EAPOL-Key PDU fields:

[…]

Secure = 0

Message 2 uses the following values for each of the (#1836)EAPOL-Key PDU fields:

[…]

Secure = 0 – same as message 1

And in fact various state machines assume 0. And indeed fail to cover all the messages!

Also, as CID 3596 points out, the statements about how you can distinguish M1-M4 and G1-G2 are no longer correct.

Proposed changes:

Change 12.7.6.1 as follows:

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

Message 2: Supplicant → Authenticator: EAPOL-Key(0 or 1,1,0,0,P,0,0,SNonce,MIC,{RSNE} or

{RSNE, OCI KDE} or {RSNE, RSNXE} or {RSNE, OCI KDE, RSNXE})

Change 12.7.6.2 as follows:

Message 1 uses the following values for each of the (#1836)EAPOL-Key PDU fields:

[…]

Secure = 0 in initial 4-way handshake, or 1 when PTK rekeying (but see 12.7.2)

Change 12.7.6.3 as follows:

Message 2 uses the following values for each of the (#1836)EAPOL-Key PDU fields:

[…]

Secure = 0 in initial 4-way handshake, or 1 when PTK rekeying (but see 12.7.2) ~~– same as message 1~~

In the list in 12.7.9.3 Supplicant state machine variables immediately before SNonce add:

— *PTKSAEstablished*. The Supplicant sets this variable to 0 on initialisation and to 1 when a PTKSA has been established.

Change at 2935.52 from:

**if** *A* = 1 && *State* ≠ Failed **then**

Send EAPOL-Key(0,1,0,0,K,0,0,TSNonce,MIC(TPTK),{RSNE})

**endif**

**if** UpdatePTK = 1 **then**

*MLME-SETPROTECTION.request(TA, Rx\_Tx)*

**endif**

to:

**if** *State* ≠ Failed **then**

**if** *K* = P && *M* = 0 && *A* = 1 **then**

Send EAPOL-Key(*PTKSAEstablished*,1,0,0,P,0,0,TSNonce,MIC(*TPTK*),{RSNE}) // M2

**else if** *K* = P && *M* = 1 && *A* = 1 **then**

Send EAPOL-Key(1,1,0,0,P,0,0,0,MIC(*TPTK*),{}) // M4

**else if** *K* = G && *A* = 1 **then**

Send EAPOL-Key(1,1,0,0,G,0,0,0,MIC,{}) // G2

**else**

*State* ← FAILED

**endif**

**endif**

**if** *State* ≠ Failed && *UpdatePTK* = 1 **then**

*MLME-SETPROTECTION.request(TA, Rx\_Tx)*

*PTKSAEstablished* ← 1

**endif**

At 2934.39 romanise “*P”*.

At 2934.42 change “–“ to “//”.

At 2935.28 change “*KeyData* = GTK” to “*K* = G”.

At 2937.28 change

Send EAPOL-Key( 0, 0, 1, 0, P, 0, 0, ANonce, 0, {})

to:

Send EAPOL-Key(PTK = 0 ? 0 : 1, 0, 1, 0, P, 0, 0, ANonce, 0, {}) // M1

At 2937.44 append “ // M3” after “Send EAPOL-Key(1, 1, 1, Pair, P, 0, RSC, ANonce, MIC(PTK),

{RSNE, GTK[N], IGTK [M]})”.

At 2939.16 append “ // G1” after “Send EAPOL-Key (1, 1, 1, !Pair, G, 0, RSC, 0, MIC(PTK), {GTK[GN]})” and delete the space before the opening paren.

Change 12.7.6.8 as follows:

It is critical to the correctness of the 4-way handshake that at least one bit differs in each message. Within the 4-way handshake, ~~message 1 can be recognized as the only one in which the (#1829)Key MIC Present bit is 0, meaning message 1 does not include the MIC, while message 2 to message 4 do. Message 3 differs from message 2 by not asserting the Ack bit and from message 4 by asserting the Ack Bit. Message 2 differs from message 4 by including the RSNE~~ the messages are distinguished by the Key MIC and Encrypted Key Data bits (both 0 in message 1 only), the Key Ack bit (1 in message 1 and message 3 only) and the presence of RSNEs and/or Multi-Band elements (in message 2 but not in message 4).

~~Request messages are distinct from 4-way handshake messages because the former assert(#1826) the Request bit and 4-way handshake messages do not. Group key handshake messages are distinct from 4-way handshake messages because they assert a different key type.~~

Group key handshake messages are distinct from 4-way handshake messages because they set the Key Type bit differently (0 in group key handshake messages only), and are distinguished from each other by the Key Ack bit (1 in message 1 only). Request messages are distinct from 4-way handshake and group key handshake messages because they set the Request bit differently (1 in request messages only).

At 2918.47 change “none require” to “none required”.

Change 12.7.9.3 as follows:

— AuthenticationRequest. The Supplicant sets this variable to true if its STA’s IEEE 802.11 management entity reports that an SSID is to be authenticated. ~~This might be on association or at other times.~~

Change 12.7.10.3 as follows:

— AuthenticationRequest. This variable is set to true by the STA’s IEEE 802.11 management entity in order to authenticate an association. ~~This can be set to true when the STA associates or at other times.~~

At 2950.32, restore Figure 12-56—FILS Shared Key authentication to what it was in D1.0.

At 2969.57 insert a para:

The same FT 4-way handshake is performed for PTK rekeying, except that the Secure bit is set in the first two messages (but see 12.7.2).

In Figure 13-15—R1KH state machine, including portions of the SME (part 1) move “PTK‑RekeyRequest = false” from the FT-INIT-R1\_SA state to the end of the FT-PTK-START state and in the latter change:

Send EAPOL-Key (0, 0, 1, 0, P, 0, 0, ANonce, 0, {})

to (note deletion of space before the opening paren):

Send EAPOL-Key(PTK-RekeyRequest ? 1 : 0, 0, 1, 0, P, 0, 0, ANonce, 0, {}) // M1

At 2997.46 append “ // M3” after “Send 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})” and delete the space before the opening paren.

In Figure 13-18—S1KH state machine, including portions of the SME (part 1):

* At the end of R1-START add “PTKRekeying = false”
* Add a new state FT-REKEY that sets “PTKRekeying = true” and then does UCT into FT-INIT-R1-SA
* Move the side arrow into FT-INIT-R1-SA to go into the new FT-REKEY
* In FT-PTK-START change “Send EAPOL-Key (0, 1, 0, 0, P, 0, SNonce, […])” to “Send EAPOL-Key(PTKRekeying ? 1 : 0, 1, 0, 0, P, 0, SNonce, […]) // M2” (note deletion of space before opening paren)

At 3002.36 append “ // M4” after “Send EAPOL-Key (1, 1, 0, 0, P, 0, 0, MIC-KCK)” and delete the space before the opening paren.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3631, 3596 in <this document>, which indicate that the Secure bit in M1 and M2 is not necessarily 0, and fix various issues with the security pseudocode/state machines.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3532  Mark RISON  12 | The row ordering in Table 12-7--Cipher suite key lengths seems haphazard | Sort alphabetically |

Discussion:

The reference is incorrect; it should be to Table 12-8:

**Table 12-8—Cipher suite key lengths**

|  |  |  |
| --- | --- | --- |
| **Cipher suite** | **Key length (octets)** | **TK\_bits (bits)** |
| TKIP | 32 | 256 |
| CCMP-128 | 16 | 128 |
| BIP-CMAC-128 | 16 | 128 |
| GCMP-128 | 16 | 128 |
| GCMP-256 | 32 | 256 |
| CCMP-256 | 32 | 256 |
| BIP-GMAC-128 | 16 | 128 |
| BIP-GMAC-256 | 32 | 256 |
| BIP-CMAC-256 | 32 | 256 |

It might be that this is historically the order in which these suites were added to the standard, but that’s not helpful to the reader.

Proposed changes:

Change Table 12-8 to have the following order:

**Table 12-8—Cipher suite key lengths**

|  |  |  |
| --- | --- | --- |
| **Cipher suite** | **Key length (octets)** | **TK\_bits (bits)** |
| BIP-CMAC-128 | 16 | 128 |
| BIP-CMAC-256 | 32 | 256 |
| BIP-GMAC-128 | 16 | 128 |
| BIP-GMAC-256 | 32 | 256 |
| CCMP-128 | 16 | 128 |
| CCMP-256 | 32 | 256 |
| GCMP-128 | 16 | 128 |
| GCMP-256 | 32 | 256 |
| TKIP | 32 | 256 |

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3532 in <this document>, which use alphanumeric order for Table 12-8 (Cipher suite key lengths).

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3326  Mark RISON  9.4.1.9  755.21 | "Status code" (name of field) should be "Status Code" | I can provide locations |

Discussion:

There are 75x “Status code” and 354 “Status Code”.

Proposed resolution:

REVISED

Change “Status code” to “Status Code” at 712.15, 728.33, 730.18, 1535.7, 1674.5, 3011.44/47, 3013.11/13.

At 1089.37 change “The Status Code field contains the final IEEE 802.11 Status code, as defined in Table 9-78 (Status codes) in 9.4.1.9 (Status Code field), received at the end of the applicable operation.” to “The Status Code field contains the status code, as defined in Table 9-78 (Status codes) in 9.4.1.9 (Status Code field), received at the end of the applicable operation.”

At 1559.46, 1561.60 change “The Status Code values” to “The Status Code field values”.

At 1626.54 change “the Status code field value is not set to 5” to “the BTM Status Code field is not 5”.

At 1626.56 change “if the Status code subfield contains 0” to “if the BTM Status Code field is 0”.

At 1658.30, 1660.26 change “indicated” to “defined”.

At 2396.26 change “FMS Status code” to “FMS status code”.

At 2627.40 change “Status Code” to “Status Code field”.

At 2627.43 change “a Comeback Delay and Query Response Length both set to 0” to “Comeback Delay and Query Response Length fields both set to 0”.

At 2629.45 change “Status Code” to “Status Code field”.

At 2629.47 change “a GAS Comeback Delay set to 0 and a Query Response Length set to 0” to “and GAS Comeback Delay and Query Response Length fields both set to 0”.

At 2630.1 change “Status Code” to “Status Code field”.

At 2630.3 change “a GAS Comeback Delay set to 0 and a Query Response Length set to 0” to “and GAS Comeback Delay and Query Response Length fields both set to 0”.

At 2630.23 change “a status code equal to GAS\_QUERY\_TIMEOUT” to “a Status Code field set to GAS\_QUERY\_TIMEOUT”.

At 2630.57 change “with status code equal to GAS\_RESPONSE\_NOT\_RECEIVED\_FROM \_SERVER” to “with a Status Code field set to GAS\_RESPONSE\_NOT\_RECEIVED\_FROM\_SERVER”.

At 2825.33 change “a Transaction Sequence Number of 1 and a Status Code of SUCCESS or SAE\_HASH\_TO\_ELEMENT” to “a Transaction Sequence Number field set to 1 and a Status Code field set to indicate SUCCESS or SAE\_HASH\_TO\_ELEMENT”.

At 2826.21 change “a Transaction Sequence Number of 2 and a Status Code of SUCCESS” to “a Transaction Sequence Number field set to 2 and a Status Code field set to indicate SUCCESS”.

Change “Status code” to “status code” at 2831.43/46 (and delete “set to”)/51, 2832.11/13/19/22/33/37/44.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 3291  Mark RISON  21.1  3387.4 | L\_DATARATE is missing in Table 21-1--TXVECTOR and RXVECTOR parameters (cf. CID 1057) | As it says in the comment |

Discussion:

CID 1057’s resolution was to add the following row to Table 27-1—TXVECTOR and RXVECTOR parameters for HE:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| L\_DATARATE | FORMAT is NON\_HT | Indicates the rate used to transmit the PSDU in megabits per second.  Allowed values depend on the value of the NON\_HT\_MODULATION  parameter as follows:  ERP-DSSS: 1 and 2  ERP-CCK: 5.5 and 11  ERP-OFDM, NON\_HT\_DUP\_OFDM:  6, 9, 12, 18, 24, 36, 48, and 54  OFDM: 6, 9, 12, 18, 24, 36, 48, and 54 | Y | Y |
| Otherwise | Not present | | |

Note there was already an L\_LENGTH row, though this just xreffed back to earlier PHYs for non-HE:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| L\_LENGTH | FORMAT is HE\_SU,  HE\_MU, or HE\_ER\_SU | Not present  NOTE—The LENGTH field of the L-SIG field for HE PPDU  is defined in Equation (27-11) using the TXTIME value  defined in 27.4.3 (TXTIME and PSDU\_LENGTH  calculation), which in turn depends on other parameters  including the TXVECTOR parameter APEP\_LENGTH. | N | N |
| FORMAT is HE\_TB | Indicates the value in the LENGTH field of the L-SIG field in  the range of 1 to 4095. The value is obtained from the  triggering frame to which the HE TB PPDU is a response. | Y | N |
| Otherwise | See corresponding entry in Table 19-1 (TXVECTOR and RXVECTOR  parameters) or Table 21-1 (TXVECTOR and RXVECTOR  parameters(#12)). | | |

Table 19-1—TXVECTOR and RXVECTOR parameters for HT already has L\_DATARATE (and L\_LENGTH), for NON\_HT, HT\_MF and HT\_GF formats.

Table 22-1—TXVECTOR and RXVECTOR parameters for TVHT already has L\_DATARATE (and L\_LENGTH), for NON\_HT and VHT formats.

But Table 21-1—TXVECTOR and RXVECTOR parameters has no L\_DATARATE (or L\_LENGTH).

Proposed changes:

In Table 21-1—TXVECTOR and RXVECTOR parameters, after the NON\_HT\_MODULATION row add the following:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| L\_DATARATE | FORMAT is NON\_HT | Indicates the rate used to transmit the PSDU in megabits per second. The allowed values are 6, 9, 12, 18, 24, 36, 48, and 54. | Y | Y |
| Otherwise | Not present | | |
| L\_LENGTH | FORMAT is NON\_HT | See corresponding entry in Table 19-1 (TXVECTOR and RXVECTOR parameters). | Y | Y |
| Otherwise | Not present | | |

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 3291 in <this document>, which add L\_DATARATE and L\_LENGTH parameters to the VHT TXVECTOR/RXVECTOR.

TBD:

3375/3514: “Its value is determined by device capabilities” (also post to reflector)

3544: "header field[s]" in Clause 20 and children clauses should be "Header field"

3659: Sometimes fields described as "(optional)" in their defining figure don't say ", if present," in their description

3667: Don't use hyphens for minuses

3668: Figures should not have xrefs unless they will automatically be updated when the xref number changes, otherwise there will be spec rot

3677: Some of the PICS selectors have unnecessary outside parens

3679: Things like "See 11.22.2.4 (Peer-to-peer link" should not be in DESCRIPTIONs unless there is a REFERENCE line to give the revision. Actually should be entirely in the REFERENCE line

3681: "vendor-specific element" (case-insensitively) should be "Vendor Specific element" and otherwise "vendor-specific" should be "vendor specific" (case-preservingly)

3700: The use of "or both" (~77x) implies that uses of "or" without and "or both" are exclusive, but this is not the case

3701: Should not have a comma before a modal verb (when not the end of a subclause or list)

3705: There are many editorial issues with the description of subelements (19/0856)

3723: "Key ID" should be "key ID" when not at the start of a sentence etc. and not followed by "field" etc.

3733: Delete "(bit <n>)" when it just duplicates info already in a figure or table

3735: "Classifier Parameter" is sometimes spuriously capitalised. Also "the Classifier.". Also "Classifier Type" and words after it inconsistent too

3343: CID 1851 changes to change "PTKSA" to "any PTKSA(s)" (because you could be doing EKIDs) should be done throughout

3433: In ranges expressed as x-y the hyphen should be a minus or en dash (in some places hyphens or em dashes are used

3543: "counter value" should just be "counter"

3545: "any STA" is the same as "a STA" and the latter would be more consistent so should be used

3586: "is an unsigned integer" is not needed in Clause 9 since covered by 9.2.2

3623: Identifiers should not have spaces in them

3625: "Vendor-Specific" should have a space, not a hyphen. Also case sometimes wrong ("Vendor-specific" or "vendor-specific" or "Vendor specific" or "vendor specific")

3627: The list of AANs in 9.4.1.1 Authentication Algorithm Number field should be a table

3645: CID 238 follow-up. I expect folding in new amendments will have added abbreviations that are not expanded on first use in each definition.

3706: "The structure of this/the blah field/element" is a bit odd

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID xxx  Mark RISON |  |  |

Discussion:

Proposed changes:

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID xxx in <this document>, which xxx

**References:**

802.11me/D2.0 except where otherwise specified