IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolutions for some comments on 11md/D3.0 (SB1) | | | | |
| Date: 2020-04-01 | | | | |
| 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 11md/D3.0. 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” view should be selected in Word.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4393  Mark RISON | It doesn't make sense to sometimes plonk "(no data)" after the frame name | Delete "(no data)" throughout except in Table 9-1--Valid type and subtype combinations |

Discussion:

In general, references to Data frames that contain no data are not qualified with “(no data)”. For example there are about 8 instances of “QoS Null (no data)” and over 100 instances of “QoS Null” without “(no data)”.

Proposed resolution:

REVISED

In D3.1:

At 782.10 change:

*QoS (+)Null* frame refers to all three QoS data subtypes with “no data”: the QoS Null (no data) frame, subtype 1100; the QoS CF-Poll (no data) frame, subtype 1110; and the QoS CF-Ack +CF-Poll frame, subtype 1111.

to:

*QoS (+)Null* frame refers to all three QoS data subtypes with an empty frame body: the QoS Null frame, subtype 1100; the QoS CF-Poll frame, subtype 1110; and the QoS CF-Ack +CF-Poll frame, subtype 1111.

Delete “ (no data)” at 785.60, 786.15/18/20, 790.48 (2x), 790.49, 799.43, 850.32 (3x), 850.33 (2x), 1860.24, 3605.32/35/39/47, 3613.23/27/30/41.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4432  Mark RISON  11 | "The Address 1 field of the TIM frame shall be set to the broadcast address." -- equivalent statements are needed for other Management frames that are always broadcast e.g. Beacon, FILS Discovery frames | As it says in the comment |

Discussion:

The following Management frames are already explicitly specified to be transmitted as broadcasts:

The Address 1 field of the **TIM** frame shall be set to the broadcast address.

An AP shall transmit **Measurement Pilots** *[sic]* to the broadcast address.

However, I cannot find anything to say that FILS Discovery frames or Timing Advertisement frames are broadcast, nor indeed that Beacon frames are broadcast! [Am I missing something? And are there other Management frames that are broadcast?]

Note: 11ax/D6.0 proposes adding the following sentence to 11.46.2.1:

The Address 1 field of the FILS Discovery frame shall be set to the broadcast address.

Note: DMG beacons don’t have an RA, just a BSSID, and S1G beacons don’t have an RA either, just a SA.

Note: various Management frames can be broadcast or unicast depending on the situation (e.g. Announce frames).

Proposed resolution:

REVISED

In D3.1:

At the end of the first para of 11.46.2.1 FILS Discovery frame transmission add “The Address 1 field of the FILS Discovery frame shall be set to the broadcast address.”.

At the end of the second para of 11.1.3.1 General (in 11.1.3 Maintaining synchronization) add “The Address 1 field of the Beacon or Timing Advertisement frame shall be set to the broadcast address.”.

At the end of the third para of 11.1.3.1 General (in 11.1.3 Maintaining synchronization) add “The Address 1 field of the Timing Advertisement frame shall be set to the broadcast address.”.

At the end of second para of 14.13.3.1 Beacon generation in MBSSs add “The Address 1 field of the Beacon frame shall be set to the broadcast address.”.

At 2302.47 and 2302.49 change “multiple Beacons, Measurement Pilots, or Probe Response frames” to “multiple Beacon, Measurement Pilot, or Probe Response frames”.

At 2318.56, 2320.52/53(2x)/60/61/63, 2321.1/3/5/13/19/43 change “Measurement Pilots” to “Measurement Pilot frames”. At 327.16 change “measurement pilots” to “Measurement Pilot frames”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4451  Mark RISON  10.23.2.7  1834.13 | "When an AP supports DL-MU-MIMO, frames from a higher or lower priority AC may be included  in a VHT or S1G MU PPDU with the TXVECTOR parameter(#2639) NUM\_USERS > 1 when  these frames do not increase the duration of the VHT or S1G MU PPDU beyond that required for the  transmissions of the frames of the primary AC(#2426)." -- why the not increase duration constraint, given that the previous bullet does allow extension? Maybe special-case for TXOP Limit 0, i.e. only in that case do not extend (since otherwise TXOP Limit is the limit, irrespective of content)? | Change to "When an AP supports DL-MU-MIMO, frames from a higher or lower priority AC may be included  in a VHT or S1G MU PPDU with the TXVECTOR parameter(#2639) NUM\_USERS > 1 when  the TXOP limit for the primary AC is nonzero." |

Discussion:

The current wording is:

**10.23.2.7 Sharing an EDCA TXOP**

(#1195)The AC associated with the EDCAF that gains an EDCA TXOP is referred to as the primary AC. Frames from ACs other than the primary AC shall not be included in the TXOP, with the following exceptions (TXOP sharing):

— Frames from a higher priority AC may be included when at least one frame from the primary AC has been transmitted and all frames from the primary AC have been transmitted.

— When an AP supports DL-MU-MIMO, frames from a higher or lower priority AC may be included in a VHT or S1G MU PPDU with the TXVECTOR parameter(#2639) NUM\_USERS > 1 when these frames do not increase the duration of the VHT or S1G MU PPDU beyond that required for the transmissions of the frames of the primary AC(#2426). Frames from the primary AC shall be transmitted first.

The first bullet allows higher-priority frames to be included in the TXOP, even if this causes the TXOP to be lengthened, as long as the primary AC frames have gone out (and implicitly as long as the TXOP limit is not violated).

However, the second bullet disallows (or can be read as disallowing) higher-priority frames from being included in the MU PPDU if that would lengthen the MU PPDU. It should be made clearer that this is not the case, i.e. just as for the first bullet, higher-priority frames can be included even if they lengthen the MU PPDU (and possibly also the TXOP), subject to the overall restrictions.

Also, the final sentence of the second bullet is not clear. There might not be frames for the primary AC for all users, and higher-priority stuff should go ahead of lower-priority stuff, since lower-priority stuff should only be allowed if it’s getting a completely free ride that would go unused otherwise.

Examples with the EDCAF that wins contention being VI, with a non-zero TXOP limit, and the TXOP not exceeding that limit:

* Sending some SU PPDUs with VI traffic (none left to tx) and then some SU PPDUs with VO traffic: OK
* Sending some SU PPDUs with VI traffic (none left to tx) and then some SU PPDUs with BE traffic: not OK
* Sending some MU PPDUs with one user having just VI traffic (no extraneous padding) and another user having just BE traffic (no VI/VO traffic to send): OK
* Sending some MU PPDUs with one user having just VI traffic but extraneous padding so that another user can have extra BE traffic: not OK
* Sending some MU PPDUs with one user having just VI traffic (none left to tx; no extraneous padding) and then that user having just VO traffic (no extraneous padding), and another using having just BE traffic throughout (no VI/VO traffic to send): OK
* Sending some MU PPDUs with one user having just VI traffic (none left to tx; no extraneous padding) and then that user having just VO traffic (none left to tx; no extraneous padding) and then that using having BE traffic, and another using having just BE traffic throughout (no VI/VO traffic to send): not OK
* Sending some MU PPDUs with one user having just VO traffic and then that user having VI traffic: not OK

There is no need to explicitly discuss the case where the TXOP limit is 0, since per 10.23.2.9 TXOP limits this means a single A-MPDU to each user, so the case is covered.

Proposed changes:

Change 10.3.2.7 as follows:

**10.23.2.7 Sharing an EDCA TXOP**

(#1195)The AC associated with the EDCAF that gains an EDCA TXOP is referred to as the primary AC. Frames from ACs other than the primary AC shall not be included in the TXOP, with the following exceptions (TXOP sharing):

— Frames from a higher priority AC may be included when at least one frame from the primary AC has been transmitted and all frames from the primary AC have been transmitted.

NOTE—The frames from a higher priority AC might be included in successive PPDUs in the TXOP and/or in one or more MU PPDUs.

— When an AP supports DL-MU-MIMO, frames from a ~~higher or~~ lower priority AC may be included in a VHT or S1G MU PPDU with the TXVECTOR parameter(#2639) NUM\_USERS > 1 when these frames do not increase the duration of the VHT or S1G MU PPDU beyond that required for the transmissions of the frames of the primary AC(#2426) and any frames from a higher priority AC. For a given user, any frames from the primary AC shall be transmitted first and then any frames from a higher priority AC immediately next.

The EDCAF remains bound by the TXOP limit for its AC (i.e. the primary AC), irrespective of the AC(s) of the frames transmitted during the TXOP.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 4451 in <this document>, which xx

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4433  Mark RISON  10.27.3.1  1902.28 | Table 10-22--Applicable HT protection mechanisms only goes up to 40 MHz non-HT dup. However, it also applies to VHT through 10.27.5 Protection rules for VHT STAs ("A VHT STA is subject to all of the rules for HT STAs that apply to its operating band, except that a PPDU  with the TXECTOR FORMAT parameter set to VHT may be substituted for a PPDU with the TXVECTOR  FORMAT parameter set to HT\_MF."). Therefore it should also cover the use of 80/80+80/160 non-HT dup for RTS" | In Table 10-22 change "40 MHz transmissions use non-HT duplicate frames defined in Clause 19 (High-throughput (HT) PHY  specification)" to "40 MHz, 80 MHz, 160 MHz and 80+80 MHz transmissions use non-HT duplicate frames defined in Clause 19 (High-throughput (HT) PHY  specification) and Clause 21" |

Discussion:

10.27.5 says:

**10.27.5 Protection rules for VHT STAs**

A VHT STA is subject to all of the rules for HT STAs that apply to its operating band, except that a PPDU with the (#4228)TXVECTOR FORMAT parameter set to VHT may be substituted for a PPDU with the TXVECTOR FORMAT parameter set to HT\_MF.

So this means Table 10-22 in 10.27.3 Protection mechanisms for transmissions of HT PPDUs applies to VHT STAs too. This says:

**Table 10-22—Applicable HT protection mechanisms(#67)**

|  |
| --- |
| **HT protection mechanism** |
| Control frames such as RTS/CTS or CTS-to-self prior to the HT transmissions:  — 20 MHz transmissions use the rates defined in Clause 17 (Orthogonal frequency division multiplexing (OFDM) PHY specification) or Clause 18 (Extended Rate PHY (ERP) specification)  — 40 MHz transmissions use non-HT duplicate frames defined in Clause 19 (High-throughput (HT) PHY specification) |
| As the first PPDU in the TXOP, send one of:  — a non-HT PPDU containing a frame that requires an immediate response  — (#4434)an HT-mixed format PPDU containing a frame that requires an immediate response in a non-HT PPDU  PPDUs after the first PPDU exchange may be HT-greenfield format PPDUs and/or be separated by RIFS. |

As the comment says, protection of 80+ MHz transmissions should also be covered for VHT.

TGm prefers, however, not to make changes to Table 10-22 since it sees that table as being for HT STAs.

Proposed resolution:

REVISED

At the end of the sentence in 10.27.5 Protection rules for VHT STAs add “ and that the applicable HT protection mechanisms are extended to include 80, 160 and 80+80 MHz transmissions using non-HT duplicate frames defined in Clause 21” (1906.11 in D3.0).

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4582  Mark RISON | The definition of dot11EDCATableMSDULifetime (and QAP version) needs to allow for A-MSDUs and MMPDUs, since those are/can be sent under a particular AC. Also similarly change 1763.63 in 10.3.4.4 and dot11MaxTransmitMSDU Lifetime in C.3 | As it says in the comment |

Discussion:

dot11EDCATableMSDULifetime’s definition says that it “specifies the maximum duration an MSDU, for a given AC, would be retained by the MAC before it is discarded.”

So which attribute specifies the lifetime for MMPDUs? It must be this one too, since there’s no other one for this.

However, we don’t need to explicitly allow for A-MSDUs, since MSDUs are a component of A-MSDUs (see also e.g. D3.1/1843.43).

1763.63 is:

Retries for failed transmission attempts shall continue until the SRC for the MPDU with the Type subfield

equal to Data or Management is equal to dot11ShortRetryLimit or until the LRC for the MPDU with the Type subfield equal to Data or Management is equal to dot11LongRetryLimit. When either of these limits is reached, retry attempts shall cease, and the MPDU with the Type subfield Data (and any MSDU of which it is a part) or Management shall be discarded. A DMG STA, in addition to using random access within a CBAP, may transmit retries in available scheduled SPs.

so it’s not entirely clear what the commenter is on about here (is this a similar point to CID 4168?).

The range of the EDCA MSDU lifetimes seems odd, too. It allows 0, i.e. the MSDUs are stillborn, and it only allows up to 500 TUs. It should be the same as the range for DCF MSDU lifetimes, which goes from 1 TU to 232‑1 TUs (for the *really* determined).

Proposed changes:

Change D3.1 as follows:

**10.4 MSDU and MMPDU fragmentation**

The source STA shall maintain a transmit MSDU/MMPDU timer for each MSDU/MMPDU being transmitted. The attribute dot11MaxTransmitMSDULifetime specifies the maximum amount of time allowed to transmit an MSDU/MMPDU. The timer starts on the initial attempt to transmit the (#1452)MSDU/MMPDU, or first fragment of the MSDU/MMPDU (#1452)if the MSDU/MMPDU is fragmented. If the timer exceeds dot11MaxTransmitMSDULifetime, then (#1452)any remaining fragments are discarded by the source STA and no attempt is made to complete transmission of the MSDU/MMPDU.

**10.7 MSDU transmission restrictions**

(#66)A STA should select a value of dot11MaxTransmitMSDULifetime that is sufficiently large that the STA does not discard MSDUs or MMPDUs~~A-MSDUs~~ due to the transmit MSDU/MMPDU timer being exceeded, ~~excessive Transmit MSDU timeouts~~ under normal operating conditions.

**10.23.2.12 Retransmit procedures**

**10.23.2.12.1 General**

(#2432)A QoS STA shall maintain a transmit MSDU/MMPDU timer for each MSDU passed to the MAC and for each MMPDU. dot11EDCATableMSDULifetime specifies the maximum amount of time allowed to transmit an MSDU/MMPDU for a given AC. The transmit MSDU/MMPDU timer shall be started when the MSDU/MMPDU is passed to the MAC.

(#2432)When A-MSDU aggregation is used, the HT STA maintains a single timer for the whole A-MSDU. The timer is restarted each time an MSDU is added to the A-MSDU. The result of this procedure is that no MSDU in the A-MSDU is discarded before a period of dot11EDCATableMSDULifetime has elapsed.

(#2664)(#1505)Retries for failed transmission attempts shall continue until one or more of the following conditions occur:

— (#2664)The frame retry count for the MSDU, A-MSDU, or MMPDU is equal to dot11ShortRetryLimit.

— (#2664)The drop-eligible frame retry count for the MSDU, A-MSDU, or MMPDU is equal to dot11ShortDEIRetryLimit.

— (#2664)The unsolicited frame retry count for the A-MSDU is equal to dot11UnsolicitedRetryLimit.

— (#2432)The transmit MSDU/MMPDU timer for the MSDU/MMPDU or any undelivered fragments of that MSDU/MMPDU exceeds dot11EDCATableMSDULifetime.

**C.3 MIB detail**

dot11MaxTransmitMSDULifetime OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

UNITS "TUs"

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 MaxTransmitMSDULifetime is~~This attribute specifies the elapsed time, after the initial transmission of an MSDU/MMPDU (or the first fragment thereof), after which further attempts to transmit the MSDU/MMPDU are terminated."

DEFVAL { 512 }

::= { dot11OperationEntry 6 }

dot11EDCATableMSDULifetime OBJECT-TYPE

SYNTAX Unsigned32 (~~0..500~~1..4294967295)

UNITS "TUs"

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by the MAC upon receiving an EDCA Parameter Set in a Beacon frame.

Changes take effect as soon as practical in the implementation.

This attribute specifies the maximum duration an MSDU/MMPDU, for a given AC, would be retained by the MAC at the non-AP STA before it is discarded."

DEFVAL { 500 }

::= { dot11EDCAEntry 6 }

dot11QAPEDCATableMSDULifetime OBJECT-TYPE

SYNTAX Unsigned32 (~~0..500~~1..4294967295)

UNITS "TUs"

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.

This attribute specifies the maximum duration an MSDU/MMPDU, for a given AC, would be retained by the MAC at the AP before it is discarded."

DEFVAL { 500 }

::= { dot11QAPEDCAEntry 6 }

Proposed resolution:

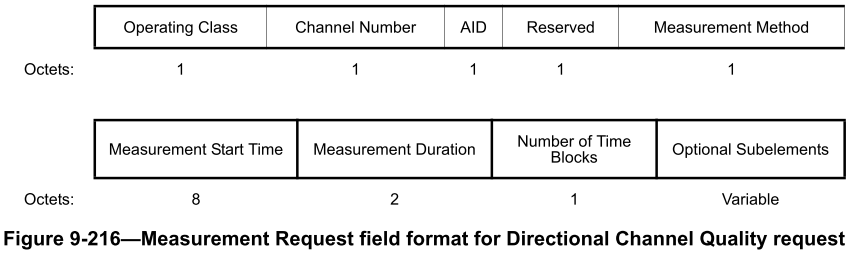
REVISED

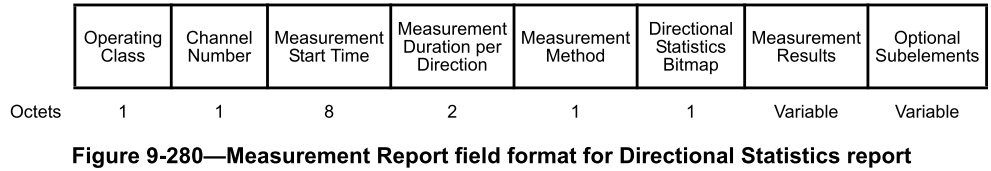
Make the changes shown under “Proposed changes” for CID 4582 in <this document>, which address the issue raised by the commenter (except for the one at 1763.63 -- see CID 4168).

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4284  Mark RISON  9.4.2.20.18  1043.59 | Should Figure 9-219--Measurement Request field format for Directional Statistics request not allow for optional subelements, like the corresponding report, and like most requests? Ditto Directional Measurement request | As it says in the comment |

Discussion:

Most requests/reports allow for optional subelements, if only to allow for vendor-specific subelements. E.g.:





However, for some reason, Directional Statistics requests and Directional Measurement requests don’t allow for optional subelements. There is no reason for this limitation.

Proposed changes:

In Figure 9-218—Measurement Request field format for Directional Measurement request and Figure 9-219—Measurement Request field format for Directional Statistics request add a field at the end “Optional Subelements” with “Variable” as the number of octets.

At the end of 9.4.2.20.17 Directional Measurement request insert:

The Optional Subelements field contains zero or more subelements. The subelement format and ordering of subelements are defined in 9.4.3.

The Subelement ID field values for the defined subelements are shown in Table 9-xx.

Immediately after, insert a Table 9-xx with caption “Optional subelement IDs for Directional Measurement request” and contents the same as Table 9-177—Optional subelement IDs for Measurement Pilot Transmission.

Immediately after, insert:

The Vendor Specific subelements have the same format as their corresponding elements (see 9.4.2.25). Zero or more Vendor Specific subelements are included in the list of optional subelements.

At the end of 9.4.2.20.18 Directional Statistics request insert:

The Optional Subelements field contains zero or more subelements. The subelement format and ordering of subelements are defined in 9.4.3.

The Subelement ID field values for the defined subelements are shown in Table 9-yy.

Immediately after, insert a Table 9-yy with caption “Optional subelement IDs for Directional Statistics request” and contents the same as Table 9-177—Optional subelement IDs for Measurement Pilot Transmission.

Immediately after, insert:

The Vendor Specific subelements have the same format as their corresponding elements (see 9.4.2.25). Zero or more Vendor Specific subelements are included in the list of optional subelements.

Remember to hyperlinkify all the xrefs!

Proposed resolution:

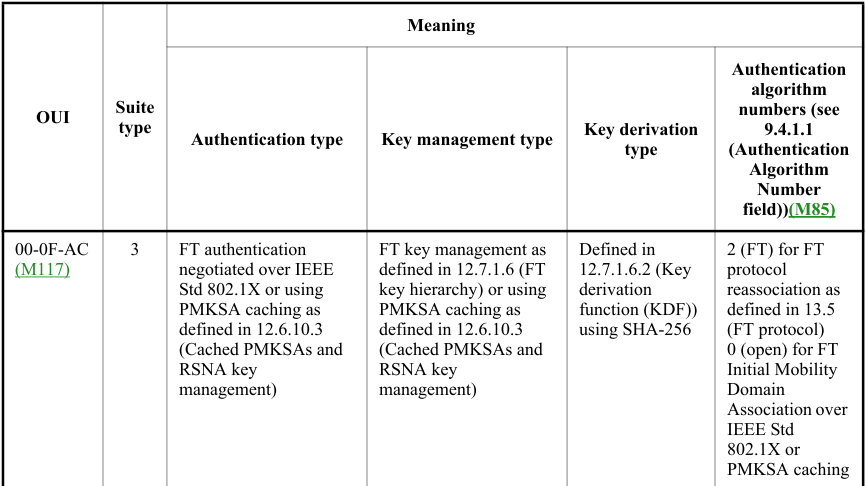
REVISED

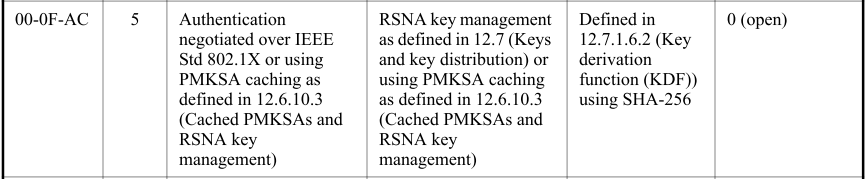
Make the changes shown under “Proposed changes” for CID 4284 in <this document>, which allow for optional subelements in the requests identified by the commenter.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4204  Mark RISON  9.4.2.24.3  1101.29 | Table 9-151--AKM suite selectors has overlapping conditions. For example, 00-0F-AC:3 and 00-0F-AC:5 have the same key derivation, can both use 0 for the auth alg num, have subset key management type (since 12.7.1.6 is a subclause of 12.7) and have subset authentication (since FT authentication  negotiated over IEEE  Std 802.1X is a type of Authentication  negotiated over IEEE  Std 802.1X). Similarly :8 and :9, etc. | Add a column to the table with heading something like "Can be used with PMKSA caching" and then state that this means that the AKM can also be used for the use of a cached PMKSA for a previous AKM of that type, and cross-reference from there to 12.6.10.3 Cached PMKSAs and RSNA key management |
| CID 4205  Mark RISON  9.4.2.24.3  1101.29 | Table 9-151--AKM suite selectors has overlapping conditions. For example, 00-0F-AC:3 and 00-0F-AC:5 have the same key derivation, can both use 0 for the auth alg num, have subset key management type (since 12.7.1.6 is a subclause of 12.7) and have subset authentication (since FT authentication  negotiated over IEEE  Std 802.1X is a type of Authentication  negotiated over IEEE  Std 802.1X). Similarly :8 and :9, etc. | Make sure each suite selector has no overlap with other suite selectors |

Discussion:

Here are the conditions for 00-0F-AC:3 and :5:





So, when the auth alg num is 0, what distinguishes :3 and :5? How does the receiver know which is intended?

* “FT authentication negotiated over IEEE Std 802.1X or using PMKSA caching as defined in 12.6.10.3”

is a subset of

“Authentication negotiated over IEEE Std 802.1X or using PMKSA caching as defined in 12.6.10.3”

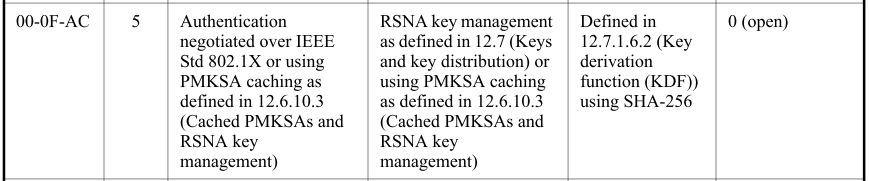
* “FT key management as defined in 12.7.1.6 or using PMKSA caching as defined in 12.6.10.3”

is a subset of

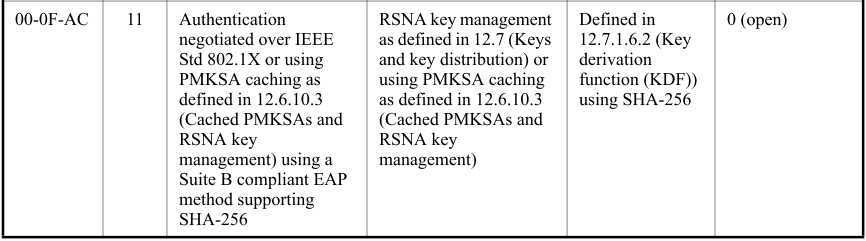
“RSNA key management as defined in 12.7 or using PMKSA caching as defined in 12.6.10.3”

* The key derivation type is the same (“Defined in 12.7.1.6.2 using SHA-256”)

Similarly, the only difference between



and



is that :11 guarantees that it's Suite B compliant (extra text at end of third cell).

Proposed changes:

Proposed resolution:

REVISED

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

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4499  Mark RISON | If the term "slave" is no longer acceptable (CID 2020), is the term "master" still acceptable (other than "master key" contexts)? There are only a few such instances | As it says in the comment |

Discussion:

If “slave” is objectionable, then so is “master”, in those contexts that imply the existence of a “slave”.

“master white space device” and “Master STA TVWS operation” are used in regulations, however, so cannot be changed.

Proposed resolution:

REVISED

In 9.5.6 Beamformed Link Maintenance field inc. Figure 9-852—Beamformed Link Maintenance field format and Table 9-344—The Beamformed Link Maintenance negotiation, change “isMaster” to “isController” (5x). In 9.5.6 Beamformed Link Maintenance field change “master of the data transfer” to “controller of the data transfer”.

In 11.1.2.1 TSF for an infrastructure BSS or a PBSS change “timing master” to “timing source”.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4715  Mark RISON  6.3.68  576.10 | MLME-FINETIMINGMSMTRQ primitives are not used anywhere | In Figure 6-17---Fine timing measurement primitives and timestamps capture add at the top a MLME-FINETIMINGMSMT.request from STA B's SME to STA B's MLME, a Fine Timing Measurement Request frame from STA B's MLME to STA A's MLME and a MLME-FINETIMINGMSMT.request from STA A's MLME to STA A's SME |
| CID 4716  Mark RISON  6.3.68  576.10 | MLME-FINETIMINGMSMTRQ primitives are not used anywhere | Delete 6.3.68 |

Discussion:

6.3.56 has information on the MLME-FINETIMINGMSMT primitives, including a figure showing how the primitives map to frames over the air. However, 6.3.68 has no similar information on the MLME-FINETIMINGMSMTRQ primitives.

Mark HAMILTON further points out:

Perhaps, if we are going to show the “RQ” primitives in the figure(s) (that is, Figure 6-17, and maybe as suggested in Figure 6-16 also), then we should merge those primitives into the subclause also.  That is, merge subclause 6.3.68 into 6.3.56 (and maybe 6.3.67 into 6.3.55)?

Those later clauses have the primitive definitions for MLME-[FINE]TIMINGMSMTRQ, .request and .indication.  In the current subclause with Figure 6-17, there are only the primitives for the FTM frames themselves.  But, the clause states this, w.r.t. Figure 6-17:

The following set of primitives supports exchange of FTM information from one SME to another.

(M138)The diagram in Figure 6-17 (Fine timing measurement primitives and timestamps capture) shows

various points in time that are of interest to the FTM procedure.

Given the combination of both sets of actions in the Figure, I think having both sets of primitives in the subclause makes sense.  Right now, the subclauses are very far apart, and there is no cross-reference hint to go look at the other one, to understand the whole picture that Figure 6-17 is trying to convey.

Proposed changes:

Change the first para of 6.3.56.1 General as follows:

The following set of primitives supports triggering an FTM(#1022) procedure or stopping an ongoing FTM procedure, and exchange of FTM information from one SME to another. (M138)~~The diagram in~~ Figure 6-17 (Fine timing measurement primitives and timestamps capture) shows the use of these primitives and various points in time that are of interest to the FTM procedure.

Change Figure 6-17 as follows, adding the material in red:



Insert 6.3.68.2 MLME-FINETIMINGMSMTRQ.request and 6.3.68.3 MLME-FINETIMINGMSMTRQ.indication after 6.3.56.1, renumbering them to 6.3.56.2 and 6.3.56.3 respectively, and renumbering the current 6.3.56.2 and 6.3.56.3 to 6.3.56.4 and 6.3.56.5 respectively. Delete 6.3.68 Fine timing measurement request and 6.3.68.1 General.

Change the first para of 6.3.55.1 General as follows:

The following set of primitives supports triggering a Timing Measurement procedure or stopping an

ongoing Timing Measurement procedure, and exchange of timing measurement information from one SME to another. (M138)The diagram in Figure 6-16 (Timing measurement primitives and timestamps

capture(#1563)) shows various points in time that are of interest to the timing measurement procedure.

Change Figure 6-16 by adding the material shown in red above, but with “FINE” and “Fine ” deleted.

Insert 6.3.67.2 MLME-TIMINGMSMTRQ.request and 6.3.67.3 MLME-TIMINGMSMTRQ.indication after 6.3.55.1, renumbering them to 6.3.55.2 and 6.3.55.3 respectively, and renumbering the current 6.3.55.2 and 6.3.55.3 to 6.3.55.4 and 6.3.55.5 respectively. Delete 6.3.67 Timing measurement request and 6.3.67.1 General.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CIDs 4715 and 4716 in <this document>, which clarify the use of MLME-FINETIMINGMSMTRQ primitives and also MLME-TIMINGMSMTRQ primitives.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4591  Mark RISON | The way UTF-8 strings are referred to is inconsistent. We have a definition of UTF-8 string (in 9.2.2) so just use that | In 9.4.2.2 change “the SSID is interpreted using UTF-8 encoding” to “the SSID is a UTF-8 string”. In 9.4.2.21.14 change “The Public Identifier URI/FQDN field contains a URI encoded using UTF-8 and formatted in accordance  with IETF RFC 3986” to “The Public Identifier URI/FQDN field contains a URI as a UTF-8 string, formatted in accordance  with IETF RFC 3986,”. In 9.4.2.26 change “The SSID in this BSS is interpreted using UTF-8 encoding” to “The SSID in this BSS is a UTF-8 string”. At 1217.8 change “an UTF” to “a UTF”. In 9.4.5.4 and 9.4.5.5 change “UTF-8 encoded field” to “UTF-8 string”. In 9.4.5.10 change “a UTF-8 encoded character string” to “a UTF-8 string” and “in UTF-8 format” to “in UTF-8”. In 9.4.5.17 change “field encoded using UTF-8 and “ to “UTF-8 string,”. In 9.4.5.21 change “UTF-8 formatted field “ to “UTF-8 string “. In 9.4.5.22 change “a UTF-8 formatted string” to “a UTF-8 string” |

Discussion:

The definition in 9.2.2 (which covers the conventions for structures defined in the MAC sublayer) defines a UTF‑8 string as follows:

A[…] UTF-8 string is a sequence of […] UTF-8 encoded code points […] without a terminating null.

[For those who are not character representation geeks: a code point is a number that represents (in general) a character in a character set, and an encoding is a mechanism for representing a code point as one or more numbers (see e.g. <https://unicode.org/glossary/> ). For example, the (non-Greek) micro character µ is Unicode code point U+00B5 and is encoded in UTF-8 (Unicode/UCS transformation format 8) as the octet sequence 0xC2 0xB5. For characters in ASCII, the UTF-8 encoding is the same as the ASCII encoding.]

An important point of the above definition is that it states that for the purposes of the MAC sublayer a UTF-8 string does not have a terminator. That is, the UTF-8 string “Password” is represented as “Password” (8 octets), not “Password\0” (9 octets, where \0 is the null character) -- this is obviously important where the string is being tested for equality or hashed.

Proposed changes:

Move the following sentence from 9.2.2 Conventions to the end of 1.4 Word usage:

An ASCII or UTF-8 string is a sequence of ASCII or UTF-8 encoded code points, respectively, without a

terminating null.

Change 9.4.2.2 SSID element as follows:

When the UTF-8 SSID subfield of the Extended Capabilities element is equal to 1 in the frame that includes the SSID element, or the Extended Capabilities of the source of the SSID information is known to include the UTF-8 SSID capability based on a previously received Extended Capabilities element, the SSID is ~~interpreted using UTF-8 encoding~~a UTF-8 string (see 1.4). Otherwise, the character encoding of the octets in this SSID element is unspecified.

Change 9.4.2.21.14 Location Identifier report as follows:

The Public Identifier URI/FQDN field contains a URI ~~encoded using UTF-8 and~~as a UTF-8 string, formatted in accordance with IETF RFC 3986, that points to a location object or an FQDN that identifies a location server.

Change Table 9-153—Extended Capabilities field as follows:

The SSID in this BSS is ~~interpreted using UTF-8 encoding~~a UTF-8 string

Change 9.4.2.68.5 Diagnostic subelement descriptions as follows:

The Certificate ID field contains a~~n~~ UTF-8 string indicating an identifier assigned to the STA in a manner

outside the scope of the standard. The Certificate ID typically takes the form of “WFA3991” and might be

used by a receiving STA to look up the certificate assigned to that ID.

Change 9.4.5.4 Venue Name ANQP-element as follows:

The Venue Name field is a variable length(#183) UTF-8 ~~encoded field~~string containing the venue’s name.

Change 9.4.5.5 Emergency Call Number ANQP-element as follows:

The Emergency Call Number field is a variable length(#183) UTF-8 ~~encoded field~~string containing information, used to reach emergency services, from the network (e.g., dialed digits, emergency service URN label [B40]).

Change 9.4.5.10 NAI Realm ANQP-element as follows:

The NAI Realm Encoding Type subfield(M101) is a 1-bit subfield. It is set to 0 to indicate that the NAI Realm in the NAI Realm subfield is formatted in accordance with IETF RFC 4282. It is set to 1 to indicate it is a UTF-8 ~~encoded character~~ string that is not formatted in accordance with IETF RFC 4282.

If there is more than one NAI Realm in this subfield, the NAI Realms are delimited by a semicolon character (i.e., “;”, which is encoded in UTF-8 ~~format~~ as 0x3B).

Change 9.4.5.17 Emergency NAI ANQP-element as follows:

The Emergency NAI Information field is a variable length(#183) ~~field encoded using~~ UTF-8 string, ~~and~~ formatted in accordance with IETF RFC 4282.

Change 9.4.5.21 Advice of Charge ANQP-element as follows:

The Plan Information field is a variable length UTF-8 ~~formatted field~~string that carries an XML description of an Advice of Charge plan.

Change 9.4.5.22 Local Content ANQP-element as follows:

(#2203)The Label field is a variable length(#183) field containing a text description of the URL. It provides the type and potential usage of the URL. This is a UTF-8 ~~formatted~~ string.

Proposed changes:

Move the following sentence from 9.2.2 Conventions to the end of 1.4 Word usage:

An ASCII or UTF-8 string is a sequence of ASCII or UTF-8 encoded code points, respectively, without a

terminating null.

and replace it in 9.2.2 with the following sentence (remember to hyperlinkify the xref!):

ASCII and UTF-8 strings are defined in 1.4.

Change 12.1 Conventions as follows (remember to hyperlinkify the xref!):

ASCII and UTF-8 strings are defined in ~~9.2.2 (Conventions)~~1.4.

Change 9.4.2.2 SSID element as follows:

When the UTF-8 SSID subfield of the Extended Capabilities element is equal to 1 in the frame that includes the SSID element, or the Extended Capabilities of the source of the SSID information is known to include the UTF-8 SSID capability based on a previously received Extended Capabilities element, the SSID is ~~interpreted using UTF-8 encoding~~a sequence of UTF-8 encoded code points. Otherwise, the character encoding of the octets in this SSID element is unspecified.

NOTE—If the SSID is a sequence of UTF-8 encoded code points, a terminating null might or might not be present.

Change 9.4.2.21.14 Location Identifier report as follows:

The Public Identifier URI/FQDN field contains a URI ~~encoded using UTF-8 and~~as a UTF-8 string, formatted in accordance with IETF RFC 3986, that points to a location object or an FQDN that identifies a location server.

Change Table 9-153—Extended Capabilities field as follows:

The SSID in this BSS is ~~interpreted using UTF-8 encoding~~a sequence of UTF-8 encoded code points

Change 9.4.2.68.5 Diagnostic subelement descriptions as follows:

The Certificate ID field contains a~~n~~ UTF-8 string indicating an identifier assigned to the STA in a manner

outside the scope of the standard. The Certificate ID typically takes the form of “WFA3991” and might be

used by a receiving STA to look up the certificate assigned to that ID.

Change 9.4.5.4 Venue Name ANQP-element as follows:

The Venue Name field is a ~~variable length(#183)~~ UTF-8 ~~encoded field~~string containing the venue’s name.

Change 9.4.5.5 Emergency Call Number ANQP-element as follows:

The Emergency Call Number field is a ~~variable length(#183)~~ UTF-8 ~~encoded field~~string containing information, used to reach emergency services, from the network (e.g., dialed digits, emergency service URN label [B40]).

Change 9.4.5.10 NAI Realm ANQP-element as follows:

The NAI Realm Encoding Type subfield(M101) is a 1-bit subfield. It is set to 0 to indicate that the NAI Realm in the NAI Realm subfield is formatted in accordance with IETF RFC 4282. It is set to 1 to indicate it is a UTF-8 ~~encoded character~~ string that is not formatted in accordance with IETF RFC 4282.

If there is more than one NAI Realm in this subfield, the NAI Realms are delimited by a semicolon character (i.e., “;”, which is encoded in UTF-8 ~~format~~ as 0x3B).

Change 9.4.5.17 Emergency NAI ANQP-element as follows:

The Emergency NAI Information field is a ~~variable length(#183)~~ ~~field encoded using~~ UTF-8 string ~~and~~ formatted in accordance with IETF RFC 4282.

Change 9.4.5.21 Advice of Charge ANQP-element as follows:

The Plan Information field is a ~~variable length~~ UTF-8 ~~formatted field~~string that carries an XML description of an Advice of Charge plan.

Change 9.4.5.22 Local Content ANQP-element as follows:

(#2203)The Label field is a ~~variable length(#183)~~ ~~field~~UTF-8 string containing a text description of the URL. It provides the type and potential usage of the URL. ~~This is a UTF-8 formatted string.~~

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 4591 in <this document>, which canonicalise references to UTF-8 strings or encoded code point sequences.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4679  Mark RISON  9.3.3.1  854.23 | "a) The Address 1 field of the Management frame is the RA (=DA) and is determined as the destination  of the frame.  b) The Address 2 field of the Management frame is the TA (=SA) and is determined as the address of  the STA transmitting the frame(#2013)." arguably duplicates 9.2.4.3.1 | As it says in the comment |

Discussion:

The description in 9.2.4.3.1 is:

Certain address field usage is specified by the relative position of the address field (1–4) within the MAC header, independent of the type of address present in that field. For example, receiver address matching is always performed on the contents of the Address 1 field in received frames, and the receiver address of CTS and Ack frames is always obtained from the Address 2 field in the corresponding RTS frame, or from the frame being acknowledged.

However, the examples duplicate more specific information, e.g. in 9.3.3.1 for Management frames:

A STA uses the contents of the Address 1 field to perform the address matching for receive decisions.

The address fields for all Management frames except Multihop Action frames are as follows:

a) The Address 1 field of the Management frame is the RA (=DA) and is determined as the destination of the frame.

b) The Address 2 field of the Management frame is the TA (=SA) and is determined as the address of the STA transmitting the frame(#2013).

and in 9.3.2.1.2 for Data frames:

The content of the address fields of Data frames are dependent upon the values of the To DS and From DS subfields in the Frame Control field and whether the Frame Body field contains either an MSDU (or fragment thereof) or an entire A-MSDU, as determined by the A-MSDU Present subfield of the QoS Control field (see 9.2.4.5.9 (A-MSDU Present subfield)). The content of the address fields transmitted by nonmesh STAs is defined in Table 9-32 (Address field contents). The content of the address fields transmitted by mesh STAs is defined in 9.3.5 (Frame addressing in an MBSS), and the content of the fields transmitted by GLK STAs is defined in 10.65 (Addressing of GLK Data frame(M101) transmission(11ak)).(11ak) Where the content of a field is shown as not applicable (N/A), the field is omitted. Note that Address 1 always holds the receiver address of the intended receiver (or, in the case of group addressed frames, receivers), and that Address 2 always holds the address of the STA that is transmitting the frame.

and in 9.3.1.x for Control frames, e.g. in 9.3.1.2 for RTS:

(MDR2)The RA field of the RTS frame is the address of the STA, on the WM, that is the intended immediate recipient of the pending individually addressed Data, Management, or Control frame.

(MDR2)The TA field is the address of the STA transmitting the RTS frame or the bandwidth signaling TA of the STA transmitting the RTS frame.

Also, it is not the case that “the receiver address of […] Ack frames is always obtained from the Address 2 field in the corresponding […] frame being acknowledged”, because per 9.3.1.4:

The RA field of the Ack frame is the nonbandwidth signaling TA from the Address 2 field of the immediately previous individually addressed Data, Management, BlockAckReq, BlockAck, or PS-Poll frames.

so the RA of the Ack might not be the same as the A2 (TA) of the preceding frame. Though I suppose “obtained from” could be argued to be different from “copied from”.

It is possible to argue that 9.2.4.3.1’s ad libbing is helpful to set out general principles, however.

Proposed changes:

Proposed resolution:

REVISED

Alternative 1:

Delete “For example, receiver address matching is always performed on the contents of the Address 1 field in received frames, and the receiver address of CTS and Ack frames is always obtained from the Address 2 field in the corresponding RTS frame, or from the frame being acknowledged.” in 9.2.4.3.1.

Alternative 2:

After “For example, receiver address matching is always performed on the contents of the Address 1 field in received frames, and the receiver address of CTS and Ack frames is always obtained from the Address 2 field in the corresponding RTS frame, or from the frame being acknowledged.” in 9.2.4.3.1 add:

NOTE—The receiver address of Ack frames does not equal the Address 2 field of the frame being acknowledged, if that field was a bandwidth signalling TA.

Alternative 3:

Change “For example, receiver address matching is always performed on the contents of the Address 1 field in received frames, and the receiver address of CTS and Ack frames is always obtained from the Address 2 field in the corresponding RTS frame, or from the frame being acknowledged.” in 9.2.4.3.1 to “Specifically, the Address 1 field in received frames always identifies the receiver(s) of the frame, and the Address 2 field in received frames, where present, always identifies the transmitter of the frame.”

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

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4689  Mark RISON  9.4.3 | "The optional  subelements are ordered by nondecreasing subelement ID." (2x) -- they're ordered per 9.4.3 ("Subelements within an element are ordered by nondecreasing Subelement ID.") | Change to refer to 9.4.3 as for most optional subelement lists |

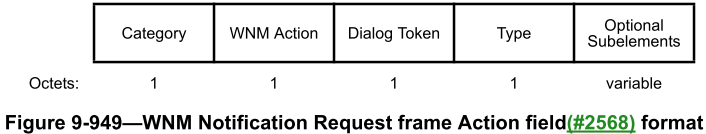
Discussion:

It has been pointed out that

In 9.4.3, subelements are within an element. In 9.6.7.37 and 9.6.7.38, subelements are within a field. Therefore, cannot change to refer to 9.4.3 in 9.6.7.37 and 9.6.7.38.

This is valid of itself, although pedantically in lots of places (if not most/all) subelements are within a field, though sometimes this is itself within an element.

Conversely, Figure 9-865—Measurement Pilot frame Action field format has an Optional Subelements field, is not in an element, and yet says "The Optional Subelements field contains zero or more subelements. The subelement format and ordering of subelements are defined in 9.4.3 (Subelements)." Ditto Figure 9-949—WNM Notification Request frame Action field(#2568) format:



Proposed resolution:

REVISED

In 9.4.3 change “Subelements within an element are ordered by nondecreasing Subelement ID.” (1465.1 in D3.0) to “Subelements within an element and subelements in a field outside of an element are in each case ordered by nondecreasing Subelement ID.”

In 9.6.7.37 and in 9.6.7.38 change “The Optional Subelements field contains zero or more subelements, each consisting of a 1-octet Subelement ID field, a 1-octet Length field, and a variable-length Data field, as defined in 9.4.3 (Subelements). The optional subelements are ordered by nondecreasing subelement ID.” (1566.1 and 1567.36 in D3.0) to “The Optional Subelements field contains zero or more subelements, The subelement format and ordering of subelements are defined in 9.4.3 (Subelements).”

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4809  Mark HAMILTON  9.6.12.3  1589.40 | Change 6 occurrences of "in response to a received", to be simpler and match the majority language in the draft. | Delete "received" (and change "a" to "an" as appropriate) at P1589L40, P1590L53, P1592L56, P2180L1, P2482L30, and P2482L39. |

Discussion:

This was ACCEPTED in motion 167. However, there are ~20 “in response to the rec\* of \*” which by the same token should be just “in response to \*”.

Proposed additional changes:

In D3.2, change “in response to the reception of”/“in response to the receipt of” to “in response to” at 637.33, 971.18/21/22, 1181.14, 1600.24, 1654.28, 1656.30, 2033.55, 2332.48, 2467.55, 2478.38, 3042.28, 3539.50, 4016.40, 4620.37, 4621.55. At 4611.43 change “in response to the receipt of MLME-” to “in response to an MLME-”. [This is all instances of “in response to the rec” except the one on page 1908.]

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4767  Mark HAMILTON  5.2.3.2  312.7 | Now that 802.11 data frames can carry a DEI indicator (in the HT control field), the 802.11 SAP (MA-UNITDATA primitives) should support a drop\_eligible parameter, to match 802.1AC's general assumptions. (802.1AC's Annex B.1.5 will also need to be updated to match this, by 802.1 WG.) | At P312L7, insert a new parameter “drop eligible”, after “priority”.  At P312L35 insert a new paragraph, “The drop eligible parameter provides guidance on whether this request can be discarded in preference to other requests when there are insufficient resources in a STA. If drop\_eligible is TRUE, the request can be discarded in preference to other requests in which drop\_eligible is FALSE. The default drop\_eligible value is FALSE.”  At P314L43, insert a new parameter “drop eligible” after “priority”.  At P315L6 insert a new paragraph, “The drop eligible parameter indicates if the received MSDU was designated as DEI in the received protocol headers. If no protocol header designation was received, the parameter value is FALSE.”  At P164L25, insert “drop eligible, “ after “priority, “.  At P808L27, append “, per the value of the drop eligibile parameter received in the MA-UNITDATA.request, if any.” to the end of the sentence. (... if there are insufficient resources at the recieving STA, per the value of the drop eligible ...”  At P1802L51, add to the end of the paragraph, “If the MSDU(s) received in an MPDU with the DEI subfield set to true are delievered via an MA-UNITDATA.indication, the drop eligibile parameter of the MA-UNITDATA.indication is set to TRUE. Otherwise, the parameter is set to FALSE.” |

Discussion:

This was ACCEPTED in motion 169. However, what does "The default drop\_eligible value is false." mean? It's not marked as an optional parameter. Similarly specious ", if any" at 808.28. More generally, the wording around drop eligibility is a bit haphazard.

Proposed additional changes:

In D3.2:

At 164.24 change “drop eligible” to “drop eligibility”.

At 312.36 change “(#4767)The drop eligible parameter provides guidance on whether this request can be discarded in preference to other requests when there are insufficient resources in a STA. If drop eligible is true,(Ed) the request can be discarded in preference to other requests in which drop eligible is false.(Ed) The default drop eligible value is false.(Ed)” to “(#4767)The drop eligible parameter is a Boolean that indicates whether this MSDU can be discarded in preference to other MSDUs when there are insufficient resources in a STA. If drop eligible is true,(Ed) the MSDU can be discarded in preference to other MSDUs for which drop eligible is false.(Ed)”

At 315.8 change “(#4767)The drop eligible parameter indicates if the received MSDU was designated as DEI in the received protocol headers. If no protocol header designation was received, the parameter value is false.(Ed)” to “(#4767)The drop eligible parameter is a Boolean that indicates whether the received MSDU was designated as drop eligible in the DEI subfield of the HT variant HT Control field of the incoming field, if present. If no HT variant HT Control field was present, the parameter value is false.(Ed)”

At 808.27 change “drop eligible parameter received in” to “drop eligible parameter in”. At 808.28 delete “, if any”.

At 1790.51 change “(#4767-Ed)If MSDU(s) received in an MPDU are delivered via an MA-UNITDATA.indication, the drop eligible parameter of the MA-UNITDATA.indication is set to true if the DEI subfield is equal to 1 and set to false if the DEI subfield is equal to 0.” to “(#4767-Ed)If MSDU(s) received in an MPDU are delivered via an MA-UNITDATA.indication, the drop eligible parameter of the MA-UNITDATA.indication is set to true if the MPDU contains an HT variant HT Control field and the DEI subfield is equal to 1 and set to false otherwise.”

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4625  Mark RISON | Since 1.4 defines x-y as being inclusive, the word "inclusive" is no longer needed for ranges | Delete "inclusive" in Clause 6 (10x), 9.2.2, 9.2.4.6.1 (2x), 9.4.2.5.1, 9.4.2.21.10, 9.4.2.45 (2x), 9.4.2.94, 9.4.2.154, 9.4.2.167 (3x), 10.19, 10.21 (3x), 11.10.14, 12.5.2.3.3, 19.3.9.3.2, 21.3.8.2.1, Table 23-1, dot11TxPowerLevelExtended in C.3 (2x). Delete "(all inclusive)" in 10.3.2.12 (x). Delete " (inclusive)" in 10.47.6 (second instance), 11.3.9.2, Table 21-23 (2x), Table 22-21 (2x), Table 23-29 (2x). Delete ", inclusive" in 12.3.3.3.6 |

Discussion:

This was ACCEPTED in motion 160. However, there is still one suspect “(inclusive)” in 10.47.6:

The PTSF subfield is set to TSF[Partial TSF Offset+4: Partial TSF Offset+11] (inclusive)

The problem is that [x:y] isn’t defined globally, though there are some hints:

* In 1.5: “dec(A[b:c]) is the cast from binary to decimal operator, where c is the least significant bit in binary value [b:c]”
* In 10.21: “AID[b:c] represents bits b to c (#4625)of the AID”
* and a bunch of MAC address slicing contexts.

The first is nearly there but (a) it has a dec() wrapper and (b) it doesn’t *quite* get around to saying what [b:c] actually *means*.

Proposed additional changes:

TBD. Maybe use dec() in 10.47.6 and clarify in 1.5 that [b:c] means bits b to c **inclusive**?

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4245  Mark RISON  10.2.3.2  1717.62 | 10.2.3.2 HCF contention based channel access (EDCA) says defaults are always used ("When communicating within a BSS, the EDCA parameters used are [...] from the default values for the parameters when [...] when the STA is a mesh STA") but various other parts of the spec say that EDCA parameters are passed around in various frames when a STA is a QoS STA (as all mesh STAs are) | As it says in the comment |

Discussion:

SAKODA Kazuyuki has confirmed that:

* “mesh STAs always use EDCA parameters from default value. As such, EDCA parameter or QoS Capability element are not present for mesh BSSs”
* “11ah/11ad STAs will not become a mesh STA”

Proposed changes:

In D3.0:

Change 14.1 Mesh STA dependencies at 2769.27 as follows:

When dot11DMGOptionImplemented or dot11S1GOptionImplemented is true, dot11MeshActivated shall be false.

In 9.4.2.34 QoS Capability element at 1144.21 delete the following sentence:

The QoS Capability element is present in Beacon frames that do not contain the EDCA Parameter Set element and in (Re)Association Request frames.

In the table in 6.3.11.2.2 Semantics of the service primitive [for 6.3.11.2 MLME-START.request] at 397.51 change the rightmost cell of the EDCAParameterSet row as follows:

The initial EDCA parameter set values to be

used in the BSS. The parameter is present if

dot11QosOptionImplemented is true and dot11MeshActivated is false;

otherwise not present.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 4245 in <this document>, which clarify that the EDCA Parameter Set element is not used in MBSSes, and that S1G STAs are not used in MBSSes.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4087  Rojan Chitrakar  12.5.3.4.1  2608.25 | As per 12.5.3.3 (P2607L59), the MIC is also encrypted along with the plaintext MPDU, so it is not possible to obtain the original MIC at this stage. The original MIC can only be obtained after CCM decryption stage. The figure 12-23 is misleading, either it should be clarified that the MIC that is fed into the CCM decryption block is encrypted MIC, or the entire encrypted MPDU (instead of MIC and data) should be passed to the CCM decryption block. | Rectify the Figure 12-23 as per comment. Specifically, the MIC that is fed into the CCM decryption module should be "encrypted MIC" |

Discussion:

It is agreed that the MIC is encrypted in CCMP (though not in GCMP; see CID 4093).

Proposed changes:

In D3.0:

In Figure 12-23—CCMP decapsulation block diagram:

* change "MIC" to "Encrypted MIC"
* change "Data" to "Encrypted data" on the left
* lowercase "Nonce"

In Figure 12-18—CCMP encapsulation block diagram:

* change "Data" to "Plaintext data" on the left
* in the arrow out of CCM encryption change label to put "Encrypted data" above the arrow and “Encrypted MIC” below the arrow
* lowercase "Nonce"

In Figure 12-29—GCMP decapsulation block diagram:

* change "Data" to "Encrypted data" on the left
* lowercase "Nonce" and "Header"

In Figure 12-27—GCMP encapsulation block diagram:

* change "Data" to "Plaintext data" on the left
* in the arrow out of GCM encryption change label to put "Encrypted data" above the arrow and “MIC” below the arrow
* lowercase "Nonce" and "Header"

Change "MIC" to "Encrypted MIC" at 2608.51½ and 2609.9½.

Sharpen up Figure 12-23—CCMP decapsulation block diagram.

In Figure 12-23—CCMP decapsulation block diagram:

* Change PN ’ to Replay counter

In Figure 12-29—GCMP decapsulation block diagram:

* Change PN\* to Replay counter

Change 12.5.3.4.2 CCM recipient processing as follows:

There are ~~four~~five inputs to CCM recipient processing (see Figure 12-23 (CCMP decapsulation block diagram)):

— Key: the temporal key (16 octets).

— Nonce: the nonce (13 octets) constructed as described in 12.5.3.3.3 (Construct AAD(#2720)) b ((11ah)For PV1 MPDUs, the format of the AAD is shown in Figure 12-20 (AAD construction for PV1 MPDUs(M110)(11ah)).).

— Encrypted ~~frame body~~data: the encrypted portion of the frame body from the received MPDU~~. The encrypted frame body includes~~ , excluding the MIC (see Figure 12-16 (Expanded CCMP MPDU) and Figure 12-17 (Expanded PV1 CCMP MPDU)).

— MIC: the encrypted MIC from the received MPDU.

— AAD: the AAD ((11ah)12–30 octets) that is the canonical MPDU header as described in (#2720)12.5.3.3.3 (Construct AAD(#2720)).

[…]

— Frame body: the plaintext frame body, which is 8 octets (CCMP-128) or 16 octets (CCMP-256) smaller than the encrypted portion of the frame body.

Change 12.5.5.4.2 GCM recipient processing as follows:

There are ~~four~~five inputs to GCM recipient processing (see Figure 12-29 (GCMP decapsulation block diagram)):

— Key: the temporal key (16 octets).

— Nonce: the nonce (12 octets) constructed as described in 12.5.5.3.4 (Construct GCM nonce).

— Encrypted ~~frame body~~data: the encrypted portion of the frame body from the received MPDU~~. The encrypted frame body includes a 16-octet MIC~~ (see Figure 12-26 (Expanded GCMP MPDU)).

— MIC: the MIC from the received MPDU.

— AAD: the AAD (22-30 octets) that is the canonical MPDU header as described in 12.5.5.3.3 (Construct AAD).

[…]

— Frame body: the plaintext frame body, which is ~~16 octets smaller than~~the same size as the encrypted portion of the frame body.

In S.1 change “encrypted Frame Body” to “encrypted portion of the frame body”.

TBC: the CCMP test vectors with unclear MIC size.

Proposed resolution:

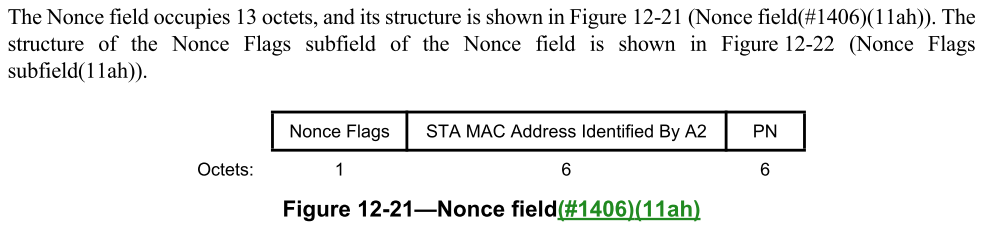
REVISED

Make the changes shown under “Proposed changes” for CID 4087 in <this document>, which tidy up the description of encryption decapsulation and recipient processing.

|  |  |  |
| --- | --- | --- |
| Identifiers | Comment | Proposed change |
| CID 4612  Mark RISON  12  2609.5 | "4) The nonce(#1406) value is constructed from the STA MAC Address Identified By A2, PN, and Nonce Flags fields." is just duplication of Figure 12-21--Nonce field. Ditto duplication of Figure 12-28--Nonce field for GCMP | Replace the cited text, and "3) (11ah)The nonce(#1406) value is constructed from the A2, PN, and Nonce Flags fields." in 12.5.3.4.1 and "c) The nonce(#1406) value is constructed from the A2 and PN fields." in 12.5.5.4.1, with references to the figures. Also remove the "The Nonce field has an internal structure of Nonce Flags || (11ah)STA MAC Address Identified By A2 || PN" and "The Nonce field has an internal structure of A2 || PN" duplication (of figures immediately above!) |

Discussion:

Duplication is baaad, m’kay? For instance:



v. (just 10 lines down!)

The Nonce field has an internal structure of Nonce Flags || STA MAC Address Identified By A2 || PN

v. (2 pages later)

3) The nonce value is constructed from the A2, PN, and Nonce Flags fields.

Proposed changes:

In D3.0:

At 2608.45 change:

1) (11ah)The encrypted MPDU is parsed to construct the AAD and nonce values.

2) (11ah)The AAD is formed from the MPDU header of the encrypted MPDU.

3) (11ah)The nonce(#1406) value is constructed from the A2, PN, and Nonce Flags fields.

to:

1) The encrypted MPDU is parsed to construct the AAD (see 12.5.3.3.3 (Construct AAD)) and nonce (see 12.5.3.3.4 (Construct CCM nonce)) values

and renumber the following list item numbers.

[TBD: and similarly for CCMP PV1 and for GCMP]

At 2606.23 change:

The Nonce field has an internal structure of Nonce Flags || (11ah)STA MAC Address Identified By A2 || PN, where

— The Priority subfield of the Nonce Flags field shall be set to the priority value of the MPDU.

— When management frame protection is negotiated, the Management field of the Nonce Flags field shall be set to 1 if the (11ah)PV0 MPDU’s Type field of the Frame Control field is 00 (Management frame) (11ah)or the PV1 MPDU’s Type field of the Frame Control field is 001 (Management frame); otherwise, it shall be set to 0.

— (11ah)The PV1 subfield of the Nonce Flags field shall be set to 1 when the Protocol Version field of the Frame Control field of the MPDU header is equal to 1. The PV1 subfield of the Nonce Flags field shall be set to 0 otherwise.

— Bits (11ah)6 to 7 of the Nonce Flags field shall be set to 0.

— (11ah)STA MAC Address Identified By A2 field occupies octets 1–6. This shall be encoded with the octets ordered with (11ah)STA MAC Address Identified By A2 octet 0 at octet index 1 and (11ah)STA MAC Address Identified By A2 octet 5 at octet index 6.

— The PN field occupies octets 7–12. The octets of PN shall be ordered so that PN0 is at octet index 12 and PN5 is at octet index 7.

to:

The Priority subfield shall be set to the priority value of the MPDU.

The Management subfield shall be set to 1 if the MPDU is a Management frame and management frame protection is negotiated; otherwise, it shall be set to 0.

The PV1 subfield shall be set to 1 for a PV1 frame; otherwise, it shall be set to 0.

The Zeros subfield shall be set to 0.

The STA MAC Address Identified By A2 subfield shall contain the Address 2 field from the MAC header for PV0 MPDUs and the MAC address identified by the A2 field in the MAC header for PV1 MPDUs.

The PN subfield shall contain the packet number, with PN0 at the highest octet index.

At 2616.55 change:

The Nonce field has an internal structure of A2 || PN, where

— MPDU address A2 field occupies octets 0 to 5. This shall be encoded with the octets ordered with A2 octet 0 at octet index 0 and A2 octet 5 at octet index 5.

— The PN field occupies octets 6 to 11. The octets of PN shall be ordered so that PN0 is at octet index 11 and PN5 is at octet index 6.

to:

The A2 subfield shall contain the Address 2 field from the MAC header.

The PN subfield shall contain the packet number, with PN0 at the highest octet index.

Proposed resolution:

REVISED

Make the changes shown under “Proposed changes” for CID 4612 in <this document>, which make changes in the direction suggested by the commenter.

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

Discussion:

Proposed changes:

Proposed resolution:

REVISED

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

**References:**

802.11md/D3.0 except where otherwise specified