IEEE P802.11
Wireless LANs

|  |
| --- |
| Multipurpose Alternate Replay Counters |
| Date: 2024-03-12 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Henry Ptasinski | Element78 Communications LLC |  |  | henry@e78com.com |

Abstract

This submission presents a proposed resolution for the following P802.11REVme CIDs:

 7028

regarding issues with protection of QMFs with GCMP.

It also provides a multipurpose solution that can be used to help address the following CIDs from LB281 on P802.11bf:

 4187

 4188

regarding replay sequence counter management for Protected Sensing frames.

The proposed changes are based on Draft P802.11REVme\_D5.0.pdf.

Revision history:

R0 – Initial version

R1 – Removed option 1; changed Extended RSN Capabilities bit usage to defer to ANA

R2 – Corrected number of bits included in AAD construction

## P802.11REVme CID 7038:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Page** | **Line** | **Comment** | **Proposed Change** |
| 7028 | 12.5.4.3.4 | 3040 | 17 | The security of QMF depends on the ACI of the MMPDU being protected. This is similar to how the security of QoS for Data frames depends on the TID being protected. ACI is protected when using CCMP, but not when using GCMP.ACI is encoded in the ACI subfield in the Sequence Number field. That field is masked out from AAD construction for both CCMP and GCMP and as such, AAD does not protect it (unlikely it does for QoS Data frames with QC being included).CCMP protects the ACI subfield value by defining the priority value of the MPDU to be equal to its value (P3021 L28). This priority value is then included in the CCM nonce (see Figure 12-21) and that provides protection to the ACI value. GCMP on the other hand does not include the priority value in the GCM nonce (see Figure 12-30). Consequently, there is no protection for the ACI value and attacker can modify it without the frame recipient being able to detect the modification based on GCMP processing. This enabled attacks that could be used to reorder Robust Management frames between different access categories.Protection of ACI with GCMP is inconvenient since there is no room in the GCM nonce for the priority value. The standard could be extended to construct the AAD for GCMP to include a new field for the QMF cases (e.g., a "virtual" octet with the ACI encoded in it at the end of the AAD) or by not masking the ACI subfield of the Sequence Number field in QMFs. This would make the AAD construction different for GCMP compared to CCMP (since we should not change CCMP definition for this and break compatibility with the original design). It might be acceptable to modify GCMP for QMF due to limited, if any, interest in deploying QMF so far. It would also be possible to negotiate use of the extended AAD for GCMP when QMF is used. That said, if there is no interest in deploying QMF, there may not be much benefit from coming up with more complex solutions for this than simply disallow use of QMF with GCMP.This comment proposes an unconditional change to the AAD construction for GCMP for QMFs to unmask the ACI field. This is not compatible with previous definition. However, this is believed to be acceptable due to no known deployment of QMF with GCMP. This comment could be satisfied with a similar change done based on negotiated capability (e.g., and RSNXE bit) or by disallowing use of QMF with GCMP. | At P3023 L45-46, replace "SC – MPDU Sequence Control field, with the Sequence Number subfield (bits 4–15 of the Sequence Control field) masked out"with"SC – MPDU Sequence Control field, with the QMF Sequence Number field (bits 4-13 of the Sequence Control field) masked out in QMFs and with the Sequence Number subfield (bits 4–15 of the Sequence Control field) masked out in frames that are not QMFs" . |

## P802.11bf CIDs 4187 and 4188:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Clause** | **Page** | **Line** | **Comment** | **Proposed Change** |
| 4187 | 12.5.4.4.4 | 201 | 15 | The Sequence Number field is not protected by CCMP or GCMP. The replay counter must be selected based on protected fields. | Define a new key specifically for Protected Sensing Frames. |
| 4188 | 12.5.2.4.4 | 200 | 37 | The Sequence Number field is not protected by CCMP or GCMP. The replay counter must be selected based on protected fields. | Define a new key specifically for Protected Sensing frames. |

## Discussion:

For Protected Sensing frames, 11bf requires the use of a special replay counter that’s distinct from the QMF replay counters. Unlike FTM, 11bf does not add a bit in the CCMP header to indicate that the frame is a Sensing frame vs. some other Action frame. There seem to be a number of issues with respect to replay counters for QMFs, part stemming from 11bf, part from FTM, and part from GCMP support for QMFs in the baseline:

ACI not protected in GCMP:

For QMFs, the ACI is used to select the replay counter (REVme Clause 12.5.2.4.4 PN and replay detection, step (f)). The ACI is bits 10-11 of the Sequence Number field (REVme 9.2.4.4.3 Sequence Number field Figure 9-9), so it’s bits 14-15 of the Sequence Control field. But those bits are masked out when the AAD is constructed (REVme 12.5.2.3.3 Construct AAD):

5) SC – MPDU Sequence Control field, with the Sequence Number subfield (bits 4–15 of the Sequence Control field) masked out. The Fragment Number subfield is not modified.

The AAD construction is performed the same for CCMP and GCMP.

Since the ACI is not included in the AAD, CCMP protects the ACI by including it in the Priority field of the CCM Nonce Flags field. The GCM nonce, however, does not include a priority field, so ACI is not included in the nonce construction. As a result, the ACI is not protected at all when GCMP is used.

*Disallowing GCMP for QMF would mean that GCMP could not be used by STAs when FTM or Sensing are enabled.*

No early discard of replayed protected Action frames:

Unlike FTM, 11bf does not include a bit in the CCMP header to identify Protected Sensing frames. As a result, selection of the correct replay counter (the sensing-specific one vs. the regular QMF replay counter for the ACI of the frame) cannot be done until after the frame has been decrypted. However, the standard explicitly allows discard before decrypt (REVme 12.5.2.4.4 and 12.5.4.4.4). Discard before decrypt allows receiving STAs to save power by not decrypting frames that will be immediately discarded post-decrypt.

Liminted number of replay counters for QMFs:

For QMFs, one replay counter per ACI is specified, so only 4 replay counters are available. For QoS Data frames, there are up to 16 replay counters, one per TID.

Scalability:

One bit of the few remaining in the CCMP Header was already consumed for FTM. Adding another bit for Protected Sensing frames would chip away at the remaining reserved bits, and is not scalable to future services that need similar functionality. A multipurpose approach should be used that can be used by multiple services without individually allocating CCMP Header bits for each service.

This proposal defines a multipurpose alternate replay counter (MARC) solution which adds four alternative counters which can be used with in addition to the existing four ACI replay counter for robust IQMFs.

The RSNXE includes signaling of MARC capability. When a PTK is derived, MARC is enabled for robust IQMFs protected with that PTK if both devices support MARC.

When MARC is not enabled for a PTK, the legacy behavior is used for robust IQMFs including FTM. When MARC is enabled for a PTK, the CCMP/GCMP headers are modified to include a MARC flag and replay counter index. When MARC is enabled for a PTK, the ACI and the MARC Index are included in the AAD for both CCMP and GCMP.

Receiver processing of IQMFs can be summarised as:

 if MARC not enabled (legacy behavior):

 use legacy AAD construction

 if FTM signalled in header

 use FTM replay counter

 else

 use replay counter selected by ACI

 else (MARC enabled):

 use MARC AAD construction

 if MARC signaled in header

 use replay counter selected by MARC Index

 else

 use replay counter selected by ACI

## Proposed Resolution for CID 7028:

REVISED. Request the TGme editor to apply the changes below:

### 3.4 Acronyms and abbreviations

Add the following acronym:

MARC Multipurpose Alternate Replay Counters

### 9.4.2.240 RSNXE

Add an entry for MARC to Table 9-371 and update the Reserved bits as needed:

**Table 9-371—Extended RSN Capabilities field**

|  |  |  |
| --- | --- | --- |
| <ANA> | MARC | A STA sets the MARC field to 1 if dot11MultipurposeAlternateReplayCountersActivated is true. Otherwise, it sets the field to 0. See 11.24.X. |

Insert a new subclause at the end of clause 11.24:

### 11.24.X Multipurpose Alternate Replay Counters (MARC)

A STA with dot11MultipurposeAlternateReplayCountersActivated set to true shall set the MARC field to 1 in transmitted any RSNXE.

When a pair of STAs establish a PTKSA, if the MARC field is set to1 in the RXNSEs from both STAS, MARC shall be enabled for the resulting PTK.

### 12.5.2.2 CCMP MPDU format

Change the text and Figure 12-15 as shown:

Figure 12-15 (Expanded CCMP MPDU) depicts the MPDU when using CCMP.

**B0 B1 B2 B3 B4 B5 B6 B7**

**MAC Header**

**CCMP Header**

8 octets

**Data (PDU)**

≥ 1 octet

**FCS**

4 octets

**MIC**

variable

**PN0**

**PN1**

**Rsvd**

**Rsvd**

**FTM/MARC**

**Ext**

**IV**

**Key**

**ID**

**PN2**

**PN3**

**PN4**

**PN5**

Key ID octet

Encrypted

**MARC Index**

Figure 12-15—Expanded CCMP MPDU

For secure PV0 MPDUs, CCMP-128 processing expands the original MPDU size by 16 octets, 8 octets for the CCMP Header field and 8 octets for the MIC field. CCMP-256 processing expands the original MPDU size by 24 octets, 8 octets for the CCMP Header field, and 16 octets for the MIC field. The CCMP Header field is constructed from the PN, ExtIV, and Key ID subfields. PN is a 48-bit PN represented as an array of 6 octets. PN5 is the most significant octet of the PN, and PN0 is the least significant.

The third octet of the CCMP Header field is reserved.

The ExtIV subfield (bit 5) of the Key ID octet is always set to 1 for CCMP.

Bits 6–7 of the Key ID octet are for the Key ID subfield. ~~The remaining bits of the Key ID octet are reserved.~~

In a protected individually addressed ~~m~~Management Action frame when MARC is not enabled for the PTK, bit 4 of the Key ID octet equals 1 if the frame is a Protected Fine Timing frame—see Table 9-51 (Category values). In a robust IQMF when MARC is enabled for the PTK, bit 4 of the Key ID octet equals 1 if an alternate replay counter is used for the frame and set to 0 otherwise. Bits 2 and 3 of the Key ID octet are the MARC Index and are set to the index of the alternate replay counter if one is used for the frame and set to 0 otherwise. In other protected individually addressed frames, and in all protected group addressed frames, bits 2 to 4 are reserved.

The remaining bits of the Key ID octet are reserved.

The CCMP header is not included in secure PV1 MPDUs, but constructed locally at the STA as defined in 12.5.2.3.6 (Construct CCMP header for PV1 MPDUs). For secure PV1 MPDUs, CCMP-128 processing expands the original MPDU size by 8 octets for the MIC field. CCMP-256 processing expands the original MPDU size by 16 octets for the MIC field. Figure 12-16 (Expanded PV1 CCMP MPDU) depicts the PV1 MPDU when using CCMP.

PV1 MAC Header

Data (PDU)

≥ 1 octet

MIC

FCS

Encrypted

Figure 12-16—Expanded PV1 CCMP MPDU

### 12.5.2.3.3 Construct AAD

Change the first paragraph as shown:

1. For PV0 MPDUs, the format of the AAD is shown in Figure 12-18 (AAD construction for PV0 MPDUs). The length of the AAD for PV0 varies depending on the presence or absence of the QC/MARC and A4 fields and is shown in Table 12-3 (AAD length for PV0 MPDUs).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | FC | A1 | A2 | A3 | SC | A4 | QC/MARC |
| Octets: | 2 | 6 | 6 | 6 | 2 | 6 | 2 |
| Figure 12-18—AAD construction for PV0 MPDUs |

|  |
| --- |
| Table 12-3—AAD length for PV0 MPDUs |
| QC/MARC field | A4 field | AAD length(octets) |
| Absent | Absent | 22 |
| Present | Absent | 24 |
| Absent | Present | 28 |
| Present | Present | 30 |

Change step (a)(7) as shown:

1. QC/MARC – For robust IQMFs when MARC is not enabled for the PTK, this field is not present. For robust IQMFs when MARC is enabled for the PTK, bits 0 to 1 are set to the value in the ACI subfield of the Sequence Number field, and bits 2 to 3 are set to the MARC Index value of the Key ID octet in the CCMP Header or GCMP Header of the frame, and the remaining bits are reserved. Otherwise, MDPU QoS Control field contains the MSDU priority, if present. The QC TID is used in the construction of the AAD. When in a non-DMG BSS, if both the STA and its peer have their SPP A-MSDU Capable subfields (see 9.4.2.240 (RSNXE)) equal to 1, the A-MSDU Present field is also used in the construction of the AAD. When in a DMG BSS, the A-MSDU Present field and A-MSDU Type field are also used in the construction of the AAD. The remaining QC fields are not used and are masked out for the AAD calculation (for a non-DMG BSS, bits 4 to 6, bits 8 to 15, and bit 7 when either the STA or its peer has the SPP A-MSDU Capable field equal to 0; for a DMG BSS, bits 4 to 6 and bits 9 to 15). When in a DMG BSS, the A-MSDU Present bit 7 and A-MSDU Type bit 8 are used in the construction of the AAD, and the remaining QC fields are masked out for the AAD calculation (bits 4 to 6, bits 9 to 15).

### 12.5.2.3.7 CCM originator processing

Change the text starting at the 8th paragraph as shown:

The transmitter shall preserve the order of protected robust Management frames that are transmitted to the same ~~DA~~RA without the QMF service. When the QMF service is used when MARC is not enabled for the PTK, the transmitter shall not reorder robust IQMFs within an AC when the frames are transmitted to the same RA. When the QMF service is used when MARC is enabled for the PTK, the transmitter shall not reorder robust IQMFs that are transmitted to the same RA within a replay counter (either an AC-specific replay counter or an alternate replay counter), but may reorder frames across replay counters.

NOTE —The transmitter may use the alternate replay counters to support various features, such as Fine Timing Measurement. Alternate replay counters are not statically mapped to any specific features, and the transmitter may select any alternate replay counter for use as needed, as long as the re-ordering restrictions above are satisfied.

A CCMP protected individually addressed robust Management frame shall be protected using the same TK as a Data frame.

### 12.5.2.4.4 PN and replay detection

Change the text as shown:

To effect replay detection, the receiver extracts the PN from the CCMP header.

NOTE 1—The CCMP header is not present in secure PV1 MPDUs, but constructed locally at the STA as defined in 12.5.2.3.6 (Construct CCMP header for PV1 MPDUs).

See 12.5.2.2 (CCMP MPDU format) for a description of how the PN is encoded in the CCMP header. The following processing rules are used to detect replay:

1. The receiver shall maintain a separate set of replay counters for each PTKSA, TPKSA, GTKSA, mesh PTKSA, and mesh GTKSA. The receiver initializes these replay counters to 0 when it resets the TK, TPK-TK or MTK for a peer, and to the value indicated by the peer when it sets the GTK or MGTK. The replay counter is set to the PN value of accepted CCMP MPDUs.
2. For each PTKSA, TPKSA, GTKSA, mesh PTKSA, and mesh GTKSA, the receiver shall maintain a separate replay counter for each TID, subject to the limitation of the number of supported replay counters indicated in the RSN Capabilities field (see 9.4.2.23 (RSNE)). In the case of a TPKSA, this shall for both the TDLS initiator STA and the TDLS responder STA be the number indicated by the TDLS initiator STA in the PTKSA Replay Counter field in the TDLS Setup Request frame.

NOTE 2—The number indicated by the TDLS responder STA (if a TDLS Discovery Response frame is sent) is ignored, as is the GTKSA Replay Counter field in the TDLS Setup Request frame and any TDLS Discovery Response frame.

NOTE 3—For the purpose of replay detection, non-QoS Data frames are treated as having TID 0, and use the replay counter corresponding to MSDU priority 0.

1. For each PTKSA, i~~I~~f management frame protection ~~is~~has been negotiated, the receiver shall set the MFPC bit on a given link to 1, it shall maintain a single replay counter for received individually addressed robust PV0 Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) that are received with the To DS subfield equal to 0, and (S1G STA only) a single replay counter for received individually addressed robust PV1 Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)).
2. ~~If dot11RSNAProtectedManagementFramesActivated is true~~For each PTKSA, if management frame protection has been negotiated and dot11QMFActivated is also true, the receiver shall maintain an additional replay counter for each ACI for received individually addressed robust PV0 Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) that are received with the To DS subfield equal to 1.

NOTE 4—Separate replay counters for PV0 and PV1 Management frames allow for reordering between the two types. However, S1G STAs are required to use PV1 Management frames for individually addressed Action (and Action No Ack) frames when the peer is known to support them (see 10.57 (Generation of PV1 MPDUs and header compression procedure)), so there is no issue with PV0 Action (and Action No Ack) frames. The other robust Management frames are Deauthentication and Disassociation frames, but reordering of a PV1 Action frame and a Deauthentication/Disassociation frame is not of much concern since the Action frame is not valid after deauthentication/disassociation.

NOTE 5—QMF is not supported for PV1 Management frames (see 11.24.1.1 (Overview)).

1. ~~If dot11RSNAProtectedManagementFramesActivated is true~~For each PTKSA, if management frame protection has been negotiated, MARC has not been enabled, and at least one of dot11FineTimingMsmtRespActivated or dot11FineTimingMsmtInitActivated are true, the ~~recipient~~receiver shall maintain a separate replay counter for received~~ing~~ individually addressed Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) ~~and shall use the PN from the received frame to detect replays~~.
2. For each PTKSA, if management frame protection has been negotiated and MARC has been enabled, the receiver shall maintain an alternate replay counter for each MARC Index for received robust IQMFs that have the FTM/MARC bit set to 1 in the CCMP Header.
3. The receiver shall discard any Data frame that is received with its PN less than or equal to the value of the replay counter that is associated with the TA, RA (individual or group address; not if TDLS) and priority value of the received MPDU. The receiver shall discard fragmented MSDUs, A-MSDUs and MMPDUs whose constituent MPDU PN values are not incrementing in steps of 1. If management frame protection is negotiated, the receiver shall set the MFPC bit on a given link to 1, it shall discard any individually addressed robust Management frame that is received with its PN less than or equal to the value of the replay counter associated with the TA, (QMF receiver of an individually addressed robust PV0 Management frame with the To DS subfield equal to 1 only) ACI, (QMF receiver of a robust IQMF with FTM/MARC bit set to 1 in the CCMP Header only) MARC Index, and (S1G STA only) Protocol Version subfield of that individually addressed Management frame. The receiver should discard received protected Management frames if the associated replay counter does not exist.
4. When discarding a frame, the receiver shall increment by 1 dot11RSNAStatsCCMPReplays for Data frames or dot11RSNAStatsRobustMgmtCCMPReplays for robust Management frames.
5. For MSDUs or A-MSDUs sent using the block ack feature, reordering of received MSDUs or A-MSDUs according to the block ack receiver operation is performed prior to replay detection
6. If the receiver performs replay detection prior to decryption, then the receiver shall check that the replay counter used to detect replays is correct and discard the frame if incorrect. ~~In particular, the separate replay counter for individually addressed Protected Fine Timing frames shall be used if and only if the FTM subfield of CCMP Header (Figure 12-15 (Expanded CCMP MPDU)) signals that the management PDU is a Protected Fine Timing frame.~~ The replay counter shall not be updated unless the decryption is successful and the frame is accepted.

### 12.5.4.2 GCMP MPDU format

Figure 12-28 (Expanded GCMP MPDU) shows the MPDU format when using GCMP.

**B0 B1 B2 B3 B4 B5 B6 B7**

**MAC Header**

**GCMP Header**

8 octets

**Data (PDU)**

≥ 1 octet

**FCS**

4 octets

**MIC**

variable

**PN0**

**PN1**

**Rsvd**

**Rsvd**

**FTM/MARC**

**Ext**

**IV**

**Key**

**ID**

**PN2**

**PN3**

**PN4**

**PN5**

Key ID octet

Encrypted

**MARC Index**

Figure 12-28—Expanded GCMP MPDU

GCMP processing expands the original MPDU size by 24 octets, 8 octets for the GCMP Header field and 16 octets for the MIC field. The GCMP Header field is constructed from the PN and Key ID subfields.

The 48-bit PN is represented as an array of 6 octets. PN5 is the most significant octet of the PN, and PN0 is the least significant.

The ExtIV subfield (bit 5) of the Key ID octet is always set to 1 for GCMP.

The third octet of the GCMP Header field is reserved.

Bits 6–7 of the Key ID octet are for the Key ID subfield.

In a protected individually addressed management Action frame when MARC is not enabled for the PTK, bit 4 of the Key ID octet is set to 1 if the frame is a Protected Fine Timing frame—see Table 9-81 (Category values). In a robust IQMF when MARC is enabled for the PTK, bit 4 of the Key ID octet equals 1 if an alternate replay counter is used for the frame and set to 0 otherwise. Bits 2 and 3 of the Key ID octet are the MARC Index and are set to the index of the alternate replay counter if one is used for the frame and set to 0 otherwise. In other protected individually addressed frames, and in all protected group addressed frames, bits 2 to 4 are reserved.

The remaining bits of the Key ID octet are reserved.

### 12.5.4.3.6 GCM originator processing

Change the text starting at the 9th paragraph as shown:

When the QMF service is not used, the transmitter shall preserve the order of protected individually addressed robust Management frames that are transmitted to the same RA. When the QMF service is used when MARC is not enabled for the PTK, the transmitter shall preserve the order of protected robust IQMFs within an AC that are transmitted to the same RA. When the QMF service is used when MARC is enabled for the PTK, the transmitter shall not reorder robust IQMFs that are transmitted to the same RA within a replay counter (either an AC-specific replay counter or an alternate replay counter), but may reorder frames across replay counters.

NOTE —The transmitter may use the alternate replay counters to support various features, such as Fine Timing Measurement. Alternate replay counters are not statically mapped to any specific features, and the transmitter may select any alternate replay counter for use as needed, as long as the re-ordering restrictions above are satisfied.

A GCMP protected individually addressed robust Management frame shall be protected using the same TK as a Data frame.

### 12.5.4.4.4 PN and replay detection

Change the text as shown:

To effect replay detection, the receiver extracts the PN from the GCMP header. See 12.5.4.2 (GCMP MPDU format) for a description of how the PN is encoded in the GCMP header. The following processing rules are used to detect replay:

1. The receiver shall maintain a separate set of replay counters for each PTKSA, TPKSA, GTKSA, mesh PTKSA, and mesh GTKSA. The receiver initializes these replay counters to 0 when it resets the temporal key for a peer. The replay counter is set to the PN value of accepted GCMP MPDUs.
2. For each PTKSA, TPKSA, GTKSA, mesh PTKSA, and mesh GTKSA, the receiver shall maintain a separate replay counter for each TID, subject to the limitation of the number of supported replay counters indicated in the RSN Capabilities field (see 9.4.2.23 (RSNE)). In the case of a TPKSA, this shall for both the TDLS initiator STA and the TDLS responder STA be the number indicated by the TDLS initiator STA in the PTKSA Replay Counter field in the TDLS Setup Request frame.

NOTE 1—The number indicated by the TDLS responder STA (if a TDLS Discovery Response frame is sent) is ignored, as is the GTKSA Replay Counter field in the TDLS Setup Request frame and any TDLS Discovery Response frame.

NOTE 2—For the purpose of replay detection, non-QoS Data frames are treated as having TID 0, and use the replay counter corresponding to MSDU priority 0.

1. For each PTKSA, i~~I~~f management frame protection ~~is~~has been negotiated, the receiver shall set the MFPC bit on a given link to 1, it shall maintain a single replay counter for received individually addressed robust Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) that are received with the To DS subfield equal to 0, and a single replay counter for received individually addressed robust PV1 Management frames except PV1 Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)).
2. I~~f dot11RSNAProtectedManagementFramesActivated is true~~For each PTKSA, if management frame protection has been negotiated and dot11QMFActivated is also true, the receiver shall maintain an additional replay counter for each ACI for received individually addressed robust PV0 Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) ~~and robust PV1 Management frames except Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details))~~ that are received with the To DS subfield equal to 1.

NOTE 3—PV1 frames are not supported with GCMP (see 12.5.4.1 (GCMP overview)).

1. ~~If dot11RSNAProtectedManagementFramesActivated is truetrue~~For each PTKSA, if management frame protection has been negotiated, MARC has not been enabled, and at least one of dot11FineTimingMsmtRespActivated or dot11FineTimingMsmtInitActivated are true, the ~~recipient~~receiver shall maintain a separate replay counter for received individually addressed Protected Fine Timing frames (see 9.6.34 (Protected Fine Timing frame details)) ~~and shall use the PN from the received frame to detect replays~~.
2. For each PTKSA, if management frame protection has been negotiated and MARC has been enabled, the receiver shall maintain an alternate replay counter for each MARC Index for received robust IQMFs that have the FTM/MARC bit set to 1 in the GCMP Header.
3. The receiver shall discard any Data frame that is received with its PN less than or equal to the value of the replay counter that is associated with the TA, RA (individual or group address; not if TDLS) and priority value of the received MPDU. The receiver shall discard fragmented MSDUs, A-MSDUs and MMPDUs whose constituent MPDU PN values are not incrementing in steps of 1. If management frame protection is negotiated, the receiver shall set the MFPC bit on a given link to 1, it shall discard any individually addressed robust Management frame that is received with its PN less than or equal to the value of the replay counter associated with the TA, ~~and~~ (QMF receiver of an individually addressed robust Management frame with the To DS subfield equal to 1 only) ACI, (QMF receiver of a robust IQMF with FTM/MARC bit set to 1 in the GCMP Header only) MARC Index of that individually addressed Management frame.The receiver should discard received protected Management frames if the associated replay counter does not exist.
4. When discarding a frame, the receiver shall increment by 1 dot11RSNAStatsGCMPReplays for Data frames or dot11RSNAStatsRobustMgmtGCMPReplays for robust Management frames.
5. For MSDUs or A-MSDUs sent using the block ack feature, reordering of received MSDUs or A-MSDUs according to the block ack receiver operation is performed prior to replay detection.
6. If the receiver performs replay detection prior to decryption, then the receiver shall discard any frame received with its PN less than or equal to the value of the associated replay counter. The replay counter shall not be updated unless the decryption is successful and the frame is accepted.

### C.3 MIB detail

Add the following entry to the end of the Dot11StationConfigEntry{} list:

 dot11MultipurposeAlternateReplayCountersActivated TruthValue

Add the following new entry to the dot11StationConfig TABLE:

dot11MultipurposeAlternateReplayCountersActivated OBJECT-TYPE

 SYNTAX TruthValue

 MAX-ACCESS read-write

 STATUS current

 DESCRIPTION

"This is a control variable.

It is written by the SME or external management entity.

Changes take effect for the next MLME-START.request primitive or MLME-JOIN.request primitive.

The purpose of dot11MultipurposeAlternateReplayCountersActivated is to enable the use of alternate CCMP and GCMP replay counters for frame sequences that need to have replay detection handled separately from frame sequences that use the normal replay counters."

 DEFVAL { false }

::= { dot11StationConfigEntry <ANA> }

Change the dot11QMFComplianceGroup as shown:

dot11QMFComplianceGroup OBJECT-GROUP

 OBJECTS {

 dot11QMFActivated,

 dot11QMFReconfigurationActivated,

 dot11QMFPolicyChangeTimeout,

 dot11MultipurposeAlternateReplayCounters }

 STATUS current

 DESCRIPTION

 "This object group provides the objects from the IEEE 802.11 MIB required

 to manage QoS management Frame functionality."

 ::= { dot11Groups 63 }

# References:

[Draft P802.11bf\_D3.0.pdf](https://grouper.ieee.org/groups/802/11/private/Draft_Standards/11bf/Draft%20P802.11bf_D3.0.pdf)

[Draft P802.11REVme\_D5.0.pdf](https://grouper.ieee.org/groups/802/11/private/Draft_Standards/11me/Draft%20P802.11REVme_D5.0.pdf)