IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Some miscellaneous CIDs of D7.0 – Part 1 | | | | |
| Date: 2016-04-17 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Alfred Asterjadhi | Qualcomm Inc. | 5775 Morehouse Dr, San Diego, CA 92109 | +1-858-658-5302 | aasterja@qti.qualcomm.com |
| Jouni Malinen | Qualcomm Inc. |  |  | jouni@qca.qualcomm.com |
| Menzo Wentink | Qualcomm Inc. |  |  | mwentink@qti.qualcomm.com |

Abstract

This submission proposes resolutions for multiple comments related to TGah D7 with the following CIDs:

* 10010, 10011, 10012, 10013, 10014, 10015, 10016, 10017

Revisions:

* Rev 0: Initial version of the document.

Interpretation of a Motion to Adopt

A motion to approve this submission means that the editing instructions and any changed or added material are actioned in the TGah Draft. This introduction is not part of the adopted material.

***Editing instructions formatted like this are intended to be copied into the TGah Draft (i.e. they are instructions to the 802.11 editor on how to merge the text with the baseline documents).***

***TGah Editor: Editing instructions preceded by “TGah Editor” are instructions to the TGah editor to modify existing material in the TGah draft. As a result of adopting the changes, the TGah editor will execute the instructions rather than copy them to the TGah Draft.***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CID** | **Commenter** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 10010 | Malinen, Jouni | 386.02 | The CCMP AAD description seems to claim that the Type subfield is only two bits (bits 3 and 4) in a PV1 frame. That does not match the definition of the Frame Control field (Figure 9-742) for a PV1 frame (9.8.3).  In addition, there does not seem to be any need for masking the Type subfield bits in case of PV1. The AAD design is stronger if unnecessary masking is removed.  Note: Approving this comment will result in a change needed to the CCMP test vector for PV1. I can provide such an update once the group has decided how to address all my comments that are applicable for CCMP. | On page 386 line 2, delete  "1) Type subfield (bits 3, 4) in a Data MPDU masked to 0"  and renumber the following items in the list. | Accepted |
| 10011 | Malinen, Jouni | 201.25 | The Header Compression frame is defined as an S1G Action frame. Table 9-47 indicates that this Action frame category (22) is not Robust. As such, the Header Compression frame cannot be protected and a STA receiving such a frame would need to accept it without being able to verify that it is from an authorized station. Since this frame can be used to update header compression state, an attacker could use this frame to replace A3 and A4 information in frames with arbitrary values. Since the current CCMP AAD construction for PV1 frames does not protect A3/A4 fields in header compressed frames, this would be a critical security flaw in the design.  There does not seem to be any technical constraint that should prevent making the Header Compression frame a Robust Action frame. That would allow the frame to be protected by negotiating use of management frame protection. | Replace the Category of the Header Compression frame to a value that is defined as a Robust Action frame (either use an existing Category with Robust=Yes in Table 9-47 or defined a new category for this). | Revised ---  Agree in principle with the comment. Proposed resolution is to move the Header Compression frame to a Robust Action frame and use a similar terminology that is used by DMG to differentiate between S1G and Unprotected S1G.  TGah editor to make the changes shown in 11-16/0539r0 under all headings that include CID 10011. |
| 10012 | Malinen, Jouni | 350.06 | (Re)Association Request frames are not protected even if management frame protection is negotiated for the association. This makes it problematic to allow the Header Compression element to be included in (Re)Association Request/Response frame when using management frame protection. An attacker could use those unprotected frames to inject arbitrary A3/A4 values into header compression state and the current CCMP AAD construction for PV1 frames would not notice this.  It would be good to disallow use of unprotected Header Compression element when management frame protection is being negotiated for the association to avoid such an attack. | On page 350 line 5, replace  "An S1G STA with dot11PV1MACHeaderOptionImplemented equal to true may include a Header Compression element in (Re) Association Request frames, (Re) Association Response frames and in Header Compression frames."  with  "An S1G STA with dot11PV1MACHeaderOptionImplemented equal to true may include a Header Compression element in Header Compression frames and when management frame protection is not negotiated for the association, in (Re) Association Request frames and (Re) Association Response frames." | Revised –  Agree in principle with the commenter. The proposed resolution accounts for the suggested change.  TGah editor to make the changes shown in 11-16/0539r0 under all headings that include CID 10012. |
| 10013 | Malinen, Jouni | 619.19 | I have number of comments that request changes to the CCMP AAD construction. If those changes (or other changes to the way AAD is constructed) are accepted, the CCMP PV1 test vector becomes invalid. I can provide an updated version of the test vector once the group has decided how to address the CCMP related comments. | Update the CCMP PV1 test vector, if needed. A submission with the updated values will be provided as soon as the exact CCMP changes are determined. | Revised –  Agree in principle with the commenter. The proposed resolution accounts for the suggested changes.  TGah editor to make the changes shown in 11-16/0539r0 under all headings that include CID 10013. |
| 10014 | Malinen, Jouni | 539.47 | The PICS entries FT47.11 (Header Compression Request frame) and FT47.12 (Header Compression Response frame) use a frame name that is not defined anywhere in P802.11ah/D7.0. Only the "Header Compression frame" is defined and it looks like that same frame is used both as a request and response. Since FT47.11 and FT47.12 have different Status values, it is not immediately clear to me how these should be combined, i.e., it may not be fine to merge them into a single entry if the different Status column value is needed.  Similarly, FR48.11 is for the undefined Header Compression Request frame, but in that case, there is no corresponding Response frame mentioned. | On page 539 line 47, replace the MAC frame column value for FT47.11 "Header Compression Request frame" with "Header Compression frame as a request".  On page 539 line 49, replace the MAC frame column value for FT47.12 "Header Compression Response frame" with "Header Compression frame as a response".  On page 543 line 47, replace the MAC frame column value for FR48.11 "Header Compression Request frame" with "Header Compression frame". | Accepted |
| 10015 | Malinen, Jouni | 555.04 | Using capital Request/Response with "Header Compression" seems to imply that there is a frame or at least a term "Header Compression Request" and "Header Compression Response" defined somewhere in P802.11ah. No such definition exists, though. Maybe these should be referring to use of Header Compression frame as a request and as a response. | On page 555 line 4 (S1GM16.4), replace "Store the optional fields indicated in the Header Compression Request" with "Store the optional fields indicated in the Header Compression frame request".  On page 555 line 6 (S1GM16.6), replace "Send back the Header Compression Response" with "Send back the Header Compression frame response". On page 574 line 44, replace "This attribute indicates the amount of time, in units of milliseconds, the STA waits before timing out a Header Compression Request." with "This attribute indicates the amount of time, in units of milliseconds, the STA waits before timing out a Header Compression frame request." | Accepted |
| 10016 | Malinen, Jouni | 76.50 | The description of a Frame Control field in S1G PPDU (Figure 9-3a) is quite confusing when this is in a generic clause called 9.2.4.1 Frame Control field). This is not really generic anymore since this applies only for PV0 frames. The text in 9.2.3 seems to try to make it clear that 9.2.4 covers PV0, but that was not exactly obvious.. Having something more explicit within 9.2.4 would hopefully make this clearer. | Replace "Frame Control field" with "Frame Control field in PV0 frames" as the title of 9.2.4.1. Number of other clarifications within 9.2.4 would also be acceptable way of addressing this comment. | Revised –  Agree in principle with the commenter. The proposed resolution accounts for the suggested change targeting clarifications in 9.2.4.1.1.  TGah editor to make the changes shown in 11-16/0539r0 under all headings that include CID 10016. |
| 10017 | Malinen, Jouni | 385.33 | The CCMP AAD construction for PV1 MPDUs (Figure 12-18a) is designed to protect what is included in the header compressed version (i.e., what is sent over air) of PV1 frames, not the actual fields this frame and header caching state maps to. This is problematic especially with the design that does not protect the frames used to update the header compression state. In fact, this seems to result in critical security flaws that would allow an attacker to replace at least A3 and A4 field values with arbitrary information in MPDUs that get accepted by the receiver even when using RSN with management frame protection. This is obviously not desirable functionality.  It could be more robust to modify the CCMP AAD construction to protect the fields in the uncompressed version of the frame (i.e., full A1, A2, A3, A4) instead of the compressed version sent over air. This would allow the recipient to detect if something unexpected has been injected into the header compression state and only MPDUs with the address field values meant by the authorized transmitted would be accepted. | Modify CCMP AAD construction (page 385 lines 30 through page 386 line 32) by replacing the inclusion of SID field in A1/A2 with the 6-octet A1/A2 that this maps to. Clarify meaning of "MPDU Address 3 field, if present" and "MPDU Address 4 field, if present" to apply to the uncompressed (post-header decompression process on the receiver) version of the frame, not on the frame that is sent over air where these A3/A4 fields may not be present even though they exist in the uncompressed version that gets actually processed by higher layers in the stack. | Revised –  Agree in principle with the commenter. The proposed resolution accounts for the suggested changes.  TGah editor to make the changes shown in 11-16/0539r0 under all headings that include CID 10017. |

**Discussions: *None.***

**TGah Editor: *Replace “S1G Action” with “Unprotected S1G Action” throughout the draft (#10011):***

**TGah Editor: *Replace “S1G Relay Action” with “S1G Action” throughout the draft (#10011):***

**TGah Editor: *Change the table below as follows (#10011):***

|  |  |  |
| --- | --- | --- |
| * Unprotected S1G Action field values | | |
| Value | Meaning | Time Priority |
| 0 | AID Switch Request | No |
| 1 | AID Switch Response | No |
| 2 | Sync Control | No |
| 3 | STA Information Announcement | No |
| 4 | EDCA Parameter Set | No |
| 5 | EL Operation | No |
| 6 | TWT Setup | No |
| 7 | TWT Teardown | No |
| 8 | Sectorized Group ID List | No |
| 9 | Sector ID Feedback | No |
| 10 | Reserved | No |
| 11 | TWT Information | No |
| 12– 255 | Reserved |  |

**TGah Editor: *Change the table below as follows (#10011):***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * Category values | | | | |
| Code | Meaning | See subclause | Robust | Group addressed privacy |
| 22 | Unprotected S1G | 9.6.25 (Unprotected S1G Action frame details) | No | No |
| 23 | S1G | 9.6.26 (S1G Action frame details) | Yes | No |
| 24 | Flow control | 9.6.27 (Flow Control Action frame details) | Yes | No |
| 25 | Control Response MCS Negotiation | 9.6.28 (Control Response MCS Negotiation frame details) | Yes | No |

**TGah Editor: *Change the subclauses below (including headers) as follows (#10011):***

* S1G Action frame details
* S1G Action field

The S1G Action field values are specified in Table 9-417o (S1G Action field values).

|  |  |
| --- | --- |
| * S1G Action field values | |
| S1G Action field value | Description |
| 0 | Reachable Address Update |
| 1 | Relay Activation Request |
| 2 | Relay Activation Response |
| 3 | Header Compression Update |
| 4-255 | Reserved |

**TGah Editor: *Change the subclause below as follows (#10011):***

* Reachable Address Update frame format

The Reachable Address Update frame is used to update the addresses that can be reached through a relay STA. The format of the Reachable Address Update frame Action field is shown in Table 9-417p (Reachable Address Update frame Action field format).

|  |  |
| --- | --- |
| * Reachable Address Update frame Action field format | |
| Order | Information |
| 1 | Category |
| 2 | S1G Action |
| 3 | Reachable Address |

The Category field is defined in 9.4.1.11 (Action field).

The S1G Action field is defined in 9.6.26.1 (S1G Action field).

…

**TGah Editor: *Change the subclause below as follows (#10011):***

* Relay Activation Request frame format

The Relay Activation Request frame is used by the STA or AP to Request start or terminate a relay function. The format of the Relay Activation Request frame Action field is shown in Table 9-417q (Relay Activation Request frame).

|  |  |
| --- | --- |
| * Relay Activation Request frame | |
| Order | Information |
| 1 | Category |
| 2 | S1GAction |
| 3 | Relay Activation |

The Category field is defined in 9.4.1.11 (Action field).

The S1G Action field is defined in 9.6.26.1 (S1G Action field).

…

**TGah Editor: *Change the subclause below as follows (#10011):***

* Relay Activation Response frame format

The Relay Activation Response frame is used by the STA or AP to confirm or reject the start or the termination of the relay function. The format of the Relay Activation Response frame Action field is shown in Table 9-417r (Relay Activation Response frame).

|  |  |
| --- | --- |
| * Relay Activation Response frame | |
| Order | Information |
| 1 | Category |
| 2 | S1G Action |
| 3 | Relay Activation |

The Category field is defined in 9.4.1.11 (Action field).

The Relay Action field is defined in 9.6.26.1 (S1G Action field).

…

**TGah Editor: *Change the subclause below (including header, as a result move it to the respective new subclause) as follows (#10011):***

9.6.26.5 Header Compression frame format

The Header Compression frame is used to update information at the recipient STA, as defined in 10.56 (Generation of PV1 MPDUs and header compression procedure). The Header Compression frame contains the information shown in Table 9-417m (Header Compression action field format).

|  |  |
| --- | --- |
| * Header Compression action field format | |
| Order | Information |
| 1 | Category |
| 2 | S1G Action |
| 3 | Dialog Token |
| 4 | Header Compression (see 9.4.2.210 (Header Compression element)) |

The Category field is defined in 9.4.1.11 (Action field).

The S1G Action field is defined in 9.6.25.1 (S1G Action field).

…

**TGah Editor: *Change the subclause below as follows (#10012):***

10.56 Generation of PV1 MPDUs and header compression procedure

An S1G STA with dot11PV1MACHeaderOptionImplemented equal to true may include a Header Compression element in Header Compression frames. The STA may include the Header Compression element in (Re) Association Request frames and (Re) Association Response frames when management frame protection is not negotiated for the association. The STA may set the PV1 Data Type 3 Supported subfield in the Header Compression element to 1 to indicate that it supports reception of PV1 frames that have the Type subfield in the Frame Control field equal to 3.

**9.2.4.1 Frame Control field**

**9.2.4.1.1 General**

**TGah Editor: *Change the paragraph below as follows (#10016):***

The first three subfields of the Frame Control field of a PV0 frame are Protocol Version, Type, and Subtype. The remaining subfields of the Frame Control field depend on the setting of the Type and Subtype subfields. The Control frames carried by S1G PPDUs are called S1G Control frames.

**12.5.3.3.3 Construct AAD**

**TGah Editor: *Change the paragraphs below as follows (#10017):***

For PV1 MPDUs, the format of the AAD is shown in Figure 12-18a (AAD construction for PV1 MPDUs).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | FC | A1 | A2 | SC | A3 | A4 |
| Octets: | 2 | 6 | 6 | 2 | 0 or 6 | 0 or 6 |
| * AAD construction for PV1 MPDUs | | | | | | |

For PV1 MPDUs, the length of the AAD varies depending on the presence or absence of the A3 and A4 fields and is shown in Table 12-1a (AAD length for PV1 MPDUs).

|  |  |  |  |
| --- | --- | --- | --- |
| * AAD length for PV1 MPDUs | | | |
| Type subfield in the Frame Control | A3 field in uncompressed header | A4 field in uncompressed header | AAD length (octets) |
| 0, 1 or 3 | Absent | Absent | 16 |
| 0, 1 or 3 | Present | Absent | 22 |
| 0 or 3 | Absent | Present | 22 |
| 0 or 3 | Present | Present | 28 |

For PV1 MPDUs, AAD construction is performed as follows:

* FC – MPDU Frame Control field, with
* Power Management subfield (bit 10) masked to 0
* More Data subfield (bit 11) masked to 0
* Protected Frame subfield (bit 12) always set to 1
* EOSP subfield (bit 13) masked to 0
* Relayed Frame subfield (bit 14) masked to 0
* Ack Policy subfield (bit 15) masked to 0
* A1 –MPDU Address 1 field if it contains a MAC address; otherwise the MAC address that corresponds to the AID value contained in the SID field of the A1 field.
* A2 –MPDU Address 2 field if it contains a MAC address; otherwise the MAC address corresponding to the AID value contained in the SID field of the A2 field.
* A3 –MPDU Address 3 field if present in the MPDU, the value of A3 stored at the receiver if A3 is stored at the receiver and is not present in the MPDU (see (10.56 Generation of PV1 MPDUs and header compression procedure)); otherwise not present.
* A4 –MPDU Address 4 field, if present in the MPDU, the value of A4 stored at the receiver if A4 is stored at the receiver and is not present in the MPDU (see (10.56 Generation of PV1 MPDUs and header compression procedure)); otherwise not present.
* SC – MPDU Sequence Control field, with the Sequence Number subfield (bits 4–15 of the Sequence Control field) masked to 0. The Fragment Number subfield is not modified.

**12.5.3.3.6 CCM originator processing**

**TGah Editor: *Change the paragraph below as follows (#10017):***

There are four inputs to CCM originator processing:

a) Key: the temporal key (16 octets).

b) Nonce: the nonce (13 octets) constructed as described in 12.5.3.3.4 (Construct CCM nonce).

c) Frame body: the plaintext frame body of the MPDU.

d) AAD: the AAD (~~2~~16–30 octets) constructed from the MPDU header as described in 12.5.3.3.3 (Construct AAD).

**TGah Editor: *Change the subclause below as follows (#10013):***

* CCMP test vectors

CCMP PV1 test vectors

BSSID: a2:ae:a5:b8:fc:ba

DA: 02:d2:e1:28:a5:7c

SA: 52:30:f1:84:44:08

Association ID: 7

Base PN: 123 (0x0000007b)

SC = 0x3380 (FragNum=0 SeqNum=824)

TID = 3

Key ID: 0

TK - hexdump(len=16): c9 7c 1f 67 ce 37 11 85 51 4a 8a 19 f2 bd d5 2f

PN = SC||BPN

PN (PN0..PN5) - hexdump(len=8): 80 33 7b 00 00 00 00 00

PV1 test vector #1:

Header compression used and A3 was previously stored at the receiver

FC=0x0061 (PV=1 Type=0 PTID/Subtype=3 From\_DS=0 More\_Fragments=0 Power\_Management=0 More\_Data=0 Protected\_Frame=0 End\_of\_SP=0 Relayed\_Frame=0 Ack\_Policy=0)

A1=a2:ae:a5:b8:fc:ba

A2=07 00 (SID: AID=7 A3\_Present=0 A4\_Present=0 A-MSDU=0); corresponds to 52:30:f1:84:44:08 in uncompressed header

Sequence Control: 80 33 (FN=0 SN=824)

A3 not present; corresponds to 02:d2:e1:28:a5:7c in uncompressed header

A4 not present

Plaintext Frame Header - hexdump(len=12): 61 00 a2 ae a5 b8 fc ba 07 00 80 33

Plaintext Frame Body - hexdump(len=20): f8 ba 1a 55 d0 2f 85 ae 96 7b b6 2f b6 cd a8 eb 7e 78 a0 50

CCMP AAD - hexdump(len=22): 61 10 a2 ae a5 b8 fc ba 52 30 f1 84 44 08 00 00 02 d2 e1 28 a5 7c

CCMP nonce - hexdump(len=13): 20 52 30 f1 84 44 08 00 00 00 7b 33 80

CCM B\_0 - hexdump(len=16): 59 20 52 30 f1 84 44 08 00 00 00 7b 33 80 00 14

CCM T - hexdump(len=8): 50 be 59 0a 6a 05 a2 8a

CCM U - hexdump(len=8): 82 62 ff 2d b5 77 65 73

CCMP encrypted - hexdump(len=20): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45

Encrypted Frame Header - hexdump(len=12): 61 10 a2 ae a5 b8 fc ba 07 00 80 33

Encrypted Frame Frame Body - hexdump(len=28): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45 82 62 ff 2d b5 77 65 73

Encrypted Frame FCS - hexdump(len=4): 97 3e b8 a7

PV1 test vector #2:

Header compression used and A3 was not previously stored at the receiver

FC=0x0061 (PV=1 Type=0 PTID/Subtype=3 From\_DS=0 More\_Fragments=0 Power\_Management=0 More\_Data=0 Protected\_Frame=0 End\_of\_SP=0 Relayed\_Frame=0 Ack\_Policy=0)

A1=a2:ae:a5:b8:fc:ba

A2=07 20 (SID: AID=7 A3\_Present=1 A4\_Present=0 A-MSDU=0) ; corresponds to 52:30:f1:84:44:08 in uncompressed header

Sequence Control: 80 33 (FN=0 SN=824)

A3=02:d2:e1:28:a5:7c

A4 not present

Plaintext Frame Header - hexdump(len=18): 61 00 a2 ae a5 b8 fc ba 07 20 80 33 02 d2 e1 28 a5 7c

Plaintext Frame Body - hexdump(len=20): f8 ba 1a 55 d0 2f 85 ae 96 7b b6 2f b6 cd a8 eb 7e 78 a0 50

CCMP AAD - hexdump(len=22): 61 10 a2 ae a5 b8 fc ba 07 20 80 33 02 d2 e1 28 a5 7c

CCMP nonce - hexdump(len=13): 20 52 30 f1 84 44 08 00 00 00 7b 33 80

CCM B\_0 - hexdump(len=16): 59 20 52 30 f1 84 44 08 00 00 00 7b 33 80 00 14

CCM T - hexdump(len=8): 50 be 59 0a 6a 05 a2 8a

CCM U - hexdump(len=8): 82 62 ff 2d b5 77 65 73

CCMP encrypted - hexdump(len=20): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45

Encrypted Frame Header - hexdump(len=18): 61 10 a2 ae a5 b8 fc ba 07 20 80 33 02 d2 e1 28 a5 7c

Encrypted Frame Frame Body - hexdump(len=28): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45 82 62 ff 2d b5 77 65 73

Encrypted Frame FCS - hexdump(len=4): a3 04 e8 51

PV1 test vector #3:

Type 3 frame from SA to DA(=BSSID) (i.e., no separate DA in this example)

FC=0x006d (PV=1 Type=3 PTID/Subtype=3 From\_DS=0 More\_Fragments=0 Power\_Management=0 More\_Data=0 Protected\_Frame=0 End\_of\_SP=0 Relayed\_Frame=0 Ack\_Policy=0)

A1=a2:ae:a5:b8:fc:ba

A2=52:30:f1:84:44:08

Sequence Control: 80 33 (FN=0 SN=824)

A3 not present ; corresponds to 02:d2:e1:28:a5:7c in uncompressed header

A4 not present

Plaintext Frame Header - hexdump(len=16): 6d 00 a2 ae a5 b8 fc ba 52 30 f1 84 44 08 80 33

Plaintext Frame Body - hexdump(len=20): f8 ba 1a 55 d0 2f 85 ae 96 7b b6 2f b6 cd a8 eb 7e 78 a0 50

CCMP AAD - hexdump(len=22): 6d 10 a2 ae a5 b8 fc ba 52 30 f1 84 44 08 00 00 02 d2 e1 28 a5 7c

CCMP nonce - hexdump(len=13): 20 52 30 f1 84 44 08 00 00 00 7b 33 80

CCM B\_0 - hexdump(len=16): 59 20 52 30 f1 84 44 08 00 00 00 7b 33 80 00 14

CCM T - hexdump(len=8): cd ef 4a 4f 36 3b bb 26

CCM U - hexdump(len=8): 1f 33 ec 68 e9 49 7c df

CCMP encrypted - hexdump(len=20): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45

Encrypted Frame Header - hexdump(len=16): 6d 10 a2 ae a5 b8 fc ba 52 30 f1 84 44 08 80 33

Encrypted Frame Frame Body - hexdump(len=28): dd d7 40 e2 a5 86 e1 2b 06 0e 45 69 d0 a3 93 61 60 41 2e 45 1f 33 ec 68 e9 49 7c df

Encrypted Frame FCS - hexdump(len=4): aa df 86 de