IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolutions for Block Ack Related Comments | | | | |
| Date: 2018-04 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Graham SMITH | SR Technology | Davie, FL, USA. | 916 799 9563 | gsmith@srtrl.com |

Abstract

This submission proposes resolutions for CIDs 1391, 1392, 1313, 1308

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.

**NOTE: CID 1238 from Menzo, may affect these resolutions**

**“A couple items need to be cleaned up as part of the deletion of the basic block ack protocol.**

**A submission will be prepared.”**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CID | Commenter | Clause | Page | Line | Comment | Proposed |
| 1391 | Mark Rison | 10.25.3 | 1716 | 54 | It is not clear what "sent under block ack policy" means | Change the cited text at the cited location to "send under a BA agreement" |

**10.25.3 Teardown of the block ack mechanism**

When the originator has no data to send and the final block ack exchange has completed, it shall signal the end

of its use of the block ack mechanism by sending the DELBA frame to its recipient. (11ah)The DELBA frame

sent by the S1G originator shall be a BAT DELBA if a BAT ADDBA Request was sent during block ack

setup or NDP DELBA if an NDP ADDBA Request was sent during block ack setup or DELBA if ADDBA

Request was sent during block ack setup. The recipient does not generate a Management frame in response to

the DELBA frame.40 The recipient of the DELBA frame shall release all resources allocated for the block ack

transfer.

*The block ack agreement may be torn down if there are no BlockAck, BlockAckReq, or (#57)MPDUs* ***(sent***

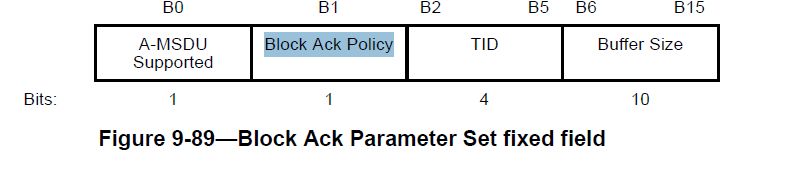
***under block ack policy)*** *for the block ack’s TID received from the peer within a duration of block ack timeout*

*value (see 11.5.4 (Error recovery upon a peer failure)).*

**Discussion**

This section is about teardown of a block ack. The cited text refers to the absence of MPDUs being sent under a block ack agreement ftime greater than the block out timeout.

“Block ack policy” is set by the B1 in the Block Ack Parameter Set fixed field and is either Immediate Block Ack or Delayed Block Ack.



The TID subfield contains the value of the TC or TS for which the BlockAck frame is being requested.

The Block Ack Parameter Set fixed field is sent in ADDBA Request and Response frames.

So is “MPDUs **(sent under block ack policy)** for the block ack’s TID received from the peer…” accurate/correct?

It is correct in that the MPDU is sent with a block ack policy and a related TID but the text and the use of the brackets, does appear (to me) to be somewhat awkward and not accurately conveying the intent.

I doubt if two BA agreements would be set up for one TID with both BlockAck policies, so just referring to the BA Agrrement for the TID should suffice.

RESOLUTION

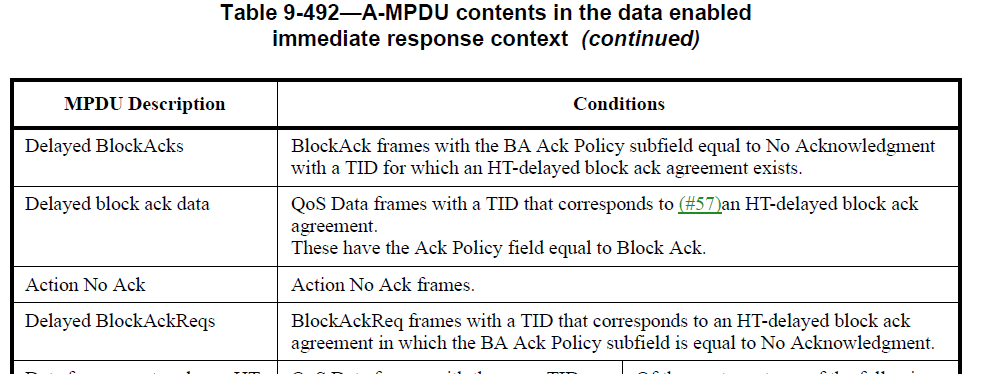
REVISE

At 1716.55 make change as follows:

The block ack agreement may be torn down if there are no BlockAck, BlockAckReq, or (#57)MPDUs sent

under the BA agreement for the block ack’s TID received from the peer within a duration of block ack timeout value (see 11.5.4 (Error recovery upon a peer failure)).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CID | Commenter | Clause | Page | Line | Comment | Proposed |
| 1392 | Mark Rison | 9.7.3 | 1536 | 15 | "BlockAckReq frames with a TID that corresponds to an HT-delayed block ack agreement in which the BA Ack Policy subfield is equal to No Acknowledgment" -- an agreement does not have a subfield, the precendence not clear, and should be BAR Ack Policy subfield not BA Ack Policy | Change the cited text at the referenced location to "BlockAckReq frames with a TID that corresponds to an HT-delayed block ack  agreement, and in which the BAR Ack Policy subfield is equal to No Acknowledgment" |



First let’s look at 2056.48

“NOTE—HT-delayed block ack agreement is obsolete. Support for this mechanism might be removed in a later

revision of the standard.”

CID 61 and 17/1137r10 removed Delayed Block Ack but kept HT-delayed block ack. I suspect that this NOTE should be deleted.

OK back to the comment

Cite

“BlockAckReq frames with a TID that corresponds to an HT-delayed block ack

agreement in which the BA Ack Policy subfield is equal to No Acknowledgment.”

Comment

an agreement does not have a subfield - TRUE

the precendence not clear - not convinced

and should be BAR Ack Policy subfield not BA Ack Policy – TRUE a BAR frame has a BAR Ack policy

Proposed

"BlockAckReq frames with a TID that corresponds to an HT-delayed block ack agreement, and in which the BAR Ack Policy subfield is equal to No Acknowledgment"

I have no problem with the Proposed.

RESOLUTION

ACCEPT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CID | Commenter | Clause | Page | Line | Comment | Proposed |
| 1313 | Li-Hsiang Sun | 10.25.5.8 | 1726 | 25 | Not clear why the situation described in the NOTE can happen: "It is possible for the Starting Sequence Number subfield value (SSN) of the received BlockAck frame to be greater than WinStartO" In the received BA frame, based on the requirement in p1722.11  1. SSN<=WinStart\_R The receiver bases on the successfully received MPDU with the highest SN x to determine WinStart\_R (10.25.5.4 step b): 2. winStart\_R= x - WinSize+1 3. x<=WinStart\_O+WinSize-1 combines 1,2,3  SSN<=WinStart\_R<=WinStart\_O So it does not seem possible SSN > winStart\_O  The originator can send a BAR with SSN>winStart\_O to initialize the recipient's BA record bitmap, but this is not the case described by the NOTE | clarify the NOTE |

1726.19

If the originator receives a BlockAck frame in response to an HT-immediate BlockAckReq frame, it shall, in addition,

— Not update the status of MPDUs with Sequence Number subfield values between *WinStartO* and *SSN* of the received BlockAck frame, and

NOTE—It is possible for the Starting Sequence Number subfield value (*SSN*) of the received BlockAck frame to be greater than *WinStartO* because of the failed reception of a nonzero number of MPDUs beginning with the MPDU with Sequence Number subfield value equal to *WinStartO* at a recipient that is using partial-state operation.

The HT-Immediate Block Ack scheme is as follows:

Originator sends ADDBA Request containing the Sequence Number (SN) of the first MSDU sent under this agreement.

WinStart R = SSN (starting sequence number) from the ADDBA request frameSN

Originator transmits with WinStart 0 as the starting SN of the transmit window.

Note that the NOTE is saying that this is possible as the initial MPDU(s) in the transmission failed in the case of **“partial-state operation”.**

10.25.5.4 (Partial-State)

Recipient maintains the temporary block ack record (This seems to state just one record)

* Bitmap, indexed by SN
* WinStart R (lowest SN in bitmap)
* WinEnd R (Highest SN in bitmap)

1. During partial-state operation, *WinStartR* is determined by the Sequence Number subfield value of received Data frames and by the Starting Sequence Number subfield value of received BlockAckReq frames as described below.

At b) for each received Data frame, create a temporary block ack acknowledgmwent record:

* WinEnd R = SN (SN of the received Data frame)
* WinStart R = WinEnd R – WinSize R +1
* Create Bit map, size WinSize R, first bit corresponds to SN WinStart R, last bit is SN WinEnd R, set all bits to 0.
* Set to 1 the bit in position in the bitmap that corresponds to SN.

Now comes the possible confusion, at d) we read

At d) for each received BlockAckReq frame, create a temporary block acknowledgment record:

* WinEnd R = SNN (starting SN in BlockAckReq)
* WinStart R = WinEnd R – WinSize R +1
* Create a temporary record Block Ack record Bit map, size
  + WinSize R, set all bits to 0.

Remember WinStart 0 is the starting sequence number of the transmit window (from the originator). Is this the same as the SSN in the BlockAckReq?

So, on the face of it we seem to have 2 possible bit maps, either based upon received Data or received BlockAckReq. SO if the recipient choses to use the Data received version, then it is possible that the SSN is higher than WinStart 0. If the recipient choses to use the BlockAckReq version, then the the SSN will be WinStart 0.

So, if I have interpreted this right, the NOTE is correct in that it may happen that the SSN is higher than WinStart 0 in the case that the recipient has chosen to use a bit map based upon the received Data. Note that the received Data would occur after the received BlockAckReq so maybe the idea is that the bit map starts off with the BAR version and hten is changed to the Data version as soon as Data is actually received.

(We could now look at ‘full-state operation’ to check the differences, but if interested read 10.25.5.3. but it definitely starts with SSN)

Having said all that, it would be good to hear from a BA implementer that my interpretation is correct. Also, if I am not right, maybe it should be clear - SSN, WinStart R and WinStart 0 is pretty confusing.

My inclination is to reject, but I would like an expert opinion.

RESOLUTION

REJECT

Using partial-state operation it is possible that the SSN is greater than WinStart 0 as the recipient may be using a bit map based upon the received Data frames.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CID | Commenter | Clause | Page | Line | Comment | Proposed |
| 1308 | Allen Heberling | 10.25.2 | 1716 | 39 | Editor's Note regarding which reference to use to describe the procedure for when transferring Data and ACK frames. | Please provide the appropriate reference that describes the procedure for when Data and ACK frames are transferred. |

“Once the block ack exchange has been set up, Data and Ack frames are transferred using the procedure

described in 10.25.3(#57).

***Editor’s Note: Given clause 10.25.3 is removed, what is the appropriate reference for the procedure that Data and Ack frames are transferred?”***

OK this is an editor error, but my fault for not checking (sorry)

17/1137contained resolutions for Obsolete Block Acks.

1524.22 edit as shown

**10.24.3 Data and acknowledgement transfer using immediate block ack policy and delayed block ack policy**

“After setting up an immediate block ack agreement following the

procedure in 10.24.2 (Setup and modification of the block ack parameters), and having gained access to the

medium and established protection, if necessary, the originator may transmit an A-MPDU

The RA field of the frames that are not delivered using the GCR block ack retransmission policy shall be

the recipient’s individual address. The RA field of GCR frames delivered using the GCR block ack

retransmission policy shall be set to the GCR concealment address. The originator requests acknowledgment of

outstanding QoS Data frames by sending a BlockAckReq frame. ”

1524.37 to 1527.12 delete entirely

**Note that this does not remove all of 10.24.3. The above text for 10.24.3 should have remained**

1527.13 delete clause 10.24.4 in its entirety. This was done (maybe a confusion and 10.24.3 was also deleted?)

Putting this into changes for D1.0 we need the following

RESOLUTION

REVISED

Insert new clause 10.25.3

**10.25.3 Data and acknowledgement transfer using immediate block ack policy and delayed block ack policy**

“After setting up an immediate block ack agreement following the procedure in 10.25.2 (Setup and modification of the block ack parameters), and having gained access to the medium and established protection, if necessary, the originator may transmit an A-MPDU. The RA field of the frames that are not delivered using the GCR block ack retransmission policy shall be the recipient’s individual address. The RA field of GCR frames delivered using the GCR block ack retransmission policy shall be set to the GCR concealment address. The originator requests acknowledgment of outstanding QoS Data frames by sending a BlockAckReq frame.”