IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CID 1505 | | | | |
| Date: September 6, 2018 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Menzo Wentink | Qualcomm | Utrecht, The Netherlands | +31-65-183-6231 | mwentink  @qualcomm.com |

Abstract

This document proposes a resolution for CID 1505.

**Comment**

|  |  |  |
| --- | --- | --- |
| CID 1505  Mark Rison  10.3.3 | There are several issues with the [QS]SRC/LRC stuff:  sometimes it's per-MPDU (p.1290 in mc/D6.0),  sometimes it's per-MSDU/MMPDU (p.1295);  sometimes it's Data frames only (p.1290),  sometimes Management too (p.1295) | Frankly, I can't work it out. Tell me whether SRC/LRC is per-MPDU or per-MSDU/MMPDU, and whether it includes Managament frames/MMPDUs, and I'll come up with the changes |

**Discussion**

**EDCA**

The current EDCA text related to retry counters is cited below, dissected and reordered. Retry counters and the associated rules are marked bold.

**10.23.2.12 Retransmit procedures**

**10.23.2.12.1 General**

A QoS STA shall maintain a **short retry counter** and a **long retry counter** for each MSDU, A-MSDU, or MMPDU that belongs to a TC that requires acknowledgment. The initial value for the **short and long retry counters shall be 0**.

QoS STAs also maintain a short retry counter and a long retry counter for each AC. They are defined as **QSRC[AC]** and **QLRC[AC]**, respectively, and **each is initialized to a value of 0**.

*[The following are the rules for the short retry count and QSRC]*

The **short retry count** for an MSDU or A-MSDU that is not part of a block ack agreement or for an MMPDU **shall be incremented** every time transmission of a frame in a PSDU of length **less than or equal to dot11RTSThreshold** fails for that MSDU, A-MSDU, or MMPDU.

After an RTS frame is transmitted to protect an MSDU or MMPDU, a QoS STA performs the CTS procedure, as defined in 10.3.2.8 (CTS and DMG CTS procedure).

If a valid CTS frame is received, the **QSRC[AC]** for the corresponding AC **shall be reset to 0**.

This **short retry count** and theQoS STA **QSRC[AC] shall be reset** when an A-MPDU or frame of length in a PSDU less than or equal to dot11RTSThreshold succeeds.

If a valid CTS frame is not received, the **short retry counter** for the MSDU or MMPDU and the **QSRC[AC]** for the corresponding AC **shall be incremented**.

**QSRC[AC] shall be incremented** every time transmission of a frame in a PSDU of length less than or equal to dot11RTSThreshold fails, regardless of the presence or value of the DEI field.

For internal collisions occurring with the EDCA access method, the appropriate retry counters (**short retry counter** for MSDU, A-MSDU, or MMPDU and **QSRC[AC]** or **long retry counter** for MSDU, A-MSDU, or MMPDU and **QLRC[AC]**) **are incremented**.

*[The following are the rules for the long retry count and QLRC]*

The **long retry count** for an MSDU or A-MSDU that is not part of a block ack agreement or for an MMPDU **shall be incremented** every time transmission of a MAC frame in a PSDU of length **greater than dot11RTSThreshold** fails for that MSDU, A-MSDU, or MMPDU.

This **long retry count** and the **QLRC[AC] shall be reset** when an A-MPDU or frame in a PSDU of length greater than dot11RTSThreshold succeeds.

**QLRC[AC] shall be incremented** every time transmission of an A-MPDU or frame in a PSDU of length greater than dot11RTSThreshold fails, regardless of the presence or value of the DEI field.

For internal collisions occurring with the EDCA access method, the appropriate retry counters (**short retry counter** for MSDU, A-MSDU, or MMPDU and **QSRC[AC]** or **long retry counter** for MSDU, A-MSDU, or MMPDU and **QLRC[AC]**) **are incremented**.

**10.23.2.2 EDCA backoff procedure**

If the backoff procedure is invoked for reason c), d), e), or f) above, or the transmission failure of a non-initial frame by the TXOP holder, the value of CW[AC] shall be updated as follows before invoking the backoff procedure:

— If the **QSRC[AC]** or the **QLRC[AC]** **has reached** **dot11ShortRetryLimit** or **dot11LongRetryLimit** respectively, CW[AC] shall be reset to CWmin[AC].

Otherwise,

— If CW[AC] is less than CWmax[AC], CW[AC] shall be set to the value (CW[AC] + 1) × 2 – 1.

— Else, CW[AC] shall be set to CWmax[AC].

**10.23.2.12 Retransmit procedures**

**10.23.2.12.1 General**

Retries for failed transmission attempts shall continue until one or more of the following conditions occurs:

— The **short retry count** for the MSDU, A-MSDU, or MMPDU is equal to **dot11ShortRetryLimit**.

— The **long retry count** for the MSDU, A-MSDU, or MMPDU is equal to **dot11LongRetryLimit**.

When any of these limits is reached, retry attempts shall cease, and the MSDU, A-MSDU, or MMPDU shall be discarded.

Some observations on these rules:

The short retry count is updated for short packets, like short MSDUs and RTS.

The long retry count is updated for long packets, which are MSDUs longer than dot11RTSThreshold.

A STA maintains retry counts per packet (short retry count / long retry count) and for the STA overall (QSRC/QLRC).

For a long packet, RTSs are transmitted until a CTS is received. The short retry count and QSRC are updated during that process. When a CTS is received but the long packet fails, the long retry count and QLRC are updated.

Therefore, the long retry count and QLRC increase only when an RTS/CTS exchange already succeeded and a collision was therefore not detected. So it seems strange that the long retry count (QLRC) ties into the backoff procedure for resetting the CW. The failing of the (long) MPDU itself may have more to do with MCS selection.

There appears to be little use for having two retry counters.

Also, the "has reached" language in 10.23.2.2 (EDCA backoff procedure) suggests that the CW keeps resetting once a retry count is equal to or larger than its limit. In an alternative interpretation, "has reached" means "is equal to", in which case the CW resets only once. This is kind of unclear and probably neither behavior was the intended one:

— If the QSRC[AC] or the QLRC[AC] **has reached** dot11ShortRetryLimit or dot11LongRetryLimit respectively, CW[AC] shall be reset to CWmin[AC].

The station counters QSRC and QLRC reset only after a succesful transmission, not after a packet discard. So while packets are being discarded, QSRC and QLRC keep counting. There does not appear to be another reset of the QSRC and the QLRC except for a successful transmission.

A possible way forward might be to replace the dual retry counter structure with a single backoff stage that governs the CW and a single retry count that governs packet discard.

Legacy implementations with the dual retry counter structure are assumed to be still be compliant under these simplified rules, also because the retry limits are configurable by a STA and an AP, so there is not one single behavior described by the current rules.

**Proposed Resolution**

The proposed resolution removes the long retry counter and the QLRC. Transmission failures are now counted in a single retry counter per packet and a QSRC per AC. There are three possible failure cases: 1) on an RTS, 2) on the packet after a CTS and 3) on a packet without prior RTS/CTS. Each failure case may cause the CW to be incremented but failure case 2 is optional because reason f) for calling the backoff procedure is optional.

The backoff procedure is rewritten for clarity and updated to allow the CW to be reset to CWmin each time QLRC reaches dot11ShortRetryLimit. This behavior appears to have been intended by the original backoff procedure, but was not clear due to the "has reached" language and the apparent omission of a reset of the QSRC and QLRC when a packet was discarded.

Changes shown are relative to REVmd draft 1.4.

***Modify as shown.***

**3. Definitions, acronyms, and abbreviations**

211.4 delete acronym QLRC

**Annex C**

Deprecate MIB variable dot11LongRetryLimit per the stock editing instructions for deprecating a MIB variable.

**10.24.2.2 EDCA backoff procedure**

1775.44

If the backoff procedure is invoked for reason c), d), e), or f) above, the value of CW[AC] shall be updated as follows before invoking the backoff procedure:

— For each AC (omitting [AC] for QSRC, CW, CWmin, and CWmax):

— If QSRC is less than dot11ShortRetryLimit,

* QSRC shall be increased by 1
* CW shall be set to the lesser of CWmax and 2^QSRC × (CWmin + 1) - 1

— Else

* QSRC shall be set to 0
* CW shall be set to CWmin

— If dot11RobustAVStreamingImplemented is true and either the QSDRC[AC] or the QLDRC[AC] has reached dot11ShortDEIRetryLimit or dot11LongDEIRetryLimit, respectively, CW[AC] shall be reset to CWmin[AC].

**10.24.2.12 Retransmit procedures**

**10.24.2.12.1 General**

1788.55

A QoS STA shall maintain a retry counter for each MSDU, A-MSDU, or MMPDU that belongs to a TC that requires acknowledgment. The initial value for the retry counter shall be 0. QoS STAs shall also maintain a retry counter for each AC, which is defined as QSRC[AC] and which shall be initialized to a value of 0.

When dot11RobustAVStreamingImplemented is true, a QoS STA shall maintain a short drop-eligible retry counter and a long drop-eligible retry counter for each AC. They are defined as QSDRC[AC] and QLDRC[AC], respectively, and each is initialized to a value of zero. APs with dot11RobustAVStreamingImplemented true and mesh STAs with dot11MeshGCRImplemented true, shall maintain an unsolicited retry counter.

After transmitting a frame that requires an immediate acknowledgment, the STA shall perform either of the acknowledgment procedures, as appropriate, that are defined in 10.3.2.11 (Acknowledgment procedure)(Ed#57). The retry counter for an MSDU or A-MSDU that is not part of a block ack agreement or for an MMPDU shall be incremented every time transmission fails for that MSDU, A-MSDU, or MMPDU, including of an associated RTS. The unsolicited retry counter shall be incremented after the transmission of every A-MSDU that is transmitted using the GCR unsolicited retry retransmission policy.

When dot11RobustAVStreamingImplemented is true, QSDRC[AC] shall be incremented every time a(#210) transmission of an A-MPDU or a frame in PSDU(#210) in which the HT variant HT Control field is present, the DEI field is equal to 1 and the length of the PSDU of length(#210) is less than or equal to dot11RTSThreshold fails. This short retry count shall be reset when an A-MPDU or frame of length in a PSDU less than or equal to dot11RTSThreshold succeeds. When dot11RobustAVStreamingImplemented is true, the QSDRC[AC] shall be reset when an A-MPDU or frame in a PSDU of length less than or equal to dot11RTSThreshold succeeds, regardless of the presence or value of the DEI field.

When dot11RobustAVStreamingImplemented is true, QLDRC[AC] shall be incremented every time transmission fails for an A-MPDU or frame in a PSDU of length greater than dot11RTSThreshold in which the HT variant HT Control field is present and the DEI field is equal to 1.

The QLDRC[AC] shall be reset when an A-MPDU or frame in a PSDU of length greater than dot11RTSThreshold succeeds, regardless of the presence or value of the DEI field.

All retransmission attempts by a non-DMG STA for an MPDU with the Type subfield equal to Data or Management that is not sent under a block ack agreement and that has failed the acknowledgment procedure one or more times shall be made with the Retry subfield set to 1. All retransmission attempts by a DMG STA for an MPDU with the Type subfield equal to Data or Management that has failed the acknowledgment procedure one or more times shall be made with the Retry subfield set to 1.

Retries for failed transmission attempts shall continue until one or more of the following conditions occurs:

— The retry count for the MSDU, A-MSDU, or MMPDU is equal to dot11ShortRetryLimit.

— The short drop-eligible retry count for the MSDU, A-MSDU, or MMPDU is equal to dot11ShortDEIRetryLimit.

— The long drop-eligible retry count for the MSDU, A-MSDU, or MMPDU is equal to dot11LongDEIRetryLimit.

— The unsolicited retry count for the A-MSDU is equal to dot11UnsolicitedRetryLimit.

When any of these limits is reached, retry attempts shall cease, and the MSDU, A-MSDU, or MMPDU shall be discarded.

For internal collisions occurring with the EDCA access method, the retry counter is incremented. For internal collisions occurring with the EDCA access method where dot11RobustAVStreamingImplemented is true, the appropriate drop-eligible retry counters (QSDRC[AC], and QLDRC[AC]) are incremented when the collision occurs for an MSDU, A-MSDU, or MMPDU that has drop eligibility equal to 1. (Ed#57)A STA shall retry failed transmissions until the transmission is successful or until the relevant retry limit is reached.

**10.24.5.5 EDCA backoff procedure in generic RAW or triggering frame RAW**

1807.36

Each STA performing EDCA access suspends an operation of its EDCAF at the start of each RAW and stores the value of the backoff counter, CW[AC], and QSRC[AC] as the first backoff state. At the end of the RAW, the stored first backoff function state is restored and an operation of the EDCAF is resumed. If the previously stored first backoff function state is empty, the EDCAF of a STA shall invoke a backoff procedure, even if no additional transmissions are currently queued.

**10.25.3.9 Access during MCCAOPs**

**10.25.3.9.1 Access by MCCAOP owners**

1820.16

At the start of the MCCAOP, the EDCAF of the MCCAOP owner shall set AIFSN[AC] equal to dot11MCCAAIFSN, CWmax[AC] equal to dot11MCCACWmax, CW[AC] equal to dot11MCCACWmin, and QSRC[AC] to 0 for all ACs. The TXOP limit shall specify a duration value no larger than the MCCAOP Duration.

During the MCCAOP, the EDCAFs of the ACs operates as specified in 10.24.2 (HCF contention based

channel access (EDCA)), with the following modifications.

— During the MCCAOP, the EDCAF of each AC shall consider only those frame whose RA matches the MAC address of the MCCAOP responder.

— When the access to the medium is delayed, the TXOP limit shall specify a duration to end no later than the MCCAOP start time plus the MCCAOP Duration.

— As specified in 10.25.3.9.2 (Access during an MCCAOP by mesh STAs that are not the MCCAOP owner), a neighboring STA shall not access the WM during an MCCAOP, until it receives a frame from either the MCCAOP owner or the MCCAOP responder. With the exception of truncation of an MCCA TXOP by means of a CF-End, standard EDCA TXOP rules apply for the remainder of the MCCAOP. For HT mesh STAs, these include the reverse direction protocol as specified in 10.30 (Reverse direction protocol).

— At the end of the MCCAOP, the parameters used by the EDCAF of the MCCAOP owner shall be set to dot11EDCATable, and QSRC[AC] shall be set to 0 for all ACs.