IEEE P802.11
Wireless LANs

|  |
| --- |
| 802.11bi – Spec text for A1 filtering |
| Date: November 13, 2023 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Antonio de la Oliva | Interdigital Ltd, UC3M |  |  | aoliva@it.uc3m.es |
| Joseph Levy | Interdigital Ltd, |  |  |  |

 Abstract

This submission proposes specification text modifying A1 filtering. The baseline for the specification is IEEE 802.11REVme/D4.0

Proposed contribution

10.2.8 MAC data service

The MAC data service provides the transport of MSDUs between MAC peer entities as characterized in 5.1.1 (Data service).

The transmission process is started by the MAC’s receipt of one or more MA-UNITDATA.request primitives, each containing an MSDU and its associated parameters. This might cause one or more Data frames, containing the MSDU(s), to be transmitted.

The reception process is started by the MAC’s receipt of one or more Data frames containing one or more MSDUs. This might cause one or more MA-UNITDATA.indication primitives, each containing an MSDU and its associated parameters, to be issued.

When dot11SSPNInterfaceActivated is true, an AP shall distribute the group addressed message into the BSS only if dot11NonAPStationAuthSourceMulticast in the dot11InterworkingEntry identified by the source MAC address in the received message is true. When dot11SSPNInterfaceActivated is false, an AP shall distribute the group addressed message into the BSS, except when dot11RelayAPOperationActivated is true and the group addressed message is received from a STA. In that case, the group addressed message shall not be distributed into the BSS, and it shall be forwarded to the S1G relay STA in the same relay. The S1G relay STA shall send the group addressed message to the associated AP as an individually addressed frame using either a four address frame format (PV0 or PV1) or an A-MSDU format as specified in 10.53.4 (Addressing and forwarding of group addressed relay frames).

Unless the MPDU is delivered via DMS, the STA originating the message receives the message as a group addressed message (prior to any filtering). Therefore, a STA shall filter out group addressed messages that contain their address as the source address; the point at which such filtering occurs in the processing of received frames is an implementation choice. When dot11SSPNInterfaceActivated is false, group addressed MSDUs shall be propagated throughout the ESS. When dot11SSPNInterfaceActivated is true, group addressed MSDUs shall be propagated throughout the ESS only if dot11NonAPStationAuthSourceMulticast in the dot11InterworkingEntry identified by the source MAC address in the received message is true.

A STA transmitting on a general link also uses the addressing rules described in 10.64 (Addressing of GLK Data frame transmission).

A MAC not contained within an S1G relay shall perform address filtering on the Address 1 field in each MPDU contained in a PPDU and, for non-GLK non-AP STAs, on the DA of each MSDU within an A-MSDU. In the case of a non-GLK STA receiver, when the Address 1 field or DA field contains a group address, address filtering is performed by comparing the value in the Address 1 field or DA field to all values in the dot11GroupAddressesTable and the broadcast address value. In case of an EP (Enhanced Privacy) STA, using Frame Anonymization (FA), address filtering is performed by comparing the value of the Address 1 field, or the DA field to all values in the dot11RxAddressesTable, dot11GroupAddressesTable and the broadcast value. The STA also validates the BSSID to verify that it either corresponds to the BSS of which the receiving STA is a member, or if dot11OCBActivated is true, that it contains the wildcard BSSID, indicating a Data frame sent outside the context of a BSS.

Address 1 filtering is as specified in 10.64 (Addressing of GLK Data frame transmission) when Address 1 is a SYNRA. A GLK AP does not perform any DA filtering for MPDUs received over a non-general link; all MSDUs so received are passed to the DS for further processing. A GLK STA does not perform DA filtering for MPDUs received over a general link; all MSDUs so received are passed to the GLK convergence function and from there to the bridge for further processing.

A mesh STA also uses the address matching rules described in 10.37.3 (Addressing and forwarding of individually addressed mesh Data frames), when it receives an individually addressed frame. When a mesh STA receives a frame with the Address 1 field equal to a group address, the mesh STA also checks the TA to determine whether the group addressed frame originated from one of its peer mesh STAs; if there is no match, the STA shall discard the frame. A mesh STA also uses the address matching rules described in 10.37.4 (Addressing and forwarding of group addressed mesh Data frames).

If the Address 1 field of an MPDU carrying an A-MSDU does not match any address at a receiving STA, then the entire A-MSDU is discarded.

In a QoS STA, the TID parameter of the MA-UNITDATA.request primitive results in a TID being specified for the transmitted MSDU. This TID associates the MSDU with the AC or TS queue for the indicated traffic.

------

**Changes or additions in Bold**

**Annex C**

**ASN.1 encoding of the MAC and PHY MIB**

**C.3 MIB detail**

- MAC Attributes

 -- DEFINED AS "The MAC object class provides the necessary support

 -- for the access control, generation, and verification of frame

 -- check sequences (FCSs), and proper delivery of valid data to

 -- upper layers."

 dot11mac OBJECT IDENTIFIER ::= { ieee802dot11 2 }

-- MAC GROUPS

 -- dot11OperationTable ::= { dot11mac 1 }

 -- dot11CountersTable ::= { dot11mac 2 }

 -- dot11GroupAddressesTable ::= { dot11mac 3 }

 -- dot11EDCATable ::= { dot11mac 4 }

 -- dot11QAPEDCATable ::= { dot11mac 5 }

 -- dot11QosCountersTable ::= { dot11mac 6 }

 -- dot11ResourceInfoTable ::= { dot11mac 7 }

 -- dot11DMGOperationTable ::= { dot11mac 8 }

 -- dot11DMGCountersTable ::= { dot11mac 9 }

 --  dot11BSSStatisticsTable ::= { dot11mac 10 }

 --  dot11CDMGOperationTable ::= { dot11mac 11 }

 --  (11ay)dot11EDMGOperationTable ::= { dot11mac 14 }

 --  (11ax)dot11MUEDCATable ::= { dot11mac 15 }

 **-- dot11RxAddressesTable ::= { dot11mac 16 }**

**-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**-- \* dot11RxAddresses TABLE**

**-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**dot11RxAddressesTable OBJECT-TYPE**

 **SYNTAX SEQUENCE OF Dot11RxAddressesEntry**

 **MAX-ACCESS not-accessible**

 **STATUS current**

 **DESCRIPTION**

 **"A conceptual table containing a set of MAC addresses identifying the MAC addresses for which this STA receives frames."**

 **::= { dot11mac 16 }**

**dot11RxAddressesEntry OBJECT-TYPE**

 **SYNTAX Dot11RxAddressesEntry**

 **MAX-ACCESS not-accessible**

 **STATUS current**

 **DESCRIPTION**

 **"An Entry (conceptual row) in the MAC Addresses Table for which this STA receives frames.**

 **ifIndex - Each IEEE 802.11 interface is represented by an ifEntry. Interface tables in this MIB module are indexed by ifIndex."**

 **INDEX { ifIndex, dot11RxAddressesIndex }**

 **::= { dot11RxAddressesTable 1 }**

**Dot11RxAddressesEntry ::=**

 **SEQUENCE {**

 **dot11RxAddressesIndex InterfaceIndex,**

 **dot11RxAddress MacAddress,**

 **dot11RxAddressesStatus RowStatus }**

**dot11RxAddressesIndex OBJECT-TYPE**

 **SYNTAX InterfaceIndex**

 **MAX-ACCESS not-accessible**

 **STATUS current**

 **DESCRIPTION**

 **"The auxiliary variable used to identify instances of the columnar objects in the Rx Addresses Table."**

 **::= { dot11RxAddressesEntry 1 }**

**dot11RxAddress OBJECT-TYPE**

 **SYNTAX MacAddress**

 **MAX-ACCESS read-create**

 **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.**

 **MAC address identifying addresses from which this STA receives frames."**

 **::= { dot11RxAddressesEntry 2 }**

**dot11RxAddressesStatus OBJECT-TYPE**

 **SYNTAX RowStatus**

 **MAX-ACCESS read-create**

 **STATUS current**

 **DESCRIPTION**

 **"The status column used for creating, modifying, and deleting instances of the columnar objects in the Rx Addresses Table."**

 **DEFVAL { active }**

 **::= { dot11RxAddressesEntry 3 }**

**-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**-- \* End of dot11RxAddresses TABLE**

**-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***