IEEE P802.11
Wireless LANs

|  |
| --- |
| Frame Anonymization (FA) normative text for 11bi |
| Date: 2023-11-14 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Duncan Ho | Qualcomm |  |  | dho@qti.qualcomm.com |
| Jouni Malinen |  |  |  |
| Philip Hawkes |  |  |  |
| George Cherian |  |  |  |

Abstract

We propose the draft specification for the following requirements in contribution “11-23-0892-03-00bi-requirements-and-issues-tracking” for TGbi draft D0.1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Requirement** | **Issue**  | **Status** | **Information** |
| 7 | 11bi shall define a mechanism for a CPE Client to initiate changing its own OTA MAC Address used with a CPE AP in Associate STA State 4 without any loss of connection. | MAC address change while associated | Discussions underway |   |
| 9 | Edited to: 11bi shall define a mechanism for a CPE Client and CPE AP to change the transmitted SN and the scrambler seed on downlink and uplink to uncorrelated new values in Associate STA State 4, without any loss of connection when the OTA MAC address of the CPE Client is changed. | MAC address change while associated | Discussions underway |   |
| 10 | Edited to: 11bi shall define a mechanism for a CPE Client and CPE AP to change the transmitted PN on downlink and uplink to uncorrelated new values in Associate STA State 4, without any loss of connection when the OTA MAC address of the CPE Client is changed. | MAC address change while associated | Discussions underway |   |
| 11 | 11bi shall define a mechanism for a CPE Client and CPE AP to change the CPE Client’s AID to an uncorrelated new value in Associate STA State 4, without any loss of connection when the OTA MAC address of the CPE Client is changed | MAC address change while associated | Discussions underway |  |

*Notes:*

* *Requirement 13 (11bi shall define or reuse a mechanism for CPE Clients and CPE APs to protect the SA/DA values from exposure OTA to 3rd parties) is expected to use a different mechanism which will be introduced separately.*

Revisions:

* Rev 0: Initial version of the document.
* Rev 1: editorial updates

**High level summary of the changes:**

* Introduce a new action frame category "Protected Enhanced Data Privacy" in 9.4.1.11 with Category code = 38
* Define FA Action frame within the new category
	+ 9.6.x "Protected Enhanced Data Privacy Action frame details"
		- 9.6.x.1 “Protected Enhanced Data Privacy Action field”. Defines values for
			* New FAPU Request frame (non-AP MLD initiates new FA epoch)
			* New FAPU Response frame (AP MLD provides new FA parameters, triggered by FAPU Request frame)
			* New FAPI Push Frame (AP MLD provides new FA parameters, unsolicited)
		- 9.6.x.2 “FAPU Request frame format”
		- 9.6.x.3 “FAPU Response frame format”
		- 9.6.x.4 “New FAPI Push Frame format”
* 10.x “Frame Anonymization”
	+ 10.x.1 “Introduction”
	+ 10.x.2 “Frame Anonymization Parameter Update”
		- Describes how FA parameters for FA epoch are established
	+ 10.x.3 “Frame Anonymization for individually addressed frames”
		- Describes how FA works for individually addressed frames
	+ 10.x.4 “Frame Anonymization and MLD upper MAC functions”
		- Covers SN /PN Anonymization.
	+ 10.x.5 “Frame Anonymization and MLD low MAC functions”
		- Covers use of FA STA MAC
	+ 10.x.6 “Frame Anonymization and beacon frames”
	+ 10.x.7 “Frame Anonymization and trigger frames”

**Key to highlighting:**

* Yellow highlighting: identifies instructions to the TGbi editor.
* Green highlighting: identifies normative terms like “shall”, “may”, “should”.
* Pink highlighting with black text: identifies cross-references that will need hyperlinks.
* Pink highlighting with yellow text: identifies cross references to unknown clauses (which will also need hyperlinks).
* Blue highlighting: identifies items that need addressing.

**List of open items:**

* MPDU encryption/decryption details (10.x.4)
	+ Which MAC address of the non-AP MLD is used in CCMP and GCMP?
	+ What details are needed for AAD and Nonce?
	+ Does clause 12 need an update?
* Receiver might need new and old otaMAC and (for S1G) new and old otaAID. Propose excluding PV1.
* Non-AP MLD timer after sending FAPU Request frame. (10.x.2.2)
* Time that a non-AP MLD waits after unsuccessful non=AP-initiated FAPU. (10.x.2.2.1)
* Do we need to worry about AID collisions during the epoch overlap? (10.x.2.4.1)
* Determining the active FA epoch (10.x.2.5)

**Proposed spec text:**

The baseline for this text is 802.11 REVme D4.1.

***TGbi editor: Add new definition to clause 3.2 (Definitions specific to IEEE Std 802.11):***

**active frame anonymization epoch:** frame anonymization epoch and its frame anonymization parameters which are being applied by the transmitter to MPDUs, beacon frames and trigger frames.

**frame anonymization**: mechanisms mitigating against presence monitoring using unencrypted fields in beacon frames and individually addressed frames.

**frame anonymization epoch:** time window in which a set of frame anonymization parameters remain constant.

**frame anonymization parameters**: set of parameters used in frame anonymization mechanisms.

**frame anonymization parameter update**: mechanism for establishing new parameters for use in frame anonymization mechanisms.

**obfuscated sequence number**: value transmitted in an individually addressed MPDU header in the place of the sequence number as part of frame anonymization.

**obfuscated packet number**: value transmitted in an individually addressed CCMP header or GCMP header in the place of the packet number as part of frame anonymization.

**presence monitoring**: determining the ongoing presence of non-AP MLDs associated to an AL MLD

**retiring frame anonymization epoch:** frame anonymization epoch which was active prior to the current active frame anonymization epoch, where MPDUs processed in that prior frame anonymization epoch are still being retransmitted.

***TGbi editor: Add new acronyms to clause 3.4 (Acronyms and abbreviations)as follows:***

FA frame anonymization

FAPU frame anonymization parameter update

OSN obfuscated sequence number

OPN obfuscated packet number

***TGbi editor: Add new subclause of 10.x (Frame Anonymization) under clause 10 (MAC sublayer functional description) as follows:***

# 10.x Frame Anonymization

10.x.1 Introduction

Frame Anonymization (FA) is a EDP feature available when MLO is supported. Some unencrypted fields in beacon frames and individually addressed frames contain values which facilitate *presence monitoring*, determining the continued presence of an non-AP MLD even if the long-term identify of the non-AP MLD cannot be determined. Presence monitoring can be a threat to privacy of the MLD user. User privacy can be improved by shortening the presence-monitoring time-windows.

The unencrypted fields which facilitate presence monitoring are:

* In beacon frames:
	+ traffic indication map (TIM), which allows determining the AIDs of associated non-AP MLDs.
* In trigger frames:
	+ AID of associated non-AP MLDs.
* In data plane individually addressed MPDUs:
	+ Address 1 (on the downlink) and Address 2 (on the uplink) which contains the MAC address of the Affiliated STA of the non-AP MLD on the link on which the frame is transmitted.
	+ Sequence Number (SN) used for Block Ack scoreboarding and MPDU re-ordering.
	+ Packet Number (SN) used for MPDU encryption / decryption and replay detection.

Two of the side effects of EDP additions to the (re-)association mechanism defined in TBD are (a) establishing new values for AID and the Affiliated STA MAC addresses which cannot be related to values in previous associations and (b) resetting SN and PN. These side effects prevent presence monitoring from one association to the next. Consequently, one mechanism a non-AP MLD can employ to create shorter presence monitoring time-windows is to initiate (re-)association while in State 4. However, (re-)association results in leaving State 4 and introduces a loss in connectivity which could create a negative user experience. Furthermore, this approach does not mitigate AP MLD presence monitoring.

FA enables restricting presence-monitoring time-windows to portions of a single association (that is, without leaving State 4). These time-windows are called *FA epochs*. FA provides of the following functions:

* **FAPU:** A *frame anonymization parameter update (FAPU)* operation provides new *FA parameters* for a new FA epoch:
	+ *FA Epoch Start TSF*: the TSF time at which the FA Epoch starts. The FA epoch ends when the next FA Epoch starts, or when the non-AP MLD’s association ends. The FA epoch start TSF can be selected by the non-AP MLD or the AP MLD using implementation-specific means.
	+ *FA AID*: new random value to be used as the non-AP MLD AID in AID randomization. The FA AID is selected by the AP MLD using implementation-specific means.
	+ Additional FA parameters are generated at both the non-AP MLD and AP MLD:
		- *FA STA MAC*: New random values for the Affiliated STA MAC address randomization for each set link are generated from KDK, FA Epoch Start, FA AID and the Link ID using a PRF.
		- *FA SN offset* and *FA* *PN offset*: new random values for use in SN / PN anonymization are generated from KDK, FA Epoch Start, FA AID and (in the case of SN) TID using a PRF.

An FAPU can be performed at any time while a non-AP MLD is in State 4. When the TSF time reaches FA Epoch Start, then the non-AP MLD and AP MLD being apply the other FA parameters to both transmitted and received frames. This results in the frames exchanged in an FA epoch appearing unrelated to the frames exchanged in other FA epoch.

* **AID randomization:** The 12-bit FA AID, provided by the AP MLD in the FAPU Response Frame, is used directly in the place of AID established at association or reassociation; for example, in computing and processing the traffic indication map (TIM) field of beacon frames, and in trigger frames. An eavesdropper would observe an effect (on the values in the beacons and trigger frames) after every FAPU which is like the effect of association or reassociation.
* **Affiliated STA MAC address randomization**. For each link, the 48-bit FA STA MAC address generated during the FAPU, is used directly in the MPDU header creation in the MLD lower MAC sublayer of the transmitter. Thes addresses are also used in the MLD lower MAC sublayer of the receiver: e.g., in Address 1 address filtering. An eavesdropper would observe an effect (on the values in the Address fields in individually addressed MPDUs) after every FAPU which is like the effect of association or reassociation.

NOTE—The AAD uses the non-AP MLD MAC address used for association and authentication, not the FA MAC address.

* **SN / PN anonymization**: The values of SN and PN cannot be reset while staying in State 4, because resetting SN breaks re-ordering and resetting PN facilitates replay attacks. Consequently, FA takes a different approach to association and reassociation. In FA, the predictable “internal” values of SN and PN (assigned in the MLD upper MAC layer) always continue to increment (i.e., are not affected by FA), but FA keeps the values secret to maintain anonymity. In the place of a reset, a FAPU establishes secret SN offset and PN offset parameters which are used to transforms the “internal” values into “obfuscated” values which can be safely transmitted in the clear while maintaining anonymity. Following an FAPU, the “obfuscated” values appear to do a random “jump” to a new starting value, and then continue incrementing from the new starting value. This transformation from “internal” values into “obfuscated” values in the transmitter is called SN/ PN anonymization. The intended receiver transforms the “obfuscated” values back to the “internal” values - a process called SN/ PN de-anonymization. Functions which use SN and/or PN in the MLD upper MAC sublayer make use of the “internal” values (e.g., MPDU reordering, MPDU encryption/decryption and replay detection), while functions which use SN in the MLD lower MAC sublayer make use of the “obfuscated” values (e.g., MPDU header creation and Block Ack scoreboarding).
	+ Each FAPU generates an independent, secret 12-bit SN offset for each combination of TID and direction (uplink and downlink). For each MPDU, the transmitter computes OSN = SN + SN\_ Offset (modulo 212), where OSN is the obfuscated SN value transmitted over-the-air, SN is the internal SN value, and SN\_Offset is the SN offset parameter for combination of TID and direction for the frame. The OSN value is transmitted in the SN field of the MPDU header. For each MPDU, the receiver reverses the operation to determine the internal SN = OSN – SN\_ Offset (modulo 212) from the OSN value received in the MPDU header.
	+ Each FAPU generates an independent, secret 48-bit PN offset for each direction (uplink and downlink). For each MPDU, the transmitter computes OPN = PN + PN\_Offset (modulo 248), where OPN is the obfuscated PN value, PN is the internal PN value and PN\_Offset is the PN offset parameter for the direction of the frame. The OPN value is transmitted in the PN0 to PN5 fields of the CCMP header or GCMP header. For each MPDU, the receiver reverses the operation to determine the internal PN = OPN – PN\_Offset (modulo 248) from the OPN value received in the PN0 to PN5 fields of the CCMP header or GCMP header.

The OSN and OPN transmitted in the frames within a FA Epoch form a trackable incrementing sequence, however the change of SN offsets and PN offsets at every FAPU introduce a random “jump” when each new FA Epoch starts. This jump prevents attackers exploiting OPN and OSN to monitor presence across multiple FA Epochs.

Most FA function are integrated to various functions in the MAC data plane architectures in Figure 5-2a (MAC data plane architecture (MLO) for individually addressed data frames) and Figure 5-2b (MAC data plane architecture for AP MLD and affiliated APs):

* The FAPU function is a MLD upper MAC sublayer function.
* The SN / PN Anonymization function is a separate MLD upper MAC sublayer function, although implementations could include this function in the MLD lower MAC sublayer.
* The Affiliated STA MAC address randomization function is integrated to MLD lower MAC sublayer functions: MPDU header creation on the transmitting flow, Address 1 filtering on the downlink receiving flow and Address 2 filtering on the uplink receiving flow.

NOTE— Address 2 filtering is not shown in Figure 5-2b (MAC data plane architecture for AP MLD and affiliated APs):

The AID Randomization function is integrated to beacon frame flow and trigger frame flow, which are separate from the MAC data plane architecture.

NOTE—The following list clarifies the scope of attacks which FA mitigates:

* FA mitigates against presence monitoring across multiple FA epochs.
* FA does not mitigate against presence monitoring within a single FA epoch.
* FA does not mitigate identifying frames belonging to a single MLD within a single FA epoch.
* FA does not mitigate using traffic analysis using known transmission behavior of upper layer protocols for presence monitoring across multiple FA epochs.

An associated non-AP MLD can initiate a FAPU by sending a individually addressed FAPU Request action frame (containing a proposed FA Epoch Start TSF element) to the AP MLD. An AP MLD sends an individually addressed FAPU Response action frame (containing an FA epoch start TSF element and FA AID element) to a non-AP MLD. The FAPU Response action frame can be sent in reply to a FAPU request frame or unsolicited.

An AP MLD can send FAPU Response action frames with identical values for the FA Epoch Start TSF element to a multiple non-AP MLDs. This results in the non-AP MLDs synchronizing their transition to new FA parameters.

When transitioning from an old FA epoch to a new FA epoch, there is a short overlap in time where the transmitter and receiver allow both retransmissions of old frames (created using FA parameters of the old FA epoch) and transmissions of new frames (created using FA parameters of the new FA epoch). Old frames and new frames are not mixed within a Block Ack or A-MPDU or (on the uplink) TXOP, since this would facilitate relating the old frames to the new frames, resulting in presence monitoring across the FA epoch transition.

10.x.2 Frame Anonymity Parameter Update (FAPU)

10.x.2.1 Introduction

The Protected FA Action frames defined in 9.6.x (Protected FA Action frame details) provide the mechanisms for a non-AP MLD and AP MLD to establish new FA parameters.

A FAPU can be initiated by the non-AP MLD or the AP MLD. A non-AP-MLD-initiated FAPU uses a FAPU Request frame (sent from the non-AP MLD to the AP MLD) and a corresponding FAPU Response frame (sent from the AP MLD to the non-AP MLD).

* The FAPU Request frame is sent by the non-AP MLD to request a new FA epoch for the non-AP MLD.
* After the AP MLD receives the FAPU Request frame from a non-AP MLD, the AP MLD sends a corresponding FAPU Response frame to provide the non-AP MLD with FA parameters for a FA epoch.

The processes for a non-AP-MLD-initiated FAPU are provided in 10.x.2.2 (Non-AP-MLD-initiated FAPU).

An AP-MLD-initiated FAPU uses the FAPU Push frame (sent from the AP MLD to the non-AP MLD) to provide FA parameters for a FA epoch. The processes for an AP-MLD-initiated FAPU are provided in 10.x.2.3 (AP-MLD-initiated FAPU).

If a non-AP MLD is disassociated with an AP MLD, then the AP MLD and non-AP MLD shall remove all FA Epoch Parameters for the non-AP MLD.

10.x.2.2 Non-AP-MLD-initiated FAPU

**Non-AP MLD processes:** The conditions under which a non-AP MLD determines to initiate a non-AP-MLD-initiated FAPU are implementation specific.

A non-AP MLD initiates a non-AP-MLD-initiated FAPU by sending an FAPU Request frame defined in 9.6.x.2 (FAPU Request frame format) with elements set as follows:

* Dialog token: shall be selected by the non-AP MLD as described in 10.28.5 (Operation of the Dialog Token field).

The non-AP MLD transitions the non-AP-MLD-initiated FAPU to “open” state when the non-AP MLD transmits the FAPU Request frame. The non-AP MLD transitions the non-AP-MLD-initiated FAPU to “closed” state when one the following events occurs:

* The non-AP MLD receives a FAPU Response Frame defined in 9.6.x.3 (FAPU Response frame format) with dialog token field matching the value of dialog token field in the corresponding FAPU Request frame.
* The association with the AP MLD is ended.
* A TBD Timer expires.

At all times, a non-AP MLD shall not have more than one non-AP-MLD-initiated FAPU in “Open” state.

A non-AP MLD shall ignore a FAPU Response Frame if the dialog token field does not match the value of dialog token field of the FAPU Request frame of an open non-AP-MLD-initiated FAPU.

**AP MLD processes:** The AP MLD shall interpret a received FAPU Request frame as a request to provide FA parameters for a FA epoch for a non-AP MLD. The AP MLD determines, via implementation-specific means, whether the AP MLD can provide FA parameters to the AP MLD.

* If the AP MLD cannot provide FA parameters, then 10.x.2.2.1 (Unsuccessful Non-AP-MLD-initiated FAPU) shall be applied.
* If the AP MLD can provide FA parameters, then 10.x.2.2.2 (Unsuccessful Non-AP-MLD-initiated FAPU) shall be applied.

10.x.2.2.1 Unsuccessful Non-AP-MLD-initiated FAPU

**AP MLD processes:** If the AP MLD determines that the AP MLD cannot provide FA parameters in response to an FAPU Request frame, then the AP MLD shall transmit an FAPU Response Frame defined in 9.6.x.3 (FAPU Response frame format) with fields set as follows:

* Dialog token field: shall match the value of dialog token field in the corresponding FAPU Request frame.
* Status code field: shall be set to REFUSED\_REASON\_UNSPECIFIED (1).
* FA Epoch Start TSF element: shall be set to all zeroes.
* FA AID: shall be set to all zeroes.

In the case of an unsuccessful non-AP-MLD-initiated FAPU:

* If there is a set of active FA Parameters for that association, then the AP MLD shall apply FA to received and transmitted frames using the active FA epoch.
* If there is no set of active FA Parameters for that association, then the AP MLD shall not apply FA to received and transmitted frames.

**Non-AP MLD processes:** If the non-AP MLD has initiated an open non-AP-MLD-initiated FAPU and receives a FAPU Response Frame which meets the following conditions, then the non-AP MLD determines that the open non-AP-MLD-initiated FAPU is unsuccessful:

* Dialog token field: matches the value of dialog token field in the FAPU Request frame of an open non-AP-MLD-initiated FAPU.
* Status code field: matches REFUSED\_REASON\_UNSPECIFIED (1).

In the case of an unsuccessful non-AP-MLD-initiated FAPU:

* The non-AP MLD shall refrain from initiating another non-AP-MLD-initiated FAPU for at least TBD seconds.
* If there is a set of active FA Parameters for that association, then the non-AP MLD shall apply FA to received and transmitted frames using the active FA epoch.
* If there is no set of active FA Parameters for that association, then the non-AP MLD shall not apply FA to received and transmitted frames.

10.x.2.2.2 Successful Non-AP-MLD-initiated FAPU

**AP MLD processes:** If the AP MLD determines that the AP MLD can provide FA parameters in response to an FAPU Request frame, then the AP MLD shall transmit an FAPU Response Frame with defined in 9.6.x.3 (FAPU Response frame format) with fields set as follows:

* Dialog token field: shall match the value of the dialog token field in the corresponding FAPU Request frame.
* Status code field: shall be set to SUCCESS (0) as defined in Table 9-50 (Status codes).
* FA Epoch Start TSF element: see 10.x.2.4 (Selecting and Deriving FA Parameters).
* FA AID: see 10.x.2.4 (Selecting and Deriving FA Parameters).

In the case of a successful non-AP-MLD-initiated FAPU, the AP MLD shall create a set of FA parameters for the FA epoch as follows:

* The AP MLD shall set the FA Epoch Start TSF parameter to the value in the FA Epoch Start TSF elements in the transmitted FAPU Response Frame.
* The AP MLD shall set the FA AID parameter to the value in the FA AID elements in the transmitted FAPU Response Frame.
* The AP MLD shall derive the remaining FA Parameters as described in 10.x.2.4 (Selecting and Deriving FA Parameters).

**Non-AP MLD processes:** If the non-AP MLD has initiated an open non-AP-MLD-initiated FAPU and receives a FAPU Response Frame which meets the following conditions, then the non-AP MLD determines that the open non-AP-MLD-initiated FAPU is successful.

In the case of a successful non-AP-MLD-initiated FAPU, the non-AP MLD shall create a set of FA parameters for the FA epoch as follows:

* The non-AP MLD shall set the FA Epoch Start TSF parameter to the value in the FA Epoch Start TSF elements in the transmitted FAPU Response Frame.
* The non-AP MLD shall set the FA AID parameter to the value in the FA AID elements in the transmitted FAPU Response Frame.
* The non-AP MLD shall derive the remaining FA Parameters as described in 10.x.2.4 (Selecting and Deriving FA Parameters).

10.x.2.3 AP-MLD-initiated FAPU

**AP MLD processes:** The conditions under which an AP MLD determines to initiate an AP-MLD-initiated FAPU are implementation specific.

A non-AP MLD initiates a non-AP-MLD-initiated FAPU by sending an FAPU Push frame defined in 9.6.x.3 (FAPU Response frame format) with fields set as follows:

* FA Epoch Start TSF element: see 10.x.2.4 (Selecting and Deriving FA Parameters).
* FA AID: see 10.x.2.4 (Selecting and Deriving FA Parameters).

If the AP MLD does not successfully deliver the FAPU Push frame, then:

* If there is a set of active FA Parameters for that association, then the AP MLD shall apply FA to received and transmitted frames using the active FA epoch.
* If there is no set of active FA Parameters for that association, then the AP MLD shall not apply FA to received and transmitted frames.

If the AP MLD successfully delivers the FAPU Push frame, then the AP MLD shall create a set of FA parameters for the FA epoch as follows:

* The AP MLD shall set the FA Epoch Start TSF parameter to the value in the FA Epoch Start TSF elements in the transmitted FAPU Push frame.
* The AP MLD shall set the FA AID parameter to the value in the FA AID elements in the transmitted FAPU Push frame.
* The AP MLD shall derive the remaining FA Parameters as described in 10.x.2.4 (Selecting and Deriving FA Parameters).

**Non-AP MLD processes:** If the non-AP MLD receives a FAPU Push frame, then the non-AP MLD shall create a set of FA parameters for the FA epoch as follows:

* The non-AP MLD shall set the FA Epoch Start TSF parameter to the value in the FA Epoch Start TSF elements in the received FAPU Response Frame.
* The non-AP MLD shall set the FA AID parameter to the value in the FA AID elements in the received FAPU Response Frame.
* The non-AP MLD shall derive the remaining FA Parameters as described in 10.x.2.4 (Selecting and Deriving FA Parameters).

10.x.2.4 Selecting and Deriving FA Parameters

10.x.2.4.1 Selecting FA Epoch Start TSF and FA AID

The AP MLD selects/assigns the values of FA Epoch Start TSF element and FA AID element using implementation-specific means, subject to the following requirements.

The FA Epoch Start TSF identifies when the FA parameters (established by a FAPU) become active.

* An AP MLD may assign identical FA Epoch Start TSF in FA Parameters for multiple non-AP MLDs.

NOTE— This this case the FA Parameters for those non-AP MLDs become active concurrently.

* An AP MLD may assign an FA Epoch Start TSF unique to a single non-AP MLD.
* An AP MLD may assign an FA Epoch Start TSF in the past (that is, with value less than the TSF at the time of transmitting the FAPU Response frame).

NOTE— In this case the FA Parameters for the non-AP MLD(s) become active upon transmission of the FAPU Response frame.

The FA AID identifies the non-AP MLD for the duration of the FA epoch.

* The AP MLD selection of the FA AIDs shall ensure that, at all times, no two non-AP MLD are assigned the identical values of FA AID. Otherwise, the AP MLD selection of this value is implementation specific.

NOTE— The FA AID are assigned by the AP MLD to ensure AID values are kept small while meeting these requirements.

* The FA AID assigned to a non-AP should change pseudo randomly at every FAPU.

10.x.2.4.2 Deriving other FA Parameters

The values for other FA parameters (that is, FA STA MAC parameters, FA PN Offset parameters and FA SN Offset parameters) are derived from KDK of the association and other inputs including FA Epoch Start TSF and FA AID.

**FA STA MAC** parameters: For each setup link established during association, the FA STA MAC value for the setup link identified by Link ID shall be derived as:

FA STA MAC (Link ID) = 6 least significant bytes of PRF-256 (KDK, “FA STA MAC”, FA Epoch Start TSF || FA AID || Link ID),

where:

* PRF is the pseudorandom function as defined in 12.7.1.2 (PRF) using the hash algorithm identified by the AKM suite selector (see Table 9-188 (AKM suite selectors)),
* FA Epoch Start TSF is the value of the FA Epoch Start TSF element in the FAPU Response frame or FAPU Push frame, with format defined in 9.4.1.10 (Timestamp field),
* Link ID indicates the identifier of the link and has format defined in 9.4.1.75 (Link ID Info field),
* FA AID is the value in the FA AID element in the FAPU Response frame, with format defined in 9.4.1.8 (AID).

The value of FA STA MAC (Link ID) is used to identify the Affiliated STA of the non-AP MLD on the link identified by Link ID in the Address 1 MPDU header field of downlink frames and the Address 2 MPDU header field of uplink frames.

**FA PN Offset** parameters: The FA PN Offset values are independent of the link on which the MPDU is transmitted. The FA PN Offset values PN\_Offset\_UL (for uplink) and PN\_Offset\_DL (for downlink) shall be derived as:

FA\_PN\_Offsets\_All = 12 least significant bytes of PRF-256 (KDK, “FA PN Offset”, FA Epoch Start TSF|| FA AID).

FA\_PN\_Offset\_UL = 6 least significant bytes of FA\_PN\_Offsets\_All,

FA\_PN\_Offset\_DL = 6 most significant bytes of FA\_PN\_Offsets\_All,

where:

* PRF is the pseudorandom function as defined in 12.7.1.2 (PRF) using the hash algorithm identified by the AKM suite selector (see Table 9-188 (AKM suite selectors)),
* FA Epoch Start TSF is the value of the FA Epoch Start TSF element received in the FAPU Response frame, or FAPU Push frame with format defined in 9.4.1.10 (Timestamp field),
* FA AID is the value in the FA AID element in the FAPU Response frame or FAPU Push frame, with format defined in 9.4.1.8 (AID).

**FA SN Offset** parameters: The FA SN Offset values are independent of the link on which the MPDU is transmitted.

The 16 per-TID 12-bit SN Offset values for the uplink direction are obtained by generating a 16 x 12 = 192-bit value FA\_SN\_Offsets\_UL using a PRF and then partitioning FA\_SN\_Offsets\_UL into an array of 16 per-TID uplink SN Offset values FA\_SN\_Offsets\_UL[0], …, FA\_SN\_Offsets\_UL[15].

Similarly, the 16 per-TID 12-bit FA SN Offset values for the downlink direction applied in a FA Epoch are obtained by generating a 16 x 12 = 192-bit value FA\_SN\_Offsets\_DL and then partitioning FA\_SN\_Offsets\_DL into an array of 16 per-TID downlink SN Offset values FA\_SN\_Offsets\_DL[0], …, FA\_SN\_Offsets\_DL[15].

The values of FA\_SN\_Offsets\_UL and FA\_SN\_Offsets\_DL shall be derived as:

FA\_SN\_Offsets\_UL = PRF-192 (KDK, “FA SN Offsets UL”, FA Epoch Start TSF || FA AID),

FA\_SN\_Offsets\_DL = PRF-192 (KDK, “FA SN Offsets DL”, FA Epoch Start TSF || FA AID).

where:

* PRF is the pseudorandom function as defined in 12.7.1.2 (PRF) using the hash algorithm identified by the AKM suite selector (see Table 9-188 (AKM suite selectors)),
* FA Epoch Start TSF is the value of the FA Epoch Start TSF element received in the FAPU Response frame, or FAPU Push frame with format defined in 9.4.1.10 (Timestamp field),
* FA AID is the value in the FA AID element in the FAPU Response frame or FAPU Push frame, with format defined in 9.4.1.8 (AID).

The 192-bit FA\_SN\_Offsets\_UL shall be partitioned into an array of 16 per-TID 12-bit uplink FA SN Offset values FA\_SN\_Offset\_UL[0], …, FA\_SN\_Offset\_UL[F] as shown in Figure 9-[DDD] (Partitioning FA\_SN\_Offsets\_UL).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B0 B11 | B12 B23 | B24 B35 | B36 B47 | B48 B59 | B60 B71 | B72 B83 | B84 B95 |
|  | FA\_SN\_Offset\_UL[0]  | FA\_SN\_Offset\_UL[1] | FA\_SN\_Offset\_UL [2] |  FA\_SN\_Offset\_UL[3] | FA\_SN\_Offset\_UL 4] | FA\_SN\_Offset\_UL[5] | FA\_SN\_Offset\_UL[6] | FA\_SN\_Offset\_UL[7] |
| Bits: | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
|  |  |  |  |  |  |  |  |  |
|  | B96 B107 | B108 B119 | B120 B131 | B132 B143 | B144 B155 | B156 B167 | B168 B179 | B180 B191 |
|  | FA\_SN\_Offset\_UL[8]  | FA\_SN\_Offset\_UL[9] | FA\_SN\_Offset\_UL [10] |  FA\_SN\_Offset\_UL[11] | FA\_SN\_Offset\_UL[12] | FA\_SN\_Offset\_UL[13] | FA\_SN\_Offset\_UL[14] | FA\_SN\_Offset\_UL[15] |
| Bits: | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |

1. Figure 9-[DDD] — Partitioning SN\_Offsets\_UL

The 192-bit FA\_FA SN Offset values FA\_SN\_Offset\_DL[0], …, FA\_SN\_Offset\_DL[F] as shown in Figure 9-[EEE] (Partitioning FA\_SN\_Offsets\_DL).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B0 B11 | B12 B23 | B24 B35 | B36 B47 | B48 B59 | B60 B71 | B72 B83 | B84 B95 |
|  | FA\_SN\_Offset\_DL[0]  | FA\_SN\_Offset\_DL[1] | FA\_SN\_Offset\_DL [2] |  FA\_SN\_Offset\_DL[3] | FA\_SN\_Offset\_DL 4] | FA\_SN\_Offset\_DL[5] | FA\_SN\_Offset\_DL[6] | FA\_SN\_Offset\_DL[7] |
| Bits: | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
|  |  |  |  |  |  |  |  |  |
|  | B96 B107 | B108 B119 | B120 B131 | B132 B143 | B144 B155 | B156 B167 | B168 B179 | B180 B191 |
|  | FA\_SN\_Offset\_DL[8]  | FA\_SN\_Offset\_DL[9] | FA\_SN\_Offset\_DL [10] |  FA\_SN\_Offset\_DL[11] | FA\_SN\_Offset\_DL[12] | FA\_SN\_Offset\_DL[13] | FA\_SN\_Offset\_DL[14] | FA\_SN\_Offset\_DL[15] |
| Bits: | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |

1. Figure 9-[EEE] — Partitioning FA\_SN\_Offsets\_DL
2. 10.x.2.5 Determining the active FA epoch

This clause will be the definitive text for identifying the active FA epoch.

10.x.3 Frame Anonymization for individually addressed frames

This clause is arranged as follows:

* For downlink individually addressed frames, AP MLD transmission processing is specified in 10.x.3.1 (AP MLD transmission processing for downlink individually addressed frames) and non-AP MLD receiving processing is specified in 10.x.3.2 (Non-AP MLD receiving processing for downlink individually addressed frames).
* For uplink individually addressed frames, non-AP MLD transmission processing is specified in 10.x.3.3 (Non-AP MLD transmission processing for uplink individually addressed frames) and AP MLD receiving processing is specified in 10.x.3.4 (AP MLD receiving processing for uplink individually addressed frames).

The descriptions assume that the FA parameters for the current FA epoch are known to the transmitter. For details see 10.x.2 (Frame Anonymity Parameter Update (FAPU)).

The processing is described with respect to MAC data plane architectures introduced in 5.1.5.1 (General):

* Common functions applied in the non-AP MLD and per-link functions applied in Affiliated STAs align with Fig 5-2a (MAC data plane architecture (MLO) for individually addressed data frames).
* Common functions applied in the AP MLD and per-link functions applied in Affiliated APs align with Fig 5-2b (MAC data plane architecture for AP MLD and affiliated APs).

NOTE— FA processing for individually addressed frames is identical for the uplink direction and downlink direction, with the following exceptions:

* The FA STA MAC is used in Address 1 of individually addressed frames in the downlink direction and in Address 2 of individually addressed frames in the uplink direction.
* The SN values in individually addressed frames in the uplink direction are obfuscated using SN Offset values which are distinct from the SN Offset values used to obfuscate SN values in individually addressed frames in the downlink direction.
* The PN values in individually addressed frames in the uplink direction are obfuscated using an PN Offset value which is distinct from the PN Offset value used to obfuscate PN values in individually addressed frames in the downlink direction.

10.x.3.1 MAC data plane architecture for FA

Figure 10-x (MAC data plane architecture for frame anonymization) is a version of the MAC data plane architecture shown in Figure 5-2a (MAC data plane architecture (MLO) for individually addressed data frames) and Figure 5-2b (MAC data plane architecture for AP MLD and affiliated APs), which illustrates where FA impacts the MAC data plane architecture. The description in 10.x.3 (Frame anonymization for individually addressed frames), 10.x.4 (Frame Anonymization and MLD upper MAC sublayer) and 10.x.5 (Frame Anonymization and MLD upper MAC sublayer) are described with reference to Figure 10-x (MAC data plane architecture for frame anonymization).



Figure 10-a MAC data plane architecture for frame anonymization

10.x.3.2 FA processing for uplink individually addressed frames

1. The FA processing of uplink individually addressed frames can be partitioned into the following meta-steps.
* The non-AP MLD applies transmitting functions described in 10.x.4.1 (FA and MLD upper MAC transmitting functions):
	+ The functions above (that is before) the SN/PN anonymization function are not impacted by FA.
	+ SN/ PN Anonymization function is applied to the MPDU frame using the FA SN Offset and FA PN Offset parameters.
* The Affiliate STA on the selected link applies transmitting functions described in 10.x.5.1 (Uplink transmitting frame anonymization processing in the Affiliated STA):
	+ The MPDU Header Creation function is applied using the FA STA MAC assigned to the Affiliated STA on selected link in Address 2.
	+ FCS Creation function is not impacted by FA.
	+ The A-MPDU aggregation function is applied, ensuring that every MPDU in an A-MPDU use identical set of FA parameters.
	+ The TXOP transmitting function is applied, ensuring that every MPDU in a TXOP use identical set of FA parameters.
	+ Retransmissions are discussed in 10.x.5.5 (Retransmission frame anonymization processing).
* MPDUs and A-MPDUs are transmitted from the Affiliated AP to the Affiliated STA via the PHY of the selected link.
* The Affiliate AP on the selected link applies receiving functions described in 10.x.5.2 (Uplink receiving frame anonymization processing in the Affiliated AP):
	+ The A-MPDU de-aggregation function, FCS Validation function, MPDU Header Validation function and Address 1 address filtering function are not impacted by FA.
	+ The Address 2 address filtering function is applied using the FA STA MAC assigned to the Affiliated STA.
	+ MLD lower MAC Block Ack Scoreboarding function is not impacted by FA.
* The AP MLD applies receiving functions described in 10.x.4.2 (FA and MLD upper MAC receiving functions):
	+ The SN/ PN de-anonymization function is applied to the MPDU frame using applicable SN Offset and PN Offset parameters.
	+ The functions above (that is after) the SN/PN de-anonymization function are not impacted by FA. These functions use the (internal) SN values and (internal) PN values.

NOTE— Since the input to the FCS Creation function is changed by FA, the resulting FCS can be changed by FA.

10.x.3.3 FA processing for downlink individually addressed frames

1. The FA processing of downlink individually addressed frames can be partitioned into the following meta-steps.
* The AP MLD applies transmitting functions described in 10.x.4.1 (FA and MLD upper MAC transmitting functions):
	+ The functions above (that is before) the SN/PN anonymization function are not impacted by FA.
	+ SN/ PN Anonymization function is applied to the MPDU frame using the FA SN Offset and FA PN Offset parameters.
* The Affiliate AP on the selected link applies transmitting functions described in 10.x.5.3 (Downlink transmitting frame anonymization processing in Affiliated AP):
	+ The MPDU Header Creation function is applied using the FA STA MAC assigned to the Affiliated STA on selected link in Address 2.
	+ FCS Creation function is not impacted by FA.
	+ The A-MPDU aggregation function is applied, ensuring that every MPDU in an A-MPDU use identical set of FA parameters.
	+ The TXOP transmitting function is not impacted by FA.
	+ Retransmissions are discussed in 10.x.5.5 (Retransmission frame anonymization processing).
* MPDUs and A-MPDUs are transmitted form the Affiliated AP to the Affiliated STA via the PHY of the selected link.
* The Affiliate STA on the selected link applies receiving functions described in 10.x.5.4 (Downlink receiving frame anonymization processing in Affiliated STA):
	+ The A-MPDU de-aggregation function, FCS Validation function and MPDU Header Validation function are not impacted by FA.
	+ Address 1 address filtering is applied using the FA STA MAC assigned to the Affiliated STA.
	+ MLD lower MAC Block Ack Scoreboarding is not impacted by FA.
* The non-AP MLD applies receiving functions described in 10.x.4.2 (FA and MLD upper MAC receiving functions):
	+ The SN/ PN de-anonymization function is applied to the MPDU frame using applicable SN Offset and PN Offset parameters.
	+ The functions above (that is after) the SN/PN de-anonymization function are not impacted by FA. These functions use the (internal) SN values and (internal) PN values.

10.x.4 Frame Anonymization and MLD upper MAC sublayer

This clause describes how FA impacts MAC upper layer functions. The description is identical for the uplink and downlink directions.

10.x.4.1 FA and MLD upper MAC transmitting functions

1. This clause describes how FA impacts MAC upper layer functions in the transmitting flow. The description is identical for the uplink and downlink directions.

FA has no interaction with the transmitting functions above (i.e., before) the Sequence Number Assignment function.

**Sequence Number (SN) Assignment** function: This function is not impacted by FA. That is, this function assigns the (internal) SN value sequentially per TID, whether FA is being applied or not.

**Packet Number (PN) Assignment** function: is not impacted by FA. That is, this function assigns the (internal) PN value, whether FA is being applied or not.

**MPDU Encryption** function: FA has no impact on this function. This function uses the (internal) PN value produced by the PN assignment function. What details about AAD and nonce are needed?

**TID-to-Link mapping** function: FA has no impact on TID, so FA has no impact on this function.

1. **SN / PN De-anonymization** function**:** This function is applied only when FA is enabled. The description of this function is similar for the uplink and downlink directions.

If there is no active FA epoch at the time when the MPDU is received at this function, then FA shall not be applied and the inputs to this function shall be passed through the next function without modification.

If there is an active FA epoch at the time when the MPDU is received at this function, then FA shall be applied using the active set of FA parameters.

When FA is applied, then this function applies the following processes.

* An obfuscated SN (OSN) value is computed from the (internal) SN value input to this function.
* An obfuscated PN (OPN) value is computed from the (internal) PN value input to this function.
* The output of this function shall be the input to this function with the following changes:
	+ The (internal) SN value shall be replaced with the OSN value.
	+ The value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP header or GCMP header of the MPDU shall be updated to contain the OPN value.
	+ Implementation specific information (e.g., FA Epoch Start TSF) shall be added so lower layer functions can determine the FA STA MAC for the selected link.

In the uplink direction, the OSN value and OPN value shall be computed as:

OSN = SN + FA\_SN\_Offset\_UL[TID] (mod 212),

OPN = PN + FA\_PN\_Offset\_UL (mod 248),

where:

* SN is an input to this function,
* PN is the value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP header or GCMP header of the MPDU input to this function,
* TID is the TID associated with the MPDU,
* FA\_SN\_Offset\_UL[TID] and FA\_PN\_Offset\_UL are generated for the active set of FA parameters as per 10.x.2.4.2 (Deriving other FA parameters), and
* “(mod 2n)” represents reducing the value modulo 2n to produces a value in the range [0, 2n-1].

 In the downlink direction, the OSN value and OPN value shall be computed as:

OSN = SN + FA\_SN\_Offset\_DL[TID] (mod 212),

OPN = PN + FA\_PN\_Offset\_DL (mod 248),

where:

* SN is an input to this function,
* PN is the value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP header or GCMP header of the MPDU input to this function,
* TID is an input to this function,
* FA\_SN\_Offset\_DL[TID] and FA\_PN\_Offset\_DL are generated for the active set of FA parameters as per 10.x.2.4.2 (Deriving other FA parameters), and
* “(mod 2n)” represents reducing the value modulo 2n to produces a value in the range [0, 2n-1].

10.x.4.2 FA and MLD upper MAC receiving functions

1. This clause describes how FA impacts MAC upper layer functions in the receiving flow. The description is identical for the uplink and downlink directions.

**SN / PN De-anonymization** function**:** This function is applied only when FA is enabled. The description of this function is similar for the uplink and downlink directions.

1. NOTE— The input to this function includes implementation specific information (e.g., FA Epoch Start TSF) identifying the set of FA Parameters used to process this MPDU. The output of this function includes the (internal) SN value and (internal) PN value (defined in 10.x.4.1), whether FA is applied or not.

If there is no active set of FA parameters identified for this MPDU is received at this function, then FA shall not be applied and the inputs to this function shall be passed through the next function without modification.

If there is an active set of FA parameters identified for this MPDU, then FA shall be applied using the active set of FA parameters.

When FA is applied, then this function applies the following processes.

* An (internal) SN value is computed from the obfuscated SN (OSN) value input to this function.
* An (internal) PN value is computed from the obfuscated PN (OPN) value in the CCMP header or GCMP header of the MPDU input to this function.
* The output of this function shall be the inputs to this function with the following changes:
	+ The OSN value shall be replaced with the (internal) SN value.
	+ The value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP header or GCMP header of the MPDU shall be updated to contain the (internal) PN value.

In the uplink direction, the (internal) SN value and (internal) PN value shall be computed as:

(internal) SN = OSN - SN\_Offset\_UL[TID] (mod 212),

(internal) PN = OPN - PN\_Offset\_ UL (mod 248),

where:

* OSN is an input to this function,
* OPN is the value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP header or GCMP header of the MPDU input to this function,
* TID is the TID associated with the MPDU,
* SN\_Offset\_UL[TID] and PN\_Offset\_UL are generated for the active set of FA parameters as per 10.x.2.4.2 (Deriving other FA parameters), and
* “(mod 2n)” represents reducing the value modulo 2n to produces a value in the range [0, 2n-1].

 In the downlink direction, the (internal) SN value and (internal) PN value shall be computed as:

(internal) SN = OSN - SN\_Offset\_UL[TID] (mod 212),

(internal) PN = OPN - PN\_Offset\_ UL (mod 248),

where:

* OSN is an input to this function,
* OPN is the value in fields PN0, PN1, PN2, PN3, PN4, PN5 of the CCMP Header or GCMP Header of the MPDU input to this function,
* TID is an input to this function,
* SN\_Offset\_DL[TID] and PN\_Offset\_DL are generated for the active set of FA parameters as per 10.x.2.4.2 (Deriving other FA parameters), and
* “(mod 2n)” represents reducing the value modulo 2n to produces a value in the range [0, 2n-1].
1. NOTE— The input to the remaining set of function includes the (internal) SN value and (internal) PN value (defined in 10.x.4.1.1), whether FA is applied or not.
2. **Link merging** function: FA has no impact on TID, so FA has no impact on this function.
3. **MLD upper MAC Block Ack Scoreboarding** function, **Duplicate Detection per SN** function and **Block Ack Buffering and Reordering per SN** function: These functions are not impacted by FA, because these functions use the (internal) SN values assigned by the Sequence Number Assignment function in the transmitter, whether FA is being applied or not.
4. **MPDU Decryption** function: This function is not impacted by FA, because these functions use the (internal) PN values assigned by the Packet Number Assignment function in the transmitter, whether FA is being applied or not. What details about AAD and nonce are needed?
5. (Optional) **Replay Detection Per PN** function: This function is not impacted by FA, because these functions use the (internal) PN values assigned by the Packet Number Assignment function in the transmitter, whether FA is being applied or not.
6. FA has no interaction with the receiving functions above (i.e., after) the Replay Detection Per PN function.

10.x.5 Frame Anonymization and MLD lower MAC sublayer

This clause describes how MLD lower MAC sublayer functions work when FA is enabled:

* Uplink transmitting frame anonymization processing in the Affiliated STA is described in 10.x.5.1.
* Uplink receiving frame anonymization processing in the Affiliated AP is described in 10.x.5.2.
* Downlink transmitting frame anonymization processing in the Affiliated AP is described in 10.x.5.3.
* Downlink receiving frame anonymization processing in the Affiliated STA is described in 10.x.5.4.
* Retransmissions processing is specified in 10.x.5.5 (Retransmission Processing).

10.x.5.1 Uplink transmitting frame anonymization processing in the Affiliated STA

When FA is enabled, then the transmitting Affiliated STA of the non-AP MLD shall apply the following uplink transmitting functions (identical to, or based on the corresponding functions applied when FA is not enabled):

* The MPDU Header Creation function, as described in SECTION\_REF, shall be applied to each MPDU, with the following changes (see also Note 1):
	+ The value (TA) used in the Address 2 field of the MPDU Header shall be set to the active FA STA MAC for the Affiliated STA on the link, where FA STA MAC is generated as per 10.x.2.4.2 (Deriving other FA Parameters).
	+ The value used in the SN field of the MPDU Header shall be the obfuscated SN (OSN) value provided to this function by the SN / PN Anonymization function in 10.x.4.1.2 (TX SN / PN anonymization).
* The FCS Creation function, as described in SECTION\_REF, shall be applied to each MPDU.
* The Retransmission function, as described in SECTION\_REF, shall be applied to each MPDU.
* The A-MPDU Aggregation function, as described in SECTION\_REF, shall be applied to a set of MPDUs to which FA has been applied with a single set of FA Parameters (i.e., from a single FA epoch). See Note 2.
* The TXOP function, as described in SECTION\_REF, shall be applied to a set of A-MPDUs and MPDUs to which FA has been applied with a single set of FA Parameters (i.e., from a single FA epoch). See Note 2.
1. NOTE 1— The PN0, PN1, …, PN5 fields of the CCMP Header or GCMP Header contain the value of the obfuscated PN (OPN) generated by the SN / PN Anonymization function, see 10.x.4.1.2 (SN / PN anonymization (TX)).
2. NOTE 2— The uplink TX processing in the Affiliated STA can have ready for transmission and retransmission MPDUs from multiple FA epochs (i.e., multiple sets of FA parameters). All MPDUs in an uplink A-MPDU or uplink TXOP are transmitted by a single Affiliated STA. Consequently, if an uplink A-MPDU or uplink TXOP includes MPDU(s) from multiple FA epochs, then an eavesdropping attacker can determine that the corresponding sets of FA parameters are being used by a single Affiliated STA. E.g., an eavesdropping attacker can then determine that the multiple FA STA MAC addresses (used in the A-MPDU or TXOP uplink function) identify a single Affiliated STA. Additional FA requirements for these functions are applied to prevent this scenario.

10.x.5.2 Uplink receiving frame anonymization processing in Affiliated AP

1. When FA is enabled, then the receiving Affiliated AP of the AP MLD shall apply the following uplink receiving functions (identical to, or based on the corresponding functions applied when FA is not enabled):
* The TXOP function, as described in SECTION\_REF, shall be applied to the received TXOPs. See Note 1.
* The FCS Validation function, as described in SECTION\_REF, shall be applied to the received MPDUs. See Note 2.
* The MPDU Header Validation function, as described in SECTION\_REF, shall be applied to the received MPDUs.
* The A-MPDU De-aggregation function, as described in SECTION\_REF, shall be applied to the received A-MPDU. See Note 3.
* The Address 1 address filtering function, as described in SECTION\_REF, shall be applied to the received MPDUs.
* The Address 2 address filtering function, as described in SECTION\_REF, shall be applied to the received MPDUs where, instead of using the MAC Address established for the Affiliated STA at association, active FA STA MAC and/or retiring FA STA MAC values for the Affiliated STA, generated as per 10.x.2.4.2, shall be matched to the Address 2 field of the MPDU Header for the purposes of
	+ identifying the non-AP MLD which transmitted the MPDU, and
	+ identifying the SN Offset and PN offset (or equivalently, the FA epoch) for the AP MLD to use in receiving SN / PN De-anonymization, described in 10.x.4.2.1 (RX SN / PN de-anonymization).
* The MLD Lower MAC Block Ack Scoreboarding function, as described in SECTION\_REF, shall be applied to the received MPDUs (from an A-MPDU), using the obfuscated SN (OSN) values received the SN field in the place of (internal) SN values. See Notes 3 and 4.
* The MPDU distribution by TA function, as described in SECTION\_REF, shall be applied to the received MPDUs.
1. NOTE 1— When FA is enabled, the uplink TX processing in the Affiliated STA in 10.x.5.1 (Uplink TX processing in the Affiliated STA) includes requirements which ensure that each uplink TXOP contains MPDUs from a single FA epoch only (i.e., all MPDUs have FA applied using a single set of FA parameters).
2. NOTE 2— In the uplink FCS Validation function, FA changes Address 1 field, SN field and CCMP Header or GCMP Header PN field which are input to the FCS computation. Consequently, the resulting FCS can be changed by FA even though the FA computation algorithm remains unchanged.
3. NOTE 3— When FA is enabled, the TX processing in 10.x.5.1 (Uplink TX processing in the Affiliated STA) or 10.x.5.3 (Uplink TX processing in the Affiliated AP) includes requirements which ensure that each uplink A-MPDU contains MPDUs from a single FA epoch only (i.e., all MPDUs have FA applied using a single set of FA parameters).
4. NOTE 4— The (internal) SN values assigned by the SN Assignment function are assigned sequentially. If FA is not enabled, then the MLD lower MAC Block Ack Scoreboarding function is applied per-TID to the set of received values in the SN field, which are the sequential assigned (internal) SN values for each TID. If FA is enabled, then the MPDUs in an A-MPDU are from the same FA epoch (i.e., the MPDU are processed using the same set of FA parameters) as noted in Note 3, and so the obfuscated SN (OSN) values are computed by the SN / PN anonymization function in 10.4.1.2 (TX SN / PN anonymization) using the same set of per-TID SN Offset values. This means that, for a given TID, the OSN values in the SN field of MPDUs in an A-MPDU are obtained by adding the same SN Offset to the sequential assigned (internal) SN values for each TID – that is, these OSN values are also sequential. Consequently, whether FA is enabled or not, the values in the SN field of MPDUs in an A-MPDU are sequential and the MLD lower MAC Block Ack Scoreboarding function works identically in both cases.

10.x.5.3 Downlink transmitting frame anonymization processing in Affiliated AP

When FA is enabled, then the transmitting Affiliated AP of the AP MLD shall apply the following downlink transmitting functions (identical to, or based on the corresponding functions applied when FA is not enabled):

* The MPDU Header Creation function, as described in SECTION\_REF, shall be applied to each MPDU, with the following changes (see also Note 1 in 10.x.5.1 (Uplink transmitting frame anonymization processing in the Affiliated STA)).
	+ The value (TA) used in the Address 1 field of the MPDU Header shall be set to the active FA STA MAC for the Affiliated STA on the link, where FA STA MAC is generated as per 10.x.2.4.2 (Deriving other FA Parameters).
	+ The value used in the SN field of the MPDU Header shall be the obfuscated SN (OSN) value provided to this function by the SN / PN Anonymization function in 10.x.4.1.2 (TX SN / PN anonymization).
* The FCS Creation function, as described in SECTION\_REF, shall be applied to each MPDU.
* The Retransmission function, as described in SECTION\_REF, shall be applied to each MPDU.
* The A-MPDU Aggregation function, as described in SECTION\_REF, shall be applied to a set of MPDUs to which FA has been applied with a single set of FA Parameters (i.e., from a single FA epoch). See Note 1.
* The TXOP function, as described in SECTION\_REF, shall be applied to the outgoing A-MPDUs and MPDUs. A downlink TXOP may including MPDUs with a single Affiliated STA as receiver with FA applied using distinct sets of FA parameters See Note 2.
1. NOTE 1— The downlink TX processing in the Affiliated AP of MPDUs for which a single Affiliated STA is the intended receiver, can have ready for transmission and retransmission MPDUs from multiple FA epochs (i.e., using multiple sets of FA parameters). All MPDUs in a downlink A-MPDU have a single Affiliated STA as the intended receiver. Consequently, if a downlink A-MPDU includes MPDU(s) from multiple FA epochs, then an eavesdropping attacker can determine that the corresponding sets of FA parameters are being used by a single Affiliated STA. E.g., an eavesdropping attacker can then determine that the multiple FA STA MAC addresses (used in A-MPDU downlink function) identify a single Affiliated STA. Additional FA requirements for this function prevents this scenario.
2. NOTE 2— The set of MPDUs in an uplink TXOP always have the same Affiliated STA as transmitter and always have the same Affiliated AP as receiver, which led to prohibit uplink TXOPs from including MPDUs from multiple FA epochs (i.e., using multiple sets of FA parameters); see Note 2 in 10.x.5.1 (Uplink transmitting frame anonymization processing in the Affiliated STA). On the other hand, while the set of MPDUs in a downlink TXOP always have the same Affiliated AP as transmitter, the MPDUs in a downlink TXOP can have distinct Affiliated STAs as receivers. Note FA obfuscates whether downlink MPDUs with distinct Address 1 values are:
* MPDUs with a single Affiliated STA as receiver with FA applied using distinct sets of FA parameters, from
* MPDUs with distinct Affiliated STAs as receivers.
1. Consequently, a downlink TXOP is not prohibited from including MPDUs with a single Affiliated STA as receiver with FA applied using distinct sets of FA parameters.

10.x.5.4 Downlink receiving frame anonymization in Affiliated STA

1. When FA is enabled, then the receiving Affiliated STA of the non-AP MLD shall apply the following downlink receiving functions (identical to, or based on the corresponding functions applied when FA is not enabled):
* The TXOP function, as described in SECTION\_REF, shall be applied to the received MPDUs.
* The FCS Validation function, as described in SECTION\_REF, shall be applied to the received MPDUs. See Note 2 in 10.x.5.2 (Uplink receiving frame anonymization processing in Affiliated AP).
* The MPDU Header Validation function, as described in SECTION\_REF, shall be applied to the received MPDUs.
* The A-MPDU De-aggregation function, as described in SECTION\_REF, shall be applied to the received A-MPDU. See Note 3 in 10.x.5.2 (Uplink receiving frame anonymization processing in Affiliated AP).
* The Address 1 address filtering function, as described in SECTION\_REF, shall be applied to the received MPDUs where, instead of using the MAC Address established for the Affiliated STA at association, active FA STA MAC and/or retiring FA STA MAC values for the Affiliated STA, generated as per 10.x.2.4.2, shall be matched to the Address 1 field of the MPDU Header for the purposes of identifying the SN Offset value and PN offset value (or equivalently, the FA epoch) for the non-AP MLD to use in SN / PN De-anonymization, described in 10.x.4.2.1 (RX SN / PN de-anonymization).
* The MLD Lower MAC Block Ack Scoreboarding function, as described in SECTION\_REF, shall be applied to the received MPDUs, using the obfuscated SN (OSN) values in the SN field. See Notes 3 and 4 in 10.x.5.2 (Uplink receiving frame anonymization processing in Affiliated AP).

10.x.5.5 Retransmission frame anonymization processing

This clause will explain what happens during the transition to a new FA epoch, where the old FA epoch is “retiring” (i.e., only retransmissions from the old epoch). Work in progress.

10.x.6 Frame Anonymization processing for beacon frames

The processing of beacon frames is identical for with FA disabled and FA enabled, with the following differences.

* When the Affiliated AP ???has a frame for the non-AP MLD, ??? then then Affiliated AP forms the TIM in the beacon as described in SECTION\_REF, where:
	+ If FA is not enabled for the non-AP MLD, then Affiliated AP forms the TIM using the AID assigned during association of the non-AP MLD and the AP MLD.
	+ If FA is enabled for the non-AP MLD, then the Affiliated AP forms the TIM using the FA AID of the active FA epoch for the non-AP MLD.
* The Affiliated STA process the TIM in each received beacon frame as described in SECTION\_REF, where:
	+ If FA is not enabled for the non-AP MLD, then the Affiliated STA processes the TIM using the AID assigned during association of the non-AP MLD and the AP MLD.
	+ If FA is enabled for the non-AP MLD, then the Affiliated STA processes the TIM using the FA AID of the active FA epoch for the non-AP MLD.

10.x.7 Frame Anonymization processing for trigger frames

1. The difference between the processing of trigger frames with FA enabled and the processing of trigger frames with FA disabled is as follows.
* When the Affiliated AP includes User Info field for the non-AP MLD in a trigger frame, in a then then Affiliated AP forms the User Info field as described in SECTION\_REF, where:
	+ If FA is not enabled for the non-AP MLD, then Affiliated AP sets the AID12 field to the value of AID assigned during association of the non-AP MLD and the AP MLD
	+ If FA is enabled for the non-AP MLD, then the Affiliated AP sets the AID12 field to the value of FA AID of the active FA epoch for the non-AP MLD.
* The Affiliated STA processes User Info fields in a received trigger frame as described in SECTION\_REF, where:
	+ If FA is not enabled for the non-AP MLD, then the Affiliated STA processes the User Info field(s) using the AID assigned during association of the non-AP MLD and the AP MLD.
	+ If FA is enabled for the non-AP MLD, then the Affiliated STA processes the User Info field(s) using the FA AID of the active FA epoch for the non-AP MLD.

***TGbi editor: Add new row to Table 9-79 (Category values) in clause 9.4.1.11 (Action field) as follows:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Code | Meaning | See subclause | Robust | Group addressed privacy |
| <ANA> | Protected Enhanced Data Privacy | 9.6.x Protected Enhanced Data Privacy Action frame details | Yes | No |

***TGbi editor: Add new a subclause 9.4.2.x (FA Action Frame details) under clause 9.4.2 as follows:***

***TGbi editor: Add new a subclause 9.6.x (Protected FA Action Frame details) under clause 9.6 as follows:***

9.6.x Protected Enhanced Data Privacy Action frame details

* + - * 1. 9.6.x.1 Protected Enhanced Data Privacy Action field

The Protected Enhanced Data Privacy Action field, in the one octet immediately after the category field, differentiates between various Protected Enhanced Data Privacy Action frames as shown in the Table 9-aaa (Protected Enhanced Data Privacy Action Field values).

**Table 9-aaa Protected** **Enhanced Data Privacy Action Field values**

|  |  |
| --- | --- |
| **Value** | **Meaning** |
| 0 | FAPU Request frame |
| 1 | FAPU Response frame  |
| 2 | FAPU Push frame |
| 3-255 | Reserved |

* + - * 1. 9.6.x.2 FAPU Request frame format

The FAPU Request frame is an Action frame of category Protected FA. The Action field of an FAPU Request frame contains the information shown in Table 9-ccc (FAPU Request frame format).

**Table 9-ccc FAPU Request Action frame format**

|  |  |
| --- | --- |
| **Order** | **Information** |
| 0 | Category |
| 1 | Protected FA Action |

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

The Protected FA Action field is defined in 9.6.x.1 (Protected Enhanced Data Privacy Action field).

A non-AP MLD sends an FAPU Request frame to an AP MLD to request a new FA epoch for the non-AP MLD.

* + - * 1. 9.6.x.3 FAPU Response frame format

The FAPU Response frame is a individually addressed Action frame of category Protected Enhanced Data Privacy. The Action field of an FAPU Response frame contains the information shown in Table 9-ddd (FAPU Response frame format).

**Table 9-ddd FAPU Response Action field format**

|  |  |
| --- | --- |
| **Order** | **Information** |
| 0 | Category |
| 1 | Protected FA Action |
| 2 | Status Code |
| 3 | FA Epoch Start TSF |
| 4 | FA AID |

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

The Protected FA Action field is defined in 9.6.x.1 (Protected Enhanced Data Privacy Action field).

The Dialog Counter field is defined in 9.4.1.12 (Dialog Counter field).

The Status Code field is defined in 9.4.1.9 (Status Code field).

The FA Epoch Start TSF element uses the Timestamp field format defined in 9.4.1.10 (Timestamp field). The value in this field indicates the intended TSF time for which the non-AP MLD and AP MLD will transition to new FA parameters. Where this value in this element is less than the TSF time when the FAPU Response frame is transmitted, then the transition will occur at the end of the TXOP in which the FAPU Response frame is transmitted.

The FA AID element has AID format defined in 9.4.1.8 (AID). The FA AID element is the AID identifying the non-AP MLD during the FA Epoch.

The values of FA Epoch Start TSF element and FA AID element in the are combined with KDK to derive other FA parameters for the FA epoch, as described in 10.x.2.4.2 (Deriving other FA parameters).

* + - * 1. 9.6.x.4 FAPU Push frame format

The FAPU Push frame is a individually addressed Action frame of category Protected Enhanced Data Privacy. The Action field of an FAPU Push frame contains the information shown in Table 9-ddd (FAPU Response frame format).

**Table 9-ddd FAPU Push Action field format**

|  |  |
| --- | --- |
| **Order** | **Information** |
| 0 | Category |
| 1 | Protected FA Action |
| 2 | FA Epoch Start TSF |
| 3 | FA AID |

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

The Protected FA Action field is defined in 9.6.x.1 (Protected Enhanced Data Privacy Action field).

The FA Epoch Start TSF element is defined in 9.6.x.4 (FAPU Response frame format).

The FA AID element is defined in 9.6.x.4 (FAPU Response frame format).