IEEE P802.11
Wireless LANs

|  |
| --- |
| 11bi D0.4 CR for Miscellaneous CIDs |
| Date: 2024-09-30 |
| Author(s): |
| Name | Affiliation | Address | Phone | Email |
| Po-Kai Huang | Intel |  |  | po-kai.huang@intel.com |

Abstract

This submission proposes resolutions for the following CIDs:

1227, 1229, 1287, 1427

Revisions:

* Rev 0: Initial version of the document.
* Rev 1: Revision based on the comments during the teleconference call
* Rev 2: Revise 1427

Interpretation of a Motion to Adopt

A motion to approve this submission means that the editing instructions and any changed or added material are actioned in the TGbi D0.6 Draft. This introduction is not part of the adopted material.

***Editing instructions formatted like this are intended to be copied into the TGbi D0.6 Draft. (i.e. they are instructions to the 802.11 editor on how to merge the text with the baseline documents). TGbi Editor: Editing instructions preceded by “TGbi Editor” are instructions to the TGbi editor to modify existing material in the TGbi draft. As a result of adopting the changes, the TGbi editor will execute the instructions rather than copy them to the TGbi Draft.***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CID** | **Commenter** | **Clause** | **P.L** | **Comment** | **Proposed Change** | **Resolution** |
| 1227 | Mark RISON | 9.4.2.23.3 | 37.37 | If a new AKM has been added the existing reserved range will change | As it says in the comment | Revised –New ANA requested has been submitted. Resrevied value will be updated once the ANA value is assigned. TGbi editor to make the changes of the reserved value for AKM suite based on the ANA assignment.  |
| 1229 | Mark RISON | 9.4.2.188 | 38.02 | TGm has been adamant that fields cannot be renamed | Do not rename the field in the element | Rejected – Have checked with Emily and Robert to understand reasons of no name change and with name change.Reason for no name change: * The name maybe connected to upper layer operation and name change will create confusion
* The name maybe used by driver implementation and name change create confusion.

Reason for name change:* Keeping the name creates confusion on protocol usage

By considering above, the name change from “FILS Nonce” to “Nonce” has more benefits. There has been similar name change for another element from 11az, which changes “FILS wrapped data” to “Wrapped Data”. In that case, there is no confusion to upper layer and the name change does no create confusion to driver implementation. The name change does help protocol usage to clarify that Wrapped Data element can be used outside of FILS. The change from “FILS Nonce” to “Nonce” follows basically the same reasoning.We also provide details below to clarify that all the 30 instances of “FILS Nonce” in Revme D7.0 have been properly modified in 11bi D0.6.(6 instances) P26L43, P151L26, P436L17, P438L17, P439L56, P441 L60 are related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8. (2 instances) P770L42 are fixed by 11bi D0.6 P29L34.(1 instance) P773L24 is fixed by 11bi D0.6 P30L33(1 instance) P773L36 is fixed by 11bi D0.6 P30L47(1 instance) P773L59 is fixed by 11bi D0.6 P31L15(1 instance) P774L16 is fixed by 11bi D0.6 P31L35(1 instance) P774L36 is fixed by 11bi D0.6 P31L60(1 instance) P774L51 is fixed by 11bi D0.6 P32L19(2 instances) P895L51 one is fixed by 11bi D0.6 P38L28, another one are related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8(1 instances) P1325L33 is fixed by 11bi D0.6 P46L8(6 instances) Inside 9.4.2.188 FILS Nonce element are fixed by 11bi D0.6 P46L11-L28. Note that reference by framemaker to Figure 9-767—FILS Nonce element format will have changed once the name of the figure in framemaker is changed. (1 instance) P3148L63 is fixed by 11bi D0.6 P78L46(1 instance) P3148L64 is related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8(2 instances) P3151L58 one is fixed by 11bi D0.6 P79L06 another one is related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8(2 instances) P3152L60 one is fixed by 11bi D0.6 P79L26 another one is related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8(1 instance) P5193L26 is related to reference to 9.4.2.188 FILS Nonce element and will be fixed by 11bi D0.6 name change in clause 9.4.2.188 P46L8 |
| 1287 | Mark RISON |   | 0.00 | "protected action frame" should be "protected Action frame" | As it says in the comment | Revised – All instances of “protected action frame” has been fixed as “protected Action frame”.No further change is required.  |
| 1427 | Mark RISON | 12.14 | 0.00 | I don't think introducing vague terminology like "authentication originator/responder" is a good idea. Stick to Authenticator/Supplicant and AP/non-AP STA | As it says in the comment | Revised – in 802.1X, we have AP and STA, and authenticator is part of the AP and supplicant is part of the STA. Eventually EAPOL-PDU are all delivered to authenticator or supplicant, but the process to package them in frame defined by 802.11 is done by AP or STA. A citation is provided below, where STA will deliver the 802.1X message to authenticator and supplicant. Now, we can see that what is described in 12.14.4 are about “construct an authentication frame”, “receiving the authentication frame”, “extract EAPOL PDU”. This will be done by STA or MLD rather than the authenticator or supplicant. On the other hand, authentication originator and authentication responder may still be confused for authenticator and supplicant, we propose to simply use originator and responder to generalize STA or MLD and describe both cases in one shot.***12.6.8 RSNA establishment in an infrastructure BSS(#1084)******12.6.8.1 General****IEEE 802.1X authentication can be initiated by any one of the following mechanisms:**— If a STA negotiates to use IEEE 802.1X authentication during (re)association, the STA’s**management entity may respond to the MLME-ASSOCIATE.confirm (or indication) or MLMEREASSOCIATE.**confirm (or indication) primitive by requesting the Supplicant (or Authenticator)**to initiate IEEE 802.1X authentication. Thus, in this case, authentication is driven by the STA’s decision to associate and the AP’s decision to accept the association.**— If a STA’s MLME-SCAN.confirm primitive finds another AP within the ESS of which the STA is a**member, a STA may signal its Supplicant to use (#3469)IEEE Std 802.1X-2020 to preauthenticate**with that AP.**NOTE 2—A BSS transitioning STA’s IEEE 802.1X Supplicant can initiate preauthentication by sending**an EAPOL-Start PDU (in one or more EAPOL-Start frames) via its old AP, through the DS, to a new AP.**— If a STA receives an IEEE 802.1X message, it delivers this to its Supplicant or Authenticator, which**may initiate a new IEEE 802.1X authentication.*TGbi editor to make the changes shown in 11-24/1679r2 under all headings that include CID 1427 |

**Discussion:**

None

**Proposal: (#1427)**

* IEEE 802.1X authentication utilizing Authentication frames

If an AP sets the IEEE 802.1X Authentication Utilizing Authentication Frame Support field in the RSNXE that it transmits to 1, then a non-AP STA (originator) with dot11EDPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated equal to true(#1426) may signal its Supplicant to authenticate with the AP (responder) using IEEE Std 802.1X-2020 utilizing Authentication frames.

If any AP affiliated with an AP MLD sets the IEEE 802.1X Authentication Utilizing Authentication Frame Support field in the RSNXE that it transmits to 1, then a non-AP MLD (originator) with dot11EDPIEEE8021XAuthenticationUtilizingAuthenticationFrameActivated equal to true(#1426) may signal its Supplicant to authenticate with the AP MLD (responder) using IEEE Std 802.1X-2020 utilizing Authentication frames by transmitting the Authentication frames to the AP through a non-AP STA affiliated with the non-AP MLD.

When the originator is non-AP MLD and the responder is AP MLD, the RA field of an Authentication frame in response to an Authentication frame from the peer shall be set to the TA field of the Authentication frame from the peer.

If an originator chooses to initiate IEEE(#1181) 802.1X authentication utilizing Authentication frames, it first selects an(#1428) IEEE(#1181) 802.1X AKM that is supported by the responder.

The originator then constructs the first Authentication frame of the exchange as follows:

* Authentication Algorithm Number field is set to <ANA> (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 1.(#1436)
* The Encapsulation field carries an(#1429) EAPOL PDU.
* Include(#1430) the AKM Suite Selector element indicating the selected IEEE(#1181) 802.1X AKM.

The originator sends the first Authentication frame to the responder.

Upon receiving the first Authentication frame, the responder:

* Validates that the AKM indicated in AKM Suite Selector element is an IEEE(#1181) 802.1X AKM.(#1431)
* Validates that the selected IEEE(#1181) 802.1X AKM indicated in AKM Suite Selector element is supported. Otherwise processing status is set to STATUS\_INVALID\_AKMP.
* Extracts an(#1423) EAPOL PDU from the Encapsulation field, and processes it.(#1433)

The responder then constructs the second Authentication frame of the exchange as follows:

* Authentication Algorithm Number field is set to <ANA> (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 2.
* Status Code field(#1435) indicates the processing status.
* (#1436)The Encapsulation Length field(#1210) indicates 0 if the status is set to STATUS\_INVALID\_AKMP.
* The Encapsulation(#1429) field (if present) carries an(#1434) EAPOL PDU.
* Includes the AKM Suite Selector element indicating the same(#1437) IEEE(#1181) 802.1X AKM indicated in the first Authentication frame.

Once the processing is complete, the responder sends the second Authentication frame to the originator. If the processing status returned in the frame was not SUCCESS, the responder shall terminate the authentication.

Upon receiving the second Authentication frame, the originator:

* Validates that the(#1438) AKM indicated in AKM Suite Selector element is the same as the one indicated in the first Authentication frame. Otherwise processing status is set to STATUS\_INVALID\_AKMP.
* Extracts an(#1434) EAPOL PDU from the Encapsulation field, and processes it.(#1433)

The originator then constructs the third Authentication of the exchange as follows:

* Authentication Algorithm Number field is set to <ANA> (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to 3.
* Status Code field(#1435) indicates the processing status.
* (#1436)The Encapsulation Length(#1210) field indicates 0 if the status is set to STATUS\_INVALID\_AKMP.

Once the processing is complete, the originator sends the third Authentication frame to the responder. If the processing status returned in the frame was not SUCCESS, the originator shall terminate the authentication.

Upon receiving the Authentication frame with Authentication Transaction Sequence Number field set to X, where X is larger than or equal to 3, the originator or the responder:

* Extracts an(#1434) EAPOL PDU from the Encapsulation field, and processes it.(#1433)

The originator or the responder then constructs the Authentication frame of the exchange in response to the Authentication frame with Authentication Transaction Sequence Number field set to X, where X is larger than or equal to 3,(#1440) as follows:

* Authentication Algorithm Number field is set to <ANA> (IEEE 802.1X authentication).
* Authentication Transaction Sequence Number field is set to X+1.
* Status Code field(#1435) indicates the processing status.(#1436)
* The Encapsulation(#1429) field (if present) carries an(#1434) EAPOL PDU.

NOTE—The number of Authentication frame exchanges depends on the EAP method in use. Hence, X is a value as defined by the EAP method.(#1440)

Once the processing is complete, the originator or the responder sends the Authentication frame to its peer (if needed by the EAP method). If the processing status returned in the frame was not SUCCESS, the originator or the responder shall terminate the authentication.

* IEEE(#Ed) 802.1X

If an originator or a responder defined in 12.14.4 (IEEE 802.1X authentication utilizing Authentication frames) sets the (Re)Association Frame Encryption Support(#1488) field in the RSNXE to 1, then the originator or the responder supports the additional rules defined in this subclause when performing IEEE(#Ed) 802.1X Authentication frame exchange.

An originator that sets the (Re)Association Frame Encryption(#1488) Support field in the RSNXE to 1, has the (#1484)SME to act as the Supplicant, receives the RSNXE from the responder with the (Re)Association Frame Encryption(#1488) Support field(#1474) set to 1, and intends to continue association after authentication shall:

* Include a Nonce element in the first Authentication frame to indicate SNonce.
* Include an RSNE in the first Authentication frame to indicate AKM and pairwise cipher suite. Version field shall be set to 1. Pairwise Cipher Suite Count field shall be set to 1. AKM Suite Count field shall be set to 1. PMKID Count field, if present, shall be set to 0. All other fields shall be as specified in 9.4.2.23 (RSNE) and 12.6.3 (RSNA policy selection in an infrastructure BSS).(#1154)
* Not include an AKM Suite Selector element.
* Include an RSNXE in the first Authentication frame.
* Include a Diffie-Hellman Parameter element in the first Authentication frame.
* Select a(#1476) finite cyclic group in the Diffie-Hellman Parameter element from the dot11RSNAConfigDLCGroupTable that is at least of the security strength provided by the AKM and cipher suites.
* With the chosen finite cyclic group, generate an ephemeral (random) private key, use the selected group's scalar operation (see 12.4.4.1 (General)) with the private key to generate its ephemeral public key, and indicate the ephemeral public key in the Diffie-Hellman Parameter element.

Otherwise, an originator shall not include a Diffie-Hellman Parameter element or an RSNE or an RSNXE or a Nonce element in the first Authentication frame for IEEE(#Ed) 802.1X authentication.

For the purpose of interoperability, an authenticator or a supplicant shall support group 19, an ECC group defined over a 256-bit prime order field.

A responder that sets the (Re)Association Frame Encryption(#1488) Support field in the RSNXE to 1, has the (#1484)SME to act as the Authenticator, and receives the first Authentication frame with a Nonce element, RSNE, RSNXE, and a Diffie-Hellman Parameter element shall:

* Verify that the AKM indicated in the RSNE rather than AKM suite selector element as defined in 12.4.4 (IEEE 802.1X authentication utilizing Authentication frames) is supported. Otherwise, the responder shall reject message 1 with status code set to STATUS\_INVALID\_AKMP.
* Verify that the pairwise cipher indicated in the RSNE is supported. Otherwise, the responder shall reject message 1 with status code set to STATUS\_INVALID\_PAIRWISE\_CIPHER.
* Validate that the finite cyclic group indicated in the Diffie-Hellman Parameter element in the first Authentication frame is supported (present in dot11RSNAConfigDLCGroupTable). Otherwise, the responder shall reject message 1 with status code set to UNSUPPORTED\_FINITE\_CYCLIC\_GROUP.
* Verify the public key indicated in the Diffie-Hellman Parameter element in message 1 as specified in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the responder shall reject the first Authentication frame with status code set to INVALID\_PUBLIC\_KEY.
* If the first Authentication frame is not rejected, store the indicated SNonce and generate an ephemeral (random) private key with the chosen finite cyclic group and use the selected group's scalar operation with the private key to generate its ephemeral public key. Perform the group's scalar-op (see 12.4.4.1 (General)) with the originator's ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.
* Include an RSNE in the second Authentication frame to indicate the AKM and pairwise cipher indicated in the first Authentication frame.
* Not include an AKM Suite Selector element in the second Authentication frame.
* Include a Diffie-Hellman Parameter element in the second Authentication frame.
* Indicate chosen finite cyclic group in the Diffie-Hellman Parameter element of the second Authentication frame, which is the same as the finite cyclic group in the Diffie-Hellman Parameter element of the first Authentication frame.
* Indicate its ephemeral public key in the Diffie-Hellman Parameter element of the second Authentication frame.
* Include a Nonce element in the second Authentication frame to indicate ANonce.

Otherwise, a responder shall not include a Diffie-Hellman Parameter element or a Nonce element or an RSNE in the second Authentication frame for IEEE(#Ed) 802.1X authentication.

After receiving the second Authentication frame with the status code set to SUCCESS, an originator shall:

* If the originator includes a Diffie-Hellman Parameter element in the first Authentication frame, validate that there is a Diffie-Hellman Parameter element and an RSNE included in the second Authentication frame and there is no AKM suite selector element in the second Authentication frame. If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* If the originator does not include a Diffie-Hellman Parameter element in the first Authentication frame, validate that there is no Diffie-Hellman Parameter element and no RSNE included in the second Authentication frame. If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* If the originator includes a Diffie-Hellman Parameter element in the first Authentication frame, validate that the finite cyclic group indicated in the Diffie-Hellman Parameter element in the second Authentication frame is the same as the finite cyclic group indicated in the Diffie-Hellman Parameter element in the first Authentication frame, validate that the pairwise cipher suite and the AKM indicated in the second Authentication frame are the same as the pairwise cipher suite and the AKM indicated in the first Authentication frame. The validation of AKM is based on the AKM indication in RSNE rather than AKM suite selector element as defined 12.14.4 (IEEE 802.1X authentication utilizing Authentication frames). If the validation fails, the originator shall discard the frame and terminate further protocol processing.
* Verify the public key indicated in the Diffie-Hellman Parameter element in the second Authentication frame as specified in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the originator shall discard the frame and terminate further protocol processing.
* If the second Authentication frame is not discarded, store the indicated ANonce, perform the group's scalar-op (see 12.4.4.1 (General)) with the originator's ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.

Before sending the Authentication frame carrying EAP Success, a responder shall:

* Derive PTK with DHss as defined in 12.7.1.3 (Pairwise key hierarchy).
* Irretrievably delete the shared secret, DHss, upon completion of PTK generation.(#1483)

After receiving the Authentication frame carrying EAP Success, an originator shall:

* Derive PTK with DHss as defined in 12.7.1.3 (Pairwise key hierarchy).
* Irretrievably delete the shared secret, DHss, upon completion of PTK generation.(#1483)

The originator and the responder then continue the operation as defined in 12.14.5 ((Re)Association Request/Response Frame Encryption(#1488)) with the following additional rules:

* Responder shall verify that the RSNE in the (Re)Association Request frame is identical to(#1154) the RSNE included in the first Authentication frame. Responder shall also verify that the RSNXE(#1153) in the (Re)Association Request is identical to(#1154) the RSNXE included in the first Authentication frame. If the validation fails, the responder shall reject the association.
* Originator shall verify that the RSNE in the (Re)Association Response frame is the same as the RSNE included in the second Authentication frame. If the validation fails, the originator shall disassociate.