**IEEE P802.11  
Wireless LANs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ML-KEM in PASN | | | | |
| **Date**: June. 24, 2025. | | | | |
| **Author(s):** | | | | |
| **Name** | **Affiliation** | **Address** | **Phone** | **email** |
| Chu-Meng Wang |  |  |  | wang.chumeng@zte.com.cn |
| Jay Yang | ZTE |  |  | yang.zhijie@zte.com.cn |
| Yan Li |  |  |  |
| Yun Li |  |  |  |
| Yurong Qian |  |  |  |
| Qisheng Huang |  |  |  |
| Zisheng Wang |  |  |  |
| Bo Cao |  |  |  |
| Chun Huang |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Abstract**

**Revisions:**

Rev 0: Initial version of the document.

Rev 1: Update the DCN information.

**Proposed Texts:**

***Change the following paragraphs as shown:***

**4.5.4.2 Authentication**

***Change the third, fourth, sixth paragraph as follows:***

IEEE Std 802.11 defines the following IEEE 802.11 authentication methods:

— Open System authentication admits any STA to the DS.

— FT authentication relies on keys derived during the initial mobility domain association to authenticate the stations as defined in Clause 13 (Fast BSS transition).

— SAE authentication uses finite field cryptography to prove knowledge of a shared password.

— IEEE 802.1X authentication uses EAP to authenticate STAs and the AS with one another.

— FILS authentication allows for faster connection to the network for FILS non-AP STAs by providing authentication, association, and key confirmation information in an efficient number of frame exchanges (see 4.10.3.6 (AKM operations using FILS authentication)).

— PASN, EDPKE and PQC PASN authentication allows for the protection of Management frames without association by establishing a PTKSA using authentication frames. The IEEE 802.11 authentication mechanism also allows definition of new authentication methods, or any combination of these authentication methods.

An RSNA might support one or more of the following authentication methods: SAE authentication, IEEE

802.1X authentication, FILS authentication, or PASN authentication, or EDPKE authentication. An RSNA also supports authentication based on IEEE Std 802.1X-2020, or preshared keys (PSKs) after Open System authentication. This standard does not specify an EAP method that is mandatory to implement. See 12.6.4 (RSNA policy selection in an IBSS) for a description of the IEEE 802.1X authentication and PSK usage within an IEEE 802.11 IBSS.

***Add the last paragraph as follows:***

PASN authentication, EDPKE authentication and PQC PASN authentication is used in an RSN for an infrastructure BSS when it is based on a PMKSA established by another RSN authentication protocol. Otherwise, it does not guarantee mutual authentication, and can be used as a non-RSN protocol in an infrastructure BSS.

6.5.5 Authenticate

***TGbt editor: Modify 6.5.5.2, 6.5.5.3, 6.5.5.4, and 6.5.5.5 as follows***

6.5.5.2 MLME-AUTHENTICATE.request

##### 6.5.5.2.1 Function

This primitive requests authentication with a specified peer MAC entity.

6.5.5.2.2 Semantics of the service primitive

***TGbt editor: change the primitive parameters as follows in the .request:***

The primitive parameters are as follows:

MLME-AUTHENTICATE.request (

………………………….

Content of FILS Authentication frame,

Content of PASN Authentication frame,

Content of EDPKE Authentication frame,

Content of PQC PASN Authentication frame,

VendorSpecificInfo

)

***TGbt editor: insert a new row in the following unnumbered table:***

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| Content of EDPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2  (EDPKE Frame  Construction and  Processing), 9.4.2.23  (RSNE), 9.4.2.240  (RSNXE), 9.4.2.186  (Wrapped Data element),  9.4.2.305 (PASN Parameters  element), 9.4.2.47 (Timeout  Interval element) | The set of elements and fields to be included in EDPKE Authentication frames. Present if Authentication Type  indicates EDPKE and  dot11EDPKEActivated is true, otherwise not present. | |
| Content of PQC PASN Authentication frame | Sequence of elements and fields | As defined in [12.XX.3.2](#H12o13o3o2) (PQC PASN Frame Construction and Processing), 9.4.2.23(RSNE), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.48 (Timeout Interval element) | The set of elements and fields to be included in PQC PASN Authentication frames. Present if Authentication Type indicates PQC PASN authentication and dot11PQC PASN Activated is true, otherwise not present. | |
| VendorSpecificInfo | A set of  elements | As defined in 9.4.2.25  (Vendor Specific element) | Zero or more elements. |

6.5.5.3 MLME-AUTHENTICATE.confirm

##### 6.5.5.3.1 Function

This primitive reports the results of an authentication attempt with a specified peer MAC entity.

6.5.5.3.2 Semantics of the service primitive

***TGbt editor: change the primitive parameters as follows in the .confirm:***

MLME-AUTHENTICATE.confirm(

………………………….

Content of FILS Authentication frame,

Content of PASN Authentication frame,

Content of EDPKE Authentication frame,

Content of PQC PASN Authentication frame,

VendorSpecificInfo

)

***TGbt editor: insert the following entry into the unnumbered table in this subclause maintaining the primitive order above:***

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| … | … | … | … |
| AuthenticationType | Enumeration | OPEN\_SYSTEM,  SHARED\_KEY  FAST\_BSS\_TRANSITION,  SAE, FILS\_SHARED  KEY\_WITHOUT\_PFS,  FILS\_SHARED\_KEY\_WI  TH\_PFS,  FILS\_PUBLIC\_KEY,  PASN | Specifies the type of authentication  algorithm that was used during the  authentication process. This value  matches the AuthenticationType  parameter specified in the corresponding  MLME-AUTHENTICATE.request  primitive. |
| … | … | … | … |
| Content of EDPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2  (EDPKE Frame  Construction and  Processing), 9.4.2.23  (RSNE), 9.4.2.240  (RSNXE), 9.4.2.186  (Wrapped Data element),  9.4.2.305 (PASN Parameters  element), 9.4.2.47 (Timeout  Interval element) | The set of elements and fields to be included in EDPKE Authentication frames. Present if Authentication Type  indicates EDPKE and  dot11EDPKEActivated is true, otherwise not present. |
| Content of PQC PASN Authentication frame | Sequence of elements and fields | As defined in [12.XX.x.2.2](#H12o13o3o2) PQC PASN Frame Construction and Processing. 9.4.2.23(RSNE), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.48 (Timeout Interval element) | The set of elements and fields to be included in PQC PASN Authentication frames. Present if Authentication Type indicates PQC PASN authentication and dot11PQCPASNActivated is true, otherwise not present. |

6.5.5.4 MLME-AUTHENTICATE.indication

6.5.5.4.2 Semantics of the service primitive

***TGbt editor: change the primitive parameters as follows in the .indication:***

MLME-AUTHENTICATE.indication(

………………………….

Content of FILS Authentication frame,

Content of PASN Authentication frame,

Content of EDPKE Authentication frame,

Content of PQC PASN Authentication frame,

VendorSpecificInfo

)

***TGbt editor: insert the following entry into the unnumbered table in this subclause maintaining the primitive order above:***

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| … | … | … | … |
| AuthenticationType | Enumeration | OPEN\_SYSTEM,  SHARED\_KEY  FAST\_BSS\_TRANSITION,  SAE, FILS\_SHARED  KEY\_WITHOUT\_PFS,  FILS\_SHARED\_KEY\_WI  TH\_PFS,  FILS\_PUBLIC\_KEY,  PASN | Specifies the type of authentication algorithm that was used during the authentication process. This value matches the AuthenticationType  parameter specified in the corresponding  MLME-AUTHENTICATE.request primitive. |
| … | … | … | … |
| Content of EDPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2  (EDPKE Frame  Construction and  Processing), 9.4.2.23  (RSNE), 9.4.2.240  (RSNXE), 9.4.2.186  (Wrapped Data element),  9.4.2.305 (PASN Parameters  element), 9.4.2.47 (Timeout  Interval element) | The set of elements and fields to be included in EDPKE Authentication frames. Present if Authentication Type  indicates EDPKE and  dot11EDPKEActivated is true, otherwise not present. |
| Content of PQC PASN Authentication frame | Sequence of elements and fields | As defined in [12.XX.3.2](#H12o13o3o2) (PQC PASN Frame Construction and Processing), 9.4.2.23(RSNE), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.XXX (PASN Parameters element), 9.4.2.48 (Timeout Interval element) | The set of elements and fields to be included in PQC PASN Authentication frames. Present if Authentication Type indicates PQC PASN authentication and dot11PQCPASNActivated is true, otherwise not present. |

6.5.5.5 MLME-AUTHENTICATE.response

6.5.5.5.2 Semantics of the service primitive

***Change the primitive parameters as follows in the .response***

MLME-AUTHENTICATE.response(

………………………….

Content of FILS Authentication frame,

Content of PASN Authentication frame,

Content of EDPKE Authentication frame,

Content of PQC PASN Authentication frame,

VendorSpecificInfo

)

***TGbt editor: insert the following entry into the unnumbered table in this subclause maintaining the primitive order above:***

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| … | … | … | … |
| AuthenticationType | Enumeration | OPEN\_SYSTEM,  SHARED\_KEY  FAST\_BSS\_TRANSITION,  SAE, FILS\_SHARED  KEY\_WITHOUT\_PFS,  FILS\_SHARED\_KEY\_WI  TH\_PFS,  FILS\_PUBLIC\_KEY,  PASN | Specifies the type of authentication  algorithm that was used during the authentication process. This value  matches the AuthenticationType  parameter specified in the corresponding  MLME-AUTHENTICATE.request primitive. |
| … | … | … | … |
| Content of EDPKE Authentication frame | Sequence of elements and fields | As defined in 12.16.9.3.2  (EDPKE Frame  Construction and  Processing), 9.4.2.23  (RSNE), 9.4.2.240  (RSNXE), 9.4.2.186  (Wrapped Data element),  9.4.2.305 (PASN Parameters  element), 9.4.2.47 (Timeout  Interval element) | The set of elements and fields to be included in EDPKE Authentication frames. Present if Authentication Type  indicates EDPKE and  dot11EDPKEActivated is true, otherwise not present. |
| Content of PQC PASN Authentication frame | Sequence of elements and fields | As defined in [12.XX.3.2](#H12o13o3o2) (PQC PASN Frame Construction and Processing), 9.4.2.23(RSNE), 9.4.2.240 (RSNXE), 9.4.2.186 (Wrapped Data element), 9.4.2.305 (PASN Parameters element), 9.4.2.48 (Timeout Interval element) | The set of elements and fields to be included in PQC PASN Authentication frames. Present if Authentication Type indicates PQC PASN authentication and dot11PQCPASNActivated is true, otherwise not present. |

9.3.3.11 Authentication frame format

***TGbt editor: insert the new rows at the end of table 9-41:***

1. Table 9-41—Presence of fields and elements in Authentication frames *(continued):*

|  |  |  |  |
| --- | --- | --- | --- |
| Authentication algorithm | **Authentication transaction sequence number** | **Status code** | **Presence of fields 4 onwards** |
| PQC PASN  authentication | 1 | Reserved | The RSNE is present.  The RSNXE is present if any subfield of the  Extended RSN Capabilities field in this element, except the Field Length subfield, is nonzero.  The PASN Parameters element is present.  The Timeout Interval element is optionally  present.  The Wrapped Data element is present if the  wrapped data format in the PASN Parameters  element is nonzero and not reserved |
| PQC PASN  authentication | 2 | Status | The RSNE is present and the PQC PASN  Parameters element is present if Status Code field is 0.  The RSNXE is present if any subfield of the  Extended RSN Capabilities field in this  element, except the Field Length subfield, is  nonzero.  The Timeout Interval element is optionally  present.  The Wrapped Data element is present if  wrapped data format in the PASN Parameters element is nonzero and not reserved and the Status Code field is 0.  The MIC element is present. |
| PQC PASN  authentication | 3 | Status | The PASN Parameters element is present if Status Code field is 0.  The Wrapped Data element is present if  wrapped data format in the PASN Parameters  element is nonzero and not reserved; and the  Status Code field is 0.  The MIC element is present. |

*TGbt editor: make the following change in subclause 9.4:*

9.4 Management and Extension frame body components

9.4.1 Fields that are not elements

9.4.1.1 Authentication Algorithm Number field

Authentication algorithm number = 6: FILS Public Key authentication

Authentication algorithm number = 7: PASN authentication

Authentication algorithm number = 8: IEEE 802.1X authentication

Authentication algorithm number = 9: EDPKE authentication

Authentication algorithm number = <ANA>: PQC PASN authentication

Authentication algorithm number = 65 535: vendor specific use

**9.4.2.305 PASN Parameters element**

The Control field, see Figure 9-1057 (PASN Parameters element Control Info field format(11az)), indicates the presence or absence of other fields in the PASN Parameters element; see Figure 9-1056 (PASN

Parameters element format(11az)).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Element ID | Length | Element ID  Extension | Control | Wrapped  Data  Format | Comeback  Info | Finite  Cyclic  Group  ID/ PQC Key Type | Ephemeral  Public Key  Length/ PQC Public Key Length/ PQC Ciphertext Length | Ephemeral  Public Key/PQC Public Key/ Ciphertext |
| Octets: | 1 | 1 | 1 | 1 | 1 | Variable | 0 or 2 | 0 or 1 or 2 | Variable |

**Figure 9-1056—PASN Parameters element format**

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | B0 | B1 | B2 | B3 | B4 B7 |
|  | Comeback Info Present | Group and Key Present | PQC Key Type Present | PQC Public Key present | Reserved |
| Bits: | 1 | 1 | 1 | 1 | 4 |

**Figure 9-1057—PASN Parameters element Control Info field format**

The Comeback Info Present subfield indicates if the Comeback Info field is included in the PASN Parameters element. The Group and Key Present subfield indicates if the PASN Parameters element includes the Finite Cyclic Group ID, the Ephemeral Public Key Length, and the Ephemeral Public Key fields.

The PQC Key Type Present subfield is set to 1 and PQC Public Key present is set to 0 indicate the PASN Parameters element includes the PQC Key Type, the Ciphertext Length, and the Ciphertext fields.

The PQC Key Type Present subfield is set to 1 and PQC Public Key present is set to 1 indicate the PASN Parameters element includes the PQC Key Type, the PQC Public Key Length, and the PQC Public Key fields.

Note: Group and Key Present and PQC Key Type Present can’t be set to 1 at the same time.

The Wrapped Data Format field indicates the format of data in the Wrapped Data element included along with the PASN Parameters element. The values defined for this format are:

— 0: No wrapped data

— 1: Fast BSS Transition Wrapped Data; see 12.13.6

— 2: FILS Shared Key authentication without PFS Wrapped Data; see 12.13.4

— 3: SAE Wrapped Data; see 12.13.5

— 4–255: Reserved

The Comeback Info field, present only when the corresponding Comeback Info Present subfield in the Control field is set to 1, is of variable length and is formatted as shown in Figure 9-788fd.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Comeback After | Cookie Length | Cookie |
| Octets: | 2 | 1 | Variable |

**Figure 9-788fd—Comeback Info field format**

The Comeback After subfield value is the time in TUs after which the non-AP STA is requested to retry the PASN authentication. The Comeback After subfield is set to 0 to indicate that the operation can be retried with a cookie of nonzero length in the Cookie subfield. The Comeback After subfield is reserved in PASN Authentication frames from a non-AP STA.

The Cookie Length subfield value is the length of the following Cookie subfield. If the Cookie Length subfield is 0, it indicates that there is no Cookie subfield.

The Cookie subfield contains a cookie, which is an opaque sequence of octets generated by an AP STA in an implementation dependent manner; see 12.13.9.

The Finite Cyclic Group ID, Ephemeral Public Key Length, and Ephemeral Public Key fields are present if the Group and Key Present subfield in the Control field is set to 1.

The Finite Cyclic Group ID field indicates the group used in PASN authentication. It has the same semantics as the field Finite Cyclic Group field (9.4.1.42).

The PQC Public Key Type is a 16-bit unsigned integer that maps an identifying number to a Encryption Key Type defining the algorithm. The following values are defined for PQC Encryption Key Type:

PQC Public Key Type = 0: ML-KEM-512

PQC Public Key Type = 1: ML-KEM-768

PQC Public Key Type = 2: ML-KEM-1024

PQC Public Key Type 3-255: Reserved

The Ephemeral Public Key Length field is set to the length in octets of the Ephemeral Public Key field.

The PQC Public Key Length is a 16-bit unsigned integer that is set to the length in octets of the PQC Public key field. Some PQC Encryption keys can be too large to fit in a single element and in many cases too large to fit in a single frame. Therefore, these elements will necessarily require fragmentation and reassembly.

The PQC Ciphertext Length is set to the length in octets of the Ciphertext field

The Ephemeral Public Key field contains the public key encoded using RFC 5480 conventions. Both compressed and uncompressed forms may be used.

The PQC Public Key field contains a public key from the indicated PQC Encryption Key Type whose length depends on the PQC Public Key Type.

The Ciphertext field contains a secret encrypted using an encryption key from a PQC key establishment algorithm.

**9.4.2.23 RSNE**

**9.4.2.23.3 AKM suites**

***Modify Table 9-190 (AKM suite selectors) as follows and update reserved suite type:***

9.4.2.24 RSNE

9.4.2.24.3 AKM suites

***TGbt editor: insert the following new row (PQC PASN) into Table 9-190 (AKM suite selectors):***

1. Table 9-190—AKM suite selectors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **OUI** | **Suite type** | **Meaning** | | | |
| **Authentication type** | **Key management type** | **Key derivation**  **type** | **Authentication**  **algorithm**  **numbers (see**  **9.4.1.1**  **(Authentication**  **Algorithm**  **Number**  **field))** |
| … |  |  |  |  |  |
| 00-0F-AC | 29 | EDPKE | EDPKE | EDPKE key  management  defined in  12.16.9  (Enhanced  Data Privacy  Key  Exchange) | Defined in 12.16.9.3  (Key establishment  with EDPKE  authentication) |
| 00-0F-AC | <ANA> | PQC PASN | PQC PASN | PQC PASN key management defined in 12.xx (PQC PASN) | Defined in 12.XX.3 (Key establishment with PQC PASN authentication) |

**9.4.2.240 RSNXE**

***TGbt editor: make the following change in Table 9-373 (Extended RSN Capabilities field)***.

**Table 9-373—Extended RSN Capabilities field**

|  |  |  |
| --- | --- | --- |
| **Bit** | **Information** | **Notes** |
| 16 | Device ID Support | A STA sets the Device ID Support field to 1 when dot11DeviceIDActivated is true to indicate that the device ID mechanism is supported. Otherwise, the STA sets the Device ID Support field to 0. |
| 17 | IRM Support | A STA sets the IRM Support field to 1 when dot11IRMActivated is true to indicate that the IRM mechanism is supported. Otherwise, the STA sets the IRM Support field to  0. |
| 18 | KEK In PASN | The field is set to 1 when dot11KEKPASNActivated is true to indicate support for deriving a KEK when using PASN or using PQC PASN. Otherwise, the field is set to 0. |

**9.4.2.186 Wrapped Data element**

***TGbt editor: change the third paragraph as follows:***

The Wrapped Data field is the data used by the FILS authentication algorithm (see 12.11 (Authentication for FILS)), PASN authentication algorithm (see 12.13 (Preassociation security negotiation)), EDPKE algorithm (see 12.16.9 (Enhanced Data Privacy Key Exchange)) and PQC PASN authentication algorithm (see 12.XX (PQC PASN)).

**11.3.4 Authentication and deauthentication**

**11.3.4.2 Authentication—originating STA or MLD**

***TGbt editor: change the second paragraph as follows:***

Upon receipt of an MLME-AUTHENTICATE.request primitive, the originating STA shall authenticate with the indicated STA using the following procedure:

b) The STA shall execute one of the following:

…

5) For PASN authentication, the authentication mechanism described in 12.13 (Preassociation security negotiation (11az)).

6) For EDPKE authentication, the authentication mechanism described in 12.16.9 (Enhanced Data Privacy Key Exchange).

7) For PQC PASN authentication, the authentication mechanism described in 12.XX (PQC PASN).

**11.3.4.3 Authentication—destination STA or MLD**

***TGbt editor: change the first paragraph as follows:***

Upon receipt of an Authentication frame with authentication transaction sequence number equal to 1, the destination STA shall authenticate with the originating STA using the following procedure:

g) Upon receipt of an MLME-AUTHENTICATE.response primitive, if the ResultCode is not SUCCESS, the MLME shall transmit an Authentication frame with the corresponding status code, as defined in 9.4.1.9 (Status Code field), and the state for the originating STA or MLD shall be left unchanged. The Authentication frame is constructed using the appropriate procedure in 12.3.3.2 (Open System authentication), 13.5 (FT protocol), 13.6 (FT resource request protocol), 12.13 (Preassociation security negotiation (11az)),12.16.9 (Enhanced Data Privacy Key Exchange) or 12.XX (PQC PASN).

12. Security

12.2 Framework

12.2.4 RSNA establishment

***TGbt editor:insert the following paragraph after “i)”:***

h) If an RSNA uses PASN authentication, an RSNA capable the STA establishes an RSNA as described in 12.13 (Preassociation security negotiation (11az)).

i) If an RSNA uses EDPKE authentication, an RSNA capable STA establishes an RSNA as described in 12.16.9 (Enhanced Data Privacy Key Exchange).

j) If an RSNA uses PQC PASN authentication, an RSNA capable STA establishes an RSNA as described in 12.XX (PQC PASN).

12.6 RSNA security association management

12.6.1 Security associations

12.6.1.1 Security association definitions

12.6.1.1.1 General

***Change the following sentence:***

***TGbt editor:modify 12.6.1.1.1 as follows:***

PTKSA: A result of a successful 4-way handshake, FT 4-way handshake, FT authentication sequence, FILS authentication, PASN authentication, EDPKE authentication, or PQC PASN authentication.

12.6.1.1.6 PTKSA

*Change 12.6.1.1.6 PTKSA as follows*

***TGbt editor: change first paragraph as follows:***

The PTKSA results from a successful 4-way handshake, FT 4-way handshake, FT protocol, FT resource request protocol, FILS authentication, PASN authentication, EDPKE authentication or PQC PASN authentication. This security association is also bidirectional. PTKSAs, except those established using PASN authentication, EDPKE authentication or PQC PASN authentication, have the same lifetime as the PMKSA or PMK-R1 security Association, whichever comes first. PTKSAs for PASN authentication, EDPKE authentication or PQC PASN authentication have a minimum of the lifetime of the PMKSA used and the timeout negotiated, if any, during PASN authentication or PQC PASN authentication. Because the PTKSA is tied to the PMKSA or to a PMK-R1 security association, it only has the additional information from the 4-way handshake, FT Protocol authentication, FILS authentication, PASN authentication, EDPKE authentication or PQC PASN authentication. There shall be only one PTKSA per key ID per band (see 12.6.20 (Multi-band RSNA)) or per MLD setup (see 35.3.5 (ML (re)setup)) with the same Supplicant and Authenticator MAC addresses.

***TGbt editor: please add the following new clause***

**12.XX PQC PASN**

**12.XX.1 General**

If dot11PQC PASN is true, then dot11KEKPASNActivated is set to true.

PQC PASN (PQC PASN) is an RSNA authentication protocol in all cases between two STAs where it relies on the existence of a PMKSA for an AKMP, termed base AKMP for PQC PASN. It is a non-RSNA protocol when there is no PMKSA and the corresponding base AKMP used with it. The protocol supports PTKSA establishment with and without mutual authentication.PQC PASN is an either RSNA or non-RSNA authentication protocol that uses the PASN procedures (see 12.13 (Preassociation security negotiation)) with the following differences:

—The three Authentication frames are exchanged by two PQC STAs.

— The three Authentication frames have the Authentication Algorithm Number field set to <ANA> (PQC Authentication).

**12.XX.2 Discovery of an PQC PASN capable AP**

An AP indicates it is capable of performing PQC PASN by including the PQC PASN AKMP as part of the RSNE included in Beacon frame or Probe Response frame. When PQC PASN AKMP is advertised, the AP shall also include at least one additional AKMP in the RSNE unless it allows PTKSA derivation without authentication using the PQC Public keys and the corresponding Ciphertext exchanged during PQC PASN authentication.

**12. XX.3 Key establishment with PQC PASN authentication**

**12. XX.3.1 Overview**

This subclause defines the procedures for establishing a PTKSA and the corresponding shared keys between the PQC PASN capable AP. The same procedures as specified in 12.13.1 (Overview) are used with the following differences:

— The three Authentication frames have the Authentication Algorithm Number field set to <ANA> (PQC PASN Authentication).

— PQC PASN AKMP is used instead of PASN AKMP.

— The RSNE indicates PQC PASN instead of PASN.

**12. XX.3.2 PQC PASN Frame Construction and Processing**

The same procedures as specified in 12.13.3.2 (PASN Frame Construction and Processing) are used with the following differences:

— The three Authentication frames have the Authentication Algorithm Number field set to <ANA> (PQC PASN Authentication).

— PQC PASN AKMP is used instead of PASN AKMP.

— The RSNE indicates PQC PASN instead of PASN.

— The PTK is generated as specified in 12.XX.3.3 (PTKSA derivation and MIC Computation with PQC PASN authentication) and 12.XX.3.4 (MIC Computation with PQC PASN authentication) respectively.

**12. XX.3.3 PTKSA derivation** **with PQC PASN authentication**

For PTKSA key derivation, the inputs to the PRF are the PMK of the PMKSA, a constant label, and a concatenation of non-AP STA’s MAC address, AP’s BSSID, and the PQC shared secret from the ephemeral exchange.

PTK = KDF-HASH-NNN (PMK, “PQC PASN PTK Derivation”, SPA || BSSID || PQCss)

where

PMK is the pairwise master key for the base AKMP if the AKMP is other than PASN AKMP; see 9.4.2.23.3 (AKM suites). Otherwise, if the base AKMP is PASN AKMP, that is, the PASN PTKSA is being setup without mutual authentication in a non-RSN, the PMK shall be set to the string “PMKz” padded with 28 0s.

NOTE—The PMK for the derivation can come from a cached PMKSA for the AKMP or from the PMKSA established with PASN by tunneling Wrapped Data or Authentication frames.

PQCss is the shared secret derived from the PQC PASN public key and ciphertext exchange encoded as an octet string (12.4.7.2.2 (Integer to octet string conversion)).

KDF-HASH-NNN is the key derivation function defined in 12.7.1.6.2 (Key derivation function (KDF)) using the hash algorithm defined for the base AKMP; see Table 9-190 (AKM suite selectors). When there is no base AKMP, the hash algorithm is selected based on the pairwise Cipher Suite provided in the RSNE provided by the AP in the second PASN frame. SHA-256 is used as the hash algorithm, except for the ciphers 00-0F-AC: 9 and 00-0F-AC:10 for which SHA-384 is used.

NNN is the Bits required for KCK, TK, and KDK depending on the pairwise cipher and whether a KDK is derived.

PTK is composed of the Key Confirmation Key (KCK), Temporal Key (TK), and the Key Derivation Key (KDK), which are derived as follows:

KCK = ExtractBits (PTK, 0, 256)

TK = ExtractBits (PTK, 256, TK\_Length\_Bits)

TK is the transient key whose length is the same as a key for the pairwise cipher in the RSNE provided by the AP in the second PASN frame. TK\_Length\_Bits is the TK\_bits in Table 12-8 (Cipher suite key lengths).

KDK = ExtractBits (PTK, 256 + TK\_Length\_Bits, KDK\_bits)

The KDK is of bit length KDK\_bits, which has the value 256 if a KDK is derived (see 12.7.1.3 (Pairwise key hierarchy)) or 0 otherwise.

KDK shall be derived if dot11SecureLTFImplemented is true and the peer STA has indicated Secure HE-LTF support capability in its advertised Extended Capabilities.

The Key ID in the PTKSA (see 12.6.1.1.6 (PTKSA)) resulting from PASN authentication shall be 0.

**12. XX.3.4 MIC Computation with PQC PASN authentication**

**12.XX.3.4.1 MIC computation for second PASN frame**

The same procedures as specified in 12.13.9.1 (MIC computation for second PASN frame) are used.

**12. XX.3.4.2 MIC computation for third PASN frame**

The same procedures as specified in 12.13.9.2 (MIC computation for third PASN frame) are used.

# Annex C

**(normative)**

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

## C. 3 MIB detail

***…***

***Modify “Dot11StationConfigEntry” as follows:***

Dot11StationConfigEntry ::=

SEQUENCE{

…….

dot11S1GOptionImplemented TruthValue,

dot11PASNActivated TruthValue,

dot11NoAuthPASNActivated TruthValue,

dot11EPDPKEActivated TruthValue,

dot11PQCPASNActivated TruthValue  
 }

***Insert the following object before the “End of.” as shown below:***

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

-- \* dot11StationConfig TABLE

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

*…….*

dot11PQCPASNActivated OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable. It is written by an external management entity or the SME. Changes take effect for the next MLME-START.request primitive or MLME JOIN.request primitive. This attribute indicates whether or not PQC PASN authentication is enabled."

DEFVAL {false}

::= { dot11StationConfigEntry <ANA> }

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

-- \* End of dot11StationConfig TABLE

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