802.11bi Draft Specification

|  |
| --- |
| Proposed spec texts for key creation using authentication frame for FT |
| Date: 2024-01-08 |
| Author(s): |
| Name | Affiliation | Address | Phone | Email |
| Po-Kai Huang | Intel |  |  | po-kai.huang@intel.com |
| Ido Ouzieli |  |  |  |  |
| Johannes Berg |  |  |  |  |
| Ilan Peer |  |  |  |  |
| Robert Stacey |  |  |  |  |

Abstract

This submission proposes spec text based on the following passed requirement.

* ***11bi shall define a mechanism for a CPE Client and CPE AP to establish keys from an Authentication exchange which can then be used to protect the (Re)Association Request/Response.***

Revision History:

* Rev 0: Initial version of the document
* Rev 1: Add tag for DHss. Highlight texts without DHss with green.

***Editing instructions formatted like this are intended to be copied into the TGbi 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 or insert 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.***

**Discussion:**

We use the two authentication frame exchange of FT to demonstrate how key can be created by reusing existing two authentication frame exchange of FT to encrypt the coming reassociation request/response frame.

* The first authentication frame will have FTO parameter. After receiving the parameters from FTO, FTR can then derive PTKSA with its own parameters and be ready.
* FTR then sends FTR parameter in second authentication frame. FTO then has parameter to derive PTKSA and send encrypted Reassociation Request frame using PMF.
* FTR will be ready to decrypt Reassociation Request frame because PTKSA is ready

As a result, existing number of message exchange can be preserved with the additional benefits of privacy.



There are also discussions to incorporate PFS mechanisms introduced in PASN to the key derivation. To accommodate the comment, we simply add Diffie-Hellman Parameter element in authentication frame 1 and authentication frame 2 to derive DHss and have it as part of the key derivation for PTKSA. Corresponding changes are tagged with <#DHss Tag>

**Proposed Texts:**

**TGbi Editor: *Instruction: Insert new rows in Table 9-363 in 9.4.2.241 RSNXE as shown below***

9.4.2.241 RSNXE

|  |
| --- |
| * **Extended RSN Capabilities field**
 |
| **Bit** | **Information** | **Notes** |
| <ANA> | Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support | An EDP STA sets the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield to 1 if dot11EDPEncryptionOfTheFrameBodyFieldOfTheReAssociation RequestResponseFrameSupportActivated is true. Otherwise, this subfield is set to 0. See 12.13.x (Encryption of the Frame Body Field of the (Re)Association Request/Response frame). |

**TGbi Editor: *Instruction: Insert 12.13.x Key derivation with authentication frame exchange as shown below***

**12.13 Client Privacy Enhancement**

**12.13.x Key derivation with Authentication frame exchange**

This subclause defines rules to derive a temporal key (TK) through Authentication frame exchange to encrypt the Frame Body field of the (Re)Association Request/Response frame.

12.13.x.1 FT

If an FTO or FTR (see 13 (Fast BSS transition)) sets the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield in the RSNXE to 1, then FTO or FTR supports the additional rules defined in this subclause.

<#DHss Tag>An FTO that sets the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield in the RSNXE to 1 and sees the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield in the RSNXE of the FTR set to 1 shall:

* Include a Diffie-Hellman Parameter element as defined in IETF RFC 8110 in the first message of the FT protocol (see 13.8 (FT authentication sequence)).
* Choose indicated 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.

<#DHss Tag>Otherwise, an FTO shall not include a Diffie-Hellman Parameter element in the first message of the FT protocol.

An FTR that sets the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield in the RSNXE to 1 and receives the first message with the Encryption of the Frame Body Field of the (Re)Association Request/Response Frame Support subfield in the RSNXE set to 1 shall:

* <#DHss Tag>Validate that finite cyclic group indicated in the Diffie-Hellman Parameter element in message 1 is supported (present in dot11RSNAConfigDLCGroupTable). Otherwise, the AP shall reject message 1 with status code set to UNSUPPORTED\_FINITE\_CYCLIC\_GROUP.
* <#DHss Tag>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 AP shall reject message 1 with status code set to INVALID\_PUBLIC\_KEY.
* <#DHss Tag>If the message 1 is not rejected, 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 FTO’s ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.
* <#DHss Tag> Append DHss at the end of context to function KDF-Hash-Length (see 12.7.1.6.2 (Key derivation function (KDF))) to derive PTK as defined in 12.7.1.6.5 (PTK)
* Derive PTK as defined in 12.7.1.6.5 (PTK)
* <#DHss Tag>Upon completion of PTK generation, the shared secret, DHss, shall be irretrievably deleted.
* <#DHss Tag>Include a Diffie-Hellman Parameter element as defined in IETF RFC 8110 in the second message of the FT protocol (see 13.8 (FT authentication sequence)).
* <#DHss Tag>Indicate chosen finite cyclic group in the Diffie-Hellman Parameter element of message 2, which is the same as the finite cyclic group in the Diffie-Hellman Parameter element of message 1
* <#DHss Tag>Indicate its ephemeral public key in the Diffie-Hellman Parameter element of message 2
* Calculate MIC in the FTE as follows:
	+ Use the key, the algorithm, and the MIC size as defined in 13.8.5 (FT authentication sequence: contents of fourth message)
	+ On the concatenation of the following data, in the order given here as the input:
		- FTO’s MAC address
		- FTR’s MAC address
		- RSNE sent in the Beacons transmitted by the AP with MAC address equal to A1 field of message 1
		- RSNXE sent in the Beacons transmitted by the AP with MAC address equal to A1 field of message 1
		- the body of the second message with MIC field of the FTE set to 0
* Include MIC in the FTE rather than set it to 0 as described in 13.8.3 (FT authentication sequence: contents of second message)

<#DHss Tag>Otherwise, an FTR shall not include a Diffie-Hellman Parameter element in the second message of the FT protocol.

After receiving the second message of the FT protocol with the status code set to SUCCESS, an FTO shall:

* <#DHss Tag>If the FTO includes a Diffie-Hellman Parameter element in the first message of the FT protocol, validate that there is a Diffie-Hellman Parameter element included in the second message of the FT protocol. If the validation fails, the FTO shall discard the frame and terminate further protocol processing.
* <#DHss Tag>If the FTO does not include a Diffie-Hellman Parameter element in the first message of the FT protocol, validate that there is no Diffie-Hellman Parameter element included in the second message of the FT protocol. If the validation fails, the FTO shall discard the frame and terminate further protocol processing.
* <#DHss Tag>If the FTO includes a Diffie-Hellman Parameter element in the first message of the FT protocol, validate that the finite cyclic group indicated in the Diffie-Hellman Parameter element in message 3 is the same as the finite cyclic group indicated in the Diffie-Hellman Parameter element in message 1. If the validation fails, the FTO shall discard the frame and terminate further protocol processing.
* <#DHss Tag>Verify the public key indicated in the Diffie-Hellman Parameter element in message 2 as specified in 5.6.2.3 of NIST SP 800-56A R2. If verification fails, the FTO shall discard the frame and terminate further protocol processing.
* <#DHss Tag>If the message 2 is not discarded, perform the group’s scalar-op (see 12.4.4.1 (General)) with the FTR’s ephemeral public key and its own ephemeral private key to produce an ephemeral Diffie-Hellman shared secret, DHss.
* <#DHss Tag> Append DHss at the end of context to function KDF-Hash-Length (see 12.7.1.6.2 (Key derivation function (KDF))) to derive PTK as defined in 12.7.1.6.5 (PTK)
* Derive PTK as defined in 12.7.1.6.5 (PTK)
* <#DHss Tag>Upon completion of PTK generation, the shared secret, DHss, shall be irretrievably deleted.
* Have the S1KH of the FTO verify the MIC in the FTE. If the verification fails, the FTO shall discard the frame and terminate further protocol processing.