IEEE P802.11
Wireless LANs

|  |
| --- |
| Returning Counters to AEAD Construction |
| Date: 2016-05-15 |
| Author(s): |
| Name | Affiliation | Address | Phone | Email |
| Peter Yee | NSA/IAD |  | +1 415 215 7733 | peter at akayla dawt com |
|  |  |  |  |  |

Abstract

This submission addresses CIDs 20045-20054, 20124, and 20167, which all deal with the use of AES-SIV in IEEE 802.11ai, Draft 7.0.

**Issue**: Draft P802.11ai\_D7.0 replaced the use of AES-GCM as the AEAD cipher with AES-SIV for use in protecting FILS frames. This replacement was made based on the argument that counter value reuse under the same key with AES-GCM breaks the security of that mode. However, that argument makes assumptions regarding the implementation of AES-GCM in IEEE 802.11ai rather than an actual protocol weakness. Furthermore, use of a non-FIPS (Federal Information Processing Standard) approved cryptographic mode will prevent this specification from being used by the US Government and those organizations that adhere to the FIPS as a matter of best practices. There is no good reason to unnecessarily limit the widest possible marketplace for IEEE 802.11ai-implementing products.

**Proposal**: Use AES-GCM as the AEAD cipher to protect FILS frames instead of AES-SIV. A FIPS-approved secure cipher mode will be accepted in a wider marketplace.

The text convention is *text that furthers discussion of resolution* and ***instructions to the editor***.

*In a few places in the table below, missing hyphens following AES or SHA have been added.*

***Instruct the editor to modify Table 9-132 as indicated:***

**9.4.2.25.3 AKM suites**

 **Table 9-132—AKM suite selectors**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  OUI |  Suite  type | Authentication type | Key management type | Key derivation type |
| 00-0F-AC |  14 | Key managementover FILS using SHA256 and AES-GCM-128 | FILS key managementdefined in 12.11.2.5 (Key establishment with FILS authentication) | Defined in 12.11.2.5 (Keyestablishmentwith FILS authentication)using SHA-256. |
| 00-0F-AC |  15 | Key managementover FILS usingSHA-384 and AES-GCM-256 | FILS key managementdefined in 12.11.2.5 (Key establishment with FILS authentication) | Defined in12.11.2.5 (Keyestablishmentwith FILS authentication)using SHA-384. |
| 00-0F-AC |  16 | FT authenticationover FILS with SHA-256 and AES-GCM-128 | FT authentication defined in 12.7.1.7.2 (Key derivation function (KDF)) | Defined in 12.7.1.7.2 (Keyderivation function(KDF)) using SHA-256. |
| 00-0F-AC |  17 | FT authenticationover FILS with SHA-384 and AES-GCM-256 | FT authentication defined in 12.7.1.7.2 (Key derivation function (KDF)) | Defined in 12.7.1.7.2 (Keyderivation function(KDF)) SHA-384. |

CIDs 20045 and 20124

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20045 | Use of AES-SIV (in 4 places on this page) prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Change AES-SIV back to AES-GCM (in all 4 places). | Accepted. |
| 20124 | "AES-SIV" is not used consistently in Table 9-132: the FT AKMs leave out "AES-" while the non-FT ones include it. | On page 55 line 51, replace "and SIV-256" with "and AES-SIV-256".On page 55 line 57, replace "and SIV-512" with "and AES-SIV-512". | Revised: AES- added where needed, but as AES-GCM, not AES-SIV |

***Instruct editor to modify section 12.6.1.1.6 as indicated:***

**12.6.1.1.6 PTKSA**

The PTKSA consists of the following elements:

— PTK

— Pairwise cipher suite selector

— Supplicant MAC address or STA’s MAC address

— Authenticator MAC address or BSSID

— Key ID

* If FT key hierarchy is used,

— R1KH-ID

— S1KH-ID

— PTKName

— If FILS is used,

— Non–AP STA’s AEAD counter

— AP’s AEAD counter

CIDs 20046:

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20046 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |

***Instruct editor to add FILS-specific modifications to section 12.7.2, sub (f) and modify sub (j) as indicated:***

**12.7.2 EAPOL-Key frames**

1. **EAPOL-Key IV**. This field is 16 octets, represented as an unsigned binary number. It contains the IV used with the KEK. It shall contain 0 when an IV is not required. When the AKM negotiated is not 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, it should be initialized by taking the current value of the global key counter (see 12.7.11 (RSNA Authenticator key management state machine)) and then incrementing the counter. Note that only the lower 16 octets of the counter value are used. When the AKM negotiated is 00-0F-AC:14, 00-0F-AC:15, 00-0F-AC:16, or 00-0F-AC:17, the current value of the transmitter’s AEAD counter from the PTKSA is encoded in the field.
2. If the Encrypted Key Data subfield (of the Key Information field) is 1, the entire Key Data field shall be encrypted. If the Key Data field uses the NIST AES key wrap, then the Key Data field shall be padded before encrypting if the key data length is less than 16 octets or if it is not a multiple of 8. The padding consists of appending a single octet 0xdd followed by zero or more 0x00 octets. When processing a received EAPOL-Key frame, the receiver shall ignore this trailing padding. If the Key Data field uses an AEAD cipher, then the Key Data field shall not be padded and the AAD for the encipherment operation shall be the data of the EAPOL-Key frame from the EAPOL protocol version field (inclusive) to the Key Data field (exclusive). If the AEAD cipher requires a unique counter, it shall use the EAPOL-Key IV. Key Data fields that are encrypted, but do not contain the GroupKey or SMK KDE, shall be accepted.

CID 20047

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20047 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |

*One instance of a double hyphen corrected below.*

***Instruct the editor to modify table 12-8 as indicated:***

**12.7.3 EAPOL-Key frame construction and processing**

 **Table 12-8—Integrity and Key Wrap Algorithms**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  AKM |  Integrity Algorithm | KCK bits | Size of MIC | Key-wrap algorithm |  KEK bits |
| 00-0F-AC:14 | AES-GCM-128 |  256 |  0 | AES-GCM-128 |  128 |
| 00-0F-AC:15 | AES-GCM-256 |  384 |  0 | AES-GCM-256 |  256 |
| 00-0F-AC:16 | AES-GCM-128 |  256 |  0 | AES-GCM-128 |  128 |
| 00-0F-AC:17 | AES-GCM-256 |  256 |  0 | AES-GCM-256 |  256 |

CID 20048

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20048 | Use of AES-SIV (in 4 places on this page) prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Change AES-SIV back to AES-GCM (in all 4 places). | Accepted. |

*Underscores added to TK\_bits below where missing.*

***Instruct the editor to modify section 12.11.2.5.3 as indicated:***

**12.11.2.5.3 PTKSA key derivation with FILS authentication**

For PTKSA key generation, the inputs to the KDF are the PMK of the PMKSA, a constant label, and a concatenation of the STA’s MAC address, the AP’s BSSID, the STA’s nonce, and the AP’s nonce. When the AKM negotiated is 00-0F-AC:14 or 00-0F-AC:16, the length of KEK shall be 128 bits, and the length of the KCK 256 bits. When the AKM negotiated is 00-0F-AC:15 or 00-0F-AC:17, the length of the KEK shall be 256 bits, and the length of KCK shall be 384 bits. When the AKM negotiated is 00-0F-AC:16, FILS-FT is 256 bits; when AKM negotiated if 00-0F-AC:17, FILS-FT is 384 bits; otherwise, FILS-FT is not derived. The total amount of bits extracted from the KDF shall therefore be 384+TK bits, 640+TK bits, 640+TK bits, or 1024+TK bits depending on the AKM negotiated, where TK\_bits are determined from Table 12-4 (Cipher suite key lengths):

 KCK || KEK || TK [ || FILS-FT ] = KDF-X(PMK, “FILS PTKSA Derivation”, SPA || AA || SNonce ||

 ANonce)

where:

* X is 384+TK\_bits, 640+TK\_bits, 640+TK\_bits, or 1024+TK\_bits from Table 12-4 (Cipher suite key lengths) depending on the AKM negotiated
* PMK is the PMK from the PMKSA, either created from an initial FILS connection or from a cached PMKSA, when PMKSA caching is used
* SPA is the STA’s MAC address and the AA is the AP’s BSSID
* SNonce is the STA’s nonce and ANonce is the AP’s nonce
* The brackets indicate the generation of FILS-FT when doing FT initial mobility domain association using FILS authentication; FILS-FT is not generated otherwise

FILS uses two AEAD counters, one for the local STA and one for its peer. The STA shall set both counters to zero when creating a PTKSA.

CID 20049

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20049 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |

***Instruct the editor to modify section 12.11.2.6.2 as indicated:***

**12.11.2.6.2 (Re)Association Request for FILS key confirmation**

The plaintext passed to the AEAD algorithm is the data that would follow the FILS Session element in an unencrypted frame. The unique counter required by the AEAD algorithm shall be the current value of the AEAD counter from the non-AP STA. The output of the AEAD algorithm becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Request frame. The output of the algorithm is as specified in RFC 5116. The resulting (Re)Association Request frame shall be transmitted to the AP.

The AP decrypts and verifies the received (Re)Association Request frame with the AEAD algorithm as defined in 12.11.2.7 (AEAD cipher mode for FILS) with the KEK as the key. The AAD is reconstructed as defined above and is passed, along with the ciphertext of the received frame to the AEAD decryption operation. The AP uses the current value of the AEAD counter for the non-AP STA to decrypt and verify the received frame.

CIDs 20050, 20051, and 20167

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20050 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |
| 20051 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |
| 20167 | The AAD used with the AEAD algorithm is defined as a concatenation of number of data items. The only AEAD algorithm that is used in the P802.11ai is AES-SIV which supports multiple variable-length octet strings ("a vector of strings") as the authenticated data. As such, the concatenation to construct a single AAD seems unnecessary for SIV. Is this in support of some other AEAD algorithms or should these be define as a vector of strings instead? | Clarify how AES-SIV vector of strings is used (a single AAD vs. separate string for each of the data items). Same for (Re)Association Response frame in 12.11.2.6.3. | Rejected: switching to AES-GCM removes the need to clarify the AES-SIV vector of strings. |

***Instruct the editor to modify section 11.11.2.6.3 as indicated:***

**12.11.2.6.3 (Re)Association response for FILS key confirmation**

The plaintext passed to the AEAD algorithm is the data that would follow the FILS Session element in an

unencrypted frame. The unique counter required by the AEAD algorithm shall be the current value of the AEAD counter for the AP. The output of the AEAD algorithm becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Request frame. The output of the algorithm is as

specified in IETF RFC 5116. The resulting (Re)Association Response frame shall be transmitted to the STA.

The STA decrypts and verifies the received (Re)Association Response frame with the AEAD algorithm as

defined in 12.11.2.5 (Key establishment with FILS authentication) with the KEK as the key. The AAD is

reconstructed as defined in this subclause above and is passed with the ciphertext of the received frame to

the AEAD decryption operation. The STA uses the current value of the AEAD counter for the AP to decrypt and verify the received frame.

CIDs 20052 and 20053

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20052 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |
| 20053 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Put back the counters that are needed to support AES-GCM. | Accepted. |

***Instruct the editor to modify section 12.11.2.7 as indicated:***

**12.11.2.7 AEAD cipher mode for FILS**

AES-GCM-128 is used when the AKM negotiated is 00-0F-AC:14 or 00-0F-AC:16 and AES-GCM-256 is used when the AKM negotiated is 00-0F-AC:15 or 00-0F-AC:17. AES-GCM-X (in Table 9-132) is GCM with an X-bit AES key.

When the AEAD cipher mode used is GCM, the nonce, N, shall be 12 octets in length and shall be constructed as a concatenation of a one octet sender indication (0x00 = non-AP STA, 0x01 = AP) and the 11 least significant octets of the AEAD counter for the local STA (from the PTKSA) in big endian encoding. The AEAD counter is implicit in the (Re)Association Request and (Re)Association Response frames (0) and explicitly identified in the EAPOL-Key frames. Each successive invocation of the encryption operation of GCM shall increment the AEAD counter by 1. To guarantee uniqueness of GCM nonce values, the STA shall either deauthenticate or reassociate to derive a new PTKSA before the AEAD counter is incremented to 288.

When processing a received EAPOL-Key frame, the STA shall verify that the received frame contains an AEAD counter that is strictly greater than the AEAD counter for the peer in the PTKSA. If the counter is not greater, the STA shall discard the received EAPOL-Key frame. Otherwise, the STA shall update the AEAD counter for the peer in the PTKSA to the value received in the EAPOL-Key frame.

CID 20054

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 20054 | Use of AES-SIV prevents this protocol from being used by some important groups. These groups must use FIPS-approved algorithms. | Change AES-SIV back to AES-GCM, and restore the text about the AES-GCM counters. | Accepted. |