IEEE P802.11
Wireless LANs

|  |
| --- |
| Removing Counters from AEAD Construction |
| Date: 2015-10-12 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Dan Harkins | HP | 1322 Crossman avenue, Sunnyvale, California, United States of America | +1 408 227 4500 | dharkins at aruba networks dot com |
|  |  |  |  |  |

Abstract

This submission addresses CIDs 10045, 10046, 10047, 10050, 10051, 10052, 10705, 10706, 10727, 10759, and 10760 which all concern the fragility of the current AEAD mode and its dependence on unique counters.

**Issue**: Draft P802.11ai\_D6.0 specifies the use of AES-GCM as the AEAD cipher to protect FILS frames. This cipher mode is provably secure but is better suited for bulk data encryption instead of protocol handshaking as it is being used here. One issue of its unsuitability for protocol handshaking is its dependence on unique counters per invocation of AES-GCM. The entire security of the cipher mode breaks down if a counter is reused. As a result there is text in various places throughout the draft dealing with the creation, construction, maintenance, and processing of counters in an effort to ensure security does not completely break down.

**Proposal**: Use AES-SIV as the AEAD cipher to protect FILS frames. This provably secure cipher mode is better suited for protocol handshaking since it was designed as an efficient deterministic authenticated encryption scheme that does not require the addition of nonces or counters. This will greatly simplify the specification, and also simplify implementations.

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

*General change necessary for addressing CIDs but text not subject to a CID. Note that the Key derivation type for Suite type 15 was wrong and that fact is being noted here—it should be SHA-384 not SHA-256:*

***Instruct the editor to modify Table 8-130 as indicated:***

**8.4.2.24.3 AKM suites**

 **Table 8-130—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 SIV-256 | FILS key managementdefined in 11.11.2.5 (Keyestablishment with FILSauthentication) | Defined in11.11.2.5 (Keyestablishmentwith FILS authentication)usingSHA-256. |
| 00-0F-AC |  15 | Key managementover FILS usingSHA-384 and AESSIV-512 | FILS key managementdefined in 11.11.2.5 (Keyestablishment with FILSauthentication) | Defined in11.11.2.5 (Keyestablishmentwith FILS authentication)usingSHA-384. |
| 00-0F-AC |  16 | FT authenticationover FILS with SHA-256 and SIV-256 | FT authentication definedin 11.6.1.7.2 (Key derivationfunction (KDF)) | Definedin 11.6.1.7.2 (Keyderivation function(KDF)) usingSHA-256. |
| 00-0F-AC |  17 | FT authenticationover FILS with SHA-384 and SIV-512 | FT authentication definedin 11.6.1.7.2 (Key derivationfunction (KDF)) | Definedin 11.6.1.7.2 (Keyderivation function(KDF)) usingSHA 354. |

CID 10045

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10045 | Maintenance of counters is not necessary | Use an AEAD mode that doesn’t require counters | Revised: AES-SIV mode replaces AES-GCM, it doesn’t require counters. |

***Instruct editor to modify section 11.5.1.1.6 as indicated:***

**11.5.1.1.6 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

CIDs 10046 and 10727:

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10046 | it will be less complicated to not have ot maintain additional counters. | use an AEAD mode that doesn't require counters. | Revised: AEAD mode no longer requires counters. Sentence deleted. Section has been reverted back. |
| 10727 | "transmitter's AEAD counter" -- this is ambiguous, because a STA has two AEAD counters (see 151.61) | Change to "AEAD counter for the local STA" | Revised: AEAD mode no longer requires counters. Sentence deleted. |

***Instruct editor to remove FILS-specific modifications to section 11.6.2, sub (f) and modify sub (j) as indicated:***

**11.6.2 EAPOL-Key frames**

1. **EAPOL-Key IV**. This field is 16 octets. It contains the IV used with the KEK. It shall contain 0 when an IV is not required. It should be initialized by taking the current value of the global key counter (see 11.6.11 (RSNA Authenticator key management state machine)) and then incrementing the counter. Note that only the lower 16 octets of the counter value are used.
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). Key Data fields that are encrypted, but do not contain the GroupKey or SMK KDE, shall be accepted.

CID 10047

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10047 | don't use a fragile AEAD mode like GCM. Use a robust and misuse resistant AEAD mode instead. | change all of the integrity algorithm and key-wrap algorithms to be a robust and misuse resistant deterministic key wrapping and AEAD mode. | Revised: AES-SIV mode is robust and misuse resistant, and it is a genuine key wrapping algorithm as opposed to AES-GCM; it replaced AES-GCM. |

***Instruct the editor to modify table 11-8 as indicated:***

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

 **Table 11-8—Integrity and Key Wrap Algorithms**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  AKM |  Integrity Algorithm | KCK bits | Size of MIC | Key-wrap algorithm |  KEK bits |
| 00-0F-AC:14 | AES-SIV-256 |  256 |  0 | AES-SIV-256 |  256 |
| 00-0F-AC:15 | AES-SIV-512 |  384 |  0 | AES-SIV-512 |  512 |
| 00-0F-AC:16 | AES-SIV-256 |  256 |  0 | AES-SIV-256 |  256 |
| 00-0F-AC:17 | AES-SIV-512 |  256 |  0 | AES-SIV-512 |  512 |

CID 10050

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10050 | It will be less complicated to not have to maintain additional counters. | Use an AEAD mode that doesn’t require counters. | Revised: AES-SIV, which does not require counters, has replaced AES-GCM. |

*In addition there are general changes related to key lengths that are necessary for addressing all CIDs but are not related to any particular CID.*

***Instruct the editor to modify section 11.11.2.5.3 as indicated:***

**11.11.2.5.3 PTK 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 256 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 512 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 512+TK bits, 768+TK bits, 896+TK bits, or 1280+TK bits depending on the AKM negotiated, where TK\_bits are determined from Table 11-4 (Cipher suite key lengths):

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

 ANonce)

where:

* X is 512+TK\_bits, 768+TK bits, 896+TK bits, or 1280+TK bits from Table 11-4 (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

CIDs 10052 and 10705

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10052 | Don’t use a fragile AEAD mode like GCM. Use a robuse and misuse resistant AEAD mode instead. | Use an AEAD mode that doesn’t require counters. | Revised: AES-SIV, which does not require counters, has replaced AES-GCM. The problematic sentence has been removed. |
| 10705 | "The AP uses the current value of the AEAD counter for the non-AP STA to decrypt and verify the received frame." is either wrong or confusing, because 11.11.2.7 says "The AEAD counter is implicit in the (Re)Association Request and (Re)Association Response frames (0)" | Change to say it uses the value 0 | Revised: the AEAD mode has been replaced and no longer uses counters. The problematic sentence has been removed. |

***Instruct the editor to modify section 11.11.2.6.2 as indicated:***

**11.11.2.6.2 (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 ciphertext output by the AEAD algorithm concatenated with the Authentication Tag becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Request frame. 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 11.11.2.5 with the KEK as the key. The AAD is reconstructed as defined in this subclause above and is passed, along with the ciphertext of the received frame to the AEAD decryption operation.

CID 10706

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10706 | "The STA uses the current value of the AEAD counter for the AP to decrypt and verify the received frame." is either wrong or confusing, because 11.11.2.7 says "The AEAD counter is implicit in the (Re)Association Request and (Re)Association Response frames (0)" | Change to say it uses the value 0 | Revised: the AEAD mode has been replaced and no longer uses counters. The problematic sentence has been removed. |

*General change to remove a sentence that is no longer relevant (and is consistent with the resolution of 10052 above).*

***Instruct the editor to modify section 11.11.2..6.3 as indicated:***

**11.11.2.6.3 (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 ciphertext output by the AEAD algorithm concatenated with the Authentication Tag becomes the data that follows the FILS Session element in the encrypted and authenticated (Re)Association Response frame. 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 11.11.2.5 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.

CIDs 10052, 10759, and 10760

|  |  |  |  |
| --- | --- | --- | --- |
| CID | Comment | Proposed Change | Proposed Resolution |
| 10052 | Don’t use a fragile AEAD mode like GCM. Use a robuse and misuse resistance AEAD mode instead. | Use an AEAD mode that doesn’t require counters. | Revised: AES-SIV, a robust and misuse resistant AEAD mode which does not require counters, has replaced AES-GCM. |
| 10759 | AES-GCM-X | No definition of reference of algorithm AES-GCM-X | Revised: AES-GCM has been replaced and the problematic sentence has been deleted. |
| 10760 | Counter required for security. This is a bad security choice and it would be better security and a simpler design if the AEAD used was “nonce misue resistant” | Replace AES-GCM with an AEAD algorithm that is nonce misuse-resistant. | Revised: AES-SIV, a nonce misuse-resistant AEAD mode, has replaced AES-GCM. |

***Instruct the editor to modify section 11.11.2.7 as indicated:***

**11.11.2.7 AEAD cipher mode for FILS**

FILS authentication uses an AEAD cipher mode to protect (Re)Association Request/Response and EAPOLKeyframes. The AEAD cipher mode is determined by the specific FILS AKM negotiated.

AES-SIV-256 is used when the AKM negotiated is 00-0F-AC:14 or 00-0F-AC:16 and AES-SIV-512 is used when the AKM negotiated is 00-0F-AC:15 or 00-0F-AC:17.

**References:**