IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AES-CCM Mode to be Used with FILS Association Frames | | | | |
| Date: 2012-11-14 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| René Struik | Struik Security Consultancy | Toronto ON | +1 415 690-7363  +1 647 867-5658  Skype: rstruik | rstruik.ext@gmail.com |

Abstract

This submission proposes the AES-CCM mode of operation as the authenticated encryption and authentication mode to be used for the FILS AEAD mode for Association frames as defined in 12/1045r6. This well-understood mode of operation is standardized in NIST SP 800-38C and has been used with IEEE 802 standards, including 802.11, since 2003 and is widely used elsewhere as well.

Change history:

R1 – uses “throw-away” key KEK2 to secure piggy-backed info in FILS Association frames, thus logically distinguishing this from the key KEK that may be used after the completion of the authenticated key agreement scheme. Introduced the temporary key KCK2 as well, which is only used during FILS key confirmation and destroyed prior to completion of the FILS protocol (for the same reason as with the KEK2 key). (Introduction of temporary keys KEK2 and KCK2 continues the sound cryptographic practice that the keying material used during the key confirmation stage of an authenticated key agreement scheme shall not be used beyond that protocol.) Added language that the shared key is destroyed upon computing the derived keying material via the used KDF. The keys TK, KEK, and KCK have the same roles as the keys with the same names in 802.11-2012.

R0 – original proposal

***Instruct the editor to modify the specification text in 12/1045r6 as follows:***

***Modify section 11.9a.2.3 as follows:***

**11.9a.2.3 Key Derivation with FILS Authentication**

Key derivation with FILS Authentication uses the KDF from section 11.6.1.7.2 to produce five keys, two key encryption keys (KEK and KEK2), two confirmation keys (KCK and KCK2), and a traffic key (TK). The inputs to the KDF are the two 16 octet nonces produced by the STA and AP, a constant label, the ERP secret result if a TTP is being used, and, the Diffie-Hellman shared secret, *ss*, if PFS is being used. The length of the KEK and KEK2 shall be 128 bits, and the KCK and KCK2 shall be 256 bits, and therefore the output from the KDF shall be 768+TK\_bits, where TK\_bits is determined from table 11-4.

KCK2 | KEK2 | KCK | KEK | TK = KDF-X(NSTA | NAP, “FILS KECK PTK Derivation”, [rMSK][ | *ss*])

Where X is 768+TK\_bits from table 11-4, rMSK is the output of the ERP exchange if a trusted third party was used, and *ss* is the shared secret resulting from the Diffie-Hellman exchange if PFS was used.

Upon completion of the key derivation computation, the shared secret shall be irretrievably destroyed.

The KCK2 shall only be used with key confirmation (see 11.9a.2.4). The KEK2 shall only be used with the encrypt-and-authenticate (see 11.9a.2.5) and decrypt-and-verify (see 11.9a.2.6) functions. Both keys KCK2 and KEK2 are temporary keys shall only be used during the FILS authentication protocol.

The keys KCK, KEK, and TK may be used after successful completion of the FILS authentication protocol and shall be used in exactly the same way as same-named keys of IEEE 802.11-2012 (but now derived as specified above).

***Replace sections 11.9a.2.5 and 11.9a.2.6 by sections 11.9a.2.5, 11.9a.2.6 and add a section 11.9a.2.7, as follows:***

**11.9a.2.5 AEAD scheme**

The authenticated encryption with associated data scheme to be used shall be the negotiated cipher indicated by the cipher suite in the Association Request and Response frames. Currently, the only such scheme specified is the AES-CCM mode of operation, which is the CCM scheme specified in NIST SP 800-38C, Appendix A, with the following instantiation:

* + The block cipher shall be AES-128 (see FIPS Pub 197);
  + The parameter *t*, *q*, *n* and shall be set to *t*=16, *q*=2, and *n*=13.

**11.9a.2.6 Encrypt and Authenticate operation for FILS Association frames**

The AEAD scheme of 11.9a.2.5 shall be used with the 802.11 Associate Request frame (for enciphering by STA) or with the 802.11 Associate Response frame (for enciphering by AP), with the following instantiation:

* The key *K* shall be set to KEK2 (see 11.9a.2.3);
* The associated data string *A* shall be set to the string AAD;
* The string *P* shall be set to the plaintext;
* The nonce *N* shall be set to

a) For processing by STA: use the 13-octet all-zero string;

b) For processing by AP: use the 13-octet all-one string.

The function shall output the string *C* as ciphertext.

**11.9a.2.7 Decrypt and Verify operation for FILS Association frames**

The AEAD scheme of 11.9a.2.5 shall be used with the 802.11 Associate Request frame (for deciphering by STA) or with the 802.11 Associate Response frame (for deciphering by AP), with the following instantiation:

* The key *K* shall be set to KEK2 (see 11.9a.2.3);
* The associated data string *A* shall be set to the AAD;
* The string C shall be set to the ciphertext;
* The nonce *N* shall be set to

a) For processing by AP: use the 13-octet all-zero string.

b) For processing by STA: use the 13-octet all-one string;

The function shall output the payload string *P* as the plaintext if the decryption –verification process is successful and shall output a failure otherwise.

NOTE – Detailed textual changes to remainder of 12/1045r6, so as to help the Editor, provided in Appendix A.

**Motion-1:** To authorize the Editor to incorporate the text changes proposed in contribution 12/1385r1 (*11-12-1385-01-00ai-fils-AEAD-mode-of-operation*) to the relevant portions of 12/1045r6 incorporated with the draft TGai Draft Specification Document.

Yes: \_\_\_\_\_\_\_\_\_\_\_\_; No: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; Abstain: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Appendix A – detailed changes to 12/1045r6

***Editorial note:***

***Removing the occurrence of the authentication tag interface in the defined interface in 1045r6: interface of AEAD mode of operation (at “right” abstraction level) is as follows:***

1. ***AEAD mode takes as input associated data A and plaintext P and produces ciphertext C (where the exact location of the authentication tag is hidden, since internal to the particular AEAD mechanics)***
2. ***AEAD decryption and verification mode takes as input ciphertext C (with usually hidden within its bowels an authentication tag). The output of this operation is the retrieved plaintext P or (if the procedure yields a cryptographic failure), an error message.***

***Note that this abstraction level of interface allows easy incorporation of other AEAD modes in the future (including those that format redundancy in the ciphertext C differently than “at the end of the ciphertext”).***

***Note: the changes to effectuate the above conceptually “right” abstraction level of the interface have the same effect as simply setting the TAG field to the empty string (of bit-length zero) in 12/1045r6. Ultimately, all occurrences of TAG field there have to be rooted out from that document (seemingly making this an unwieldy exercise – done nevertheless, indicated below).***

***Replacing KCK key to be used with key confirmation by temporary key KCK2 (so as to avoid overloading of terms). Removal of 2nd key that was assumed to be used with AEAD mode in 12/1045r6 right now (which is not needed with a single-key AEAD mode of operation).***

***Removing need for required changes to 802.11-2012 outside FILS protocol scope: everything can remain as is provided on does logical separation of keys used during FILS protocol (KCK2, KEK2) and keys used subsequent to successful completion hereof (KCK, KEK).***

***Modify section 6.3.5.5.2 as indicated:***

*Remove the following row in Table 8-22:*

|  |  |  |
| --- | --- | --- |
| 9 | FILS TAG | A field that contains an authentication tag used to secure FILS frames. |

*Remove the following row in Table 8-23:*

|  |  |  |
| --- | --- | --- |
| 7 | FILS TAG | A field that contains an authentication tag used to secure FILS frames. |

***Remove section 8.4.2.121d (FILS tag element) entirely***

***Modify section 11.9a.2.4 as indicated:***

Key confirmation for FILS Authentication is an Associate Request followed by an Associate Response. The Association Request and Association Response shall be protected using the KEK2 ~~and KMK~~ according to section 11.9a.2.5 and 11.9a.2.6.

Upon the completion of key establishment (11.9a.2.2) and key derivation (11.9a.2.3) the STA shall construct a nascent 802.11 associate request frame indicating its selected ciphersuite and the FILS AKM, and the FILS Key Confirmation element. ~~The FILS TAG field shall be set to zero~~. The content of the Key Auth field of the Key Confirmation element depends on the type of FILS authentication.

The AP transfers any necessary KDEs to the STA in the Association Response frame. The AP may include one or more KDEs using the FILS KDE container. The format and the rules for transferring the KDE shall follow section 11.6.2 (EAPOL Key Frames).

For FILS Authentication using a trusted third party, the Key Auth field of the Key Confirmation element of the Association Request shall be:

Key-Auth = HMAC-SHA256(KCK2, NSTA | NAP | STA-MAC | AP-BSSID).

For FILS Authentication without a trusted third party, the Key Auth field of the Key Confirmation element in the Association Request shall contain a digital signature using the STA’s private key, the specific construction of the digital signature depends on the crypto-system of the public/private key pair:

Key-Auth = Sig-STA(gSTA | gAP | NSTA | NAP | STA-MAC | AP-BSSID).

Where Sig-STA indicates a digital signature using the STA’s private key, gSTA is the octet-string representation of the STA’s public Diffie-Hellman value, gAP is the octet-string representation of the AP’s public Diffie-Hellman value, NSTA is the nonce selected by the STA, and NAP is the nonce selected by the AP.

The 802.11 Association Request frame shall be secured as follows:

* The input key~~s~~ shall be the KEK2 ~~and KMK~~
* The input plaintext shall be the contents of the Association Request frame that follow the FILS ~~TAG~~ Session element
* The input AAD shall be:
  + 1. The STA MAC
    2. The AP BSSID
    3. The STA’s nonce
    4. The AP’s nonce
    5. The contents of the Association Request frame from the capability (inclusive) to the FILS ~~TAG~~ Session element (~~ex~~inclusive)
* The input key~~s~~, the plaintext, and the AAD shall be passed to the encrypt-and-authenticate operation specified in 11.9a.2.5.
* ~~The output authenticating tag from 11.9a.2.5 shall be copied into the TAG field of the FILS TAG element~~
* The output ciphertext from 11.9a.2.5 shall become the remainder of the Association Request frame that follows the FILS ~~TAG~~ Session element.

The resulting 802.11 Association Request frame shall be transmitted to the AP.

The received 802.11 Association Request frame shall be processed as follows:

* The input key~~s~~ shall be the KEK2 ~~and KMK~~
* ~~The input tag shall be taken from the TAG field of the FILS TAG element~~
* The input ciphertext shall be the contents of the Association Request frame that follow the FILS ~~TAG~~ Session element
* The input AAD shall be:
  + 1. The STA MAC
    2. The AP BSSID
    3. The STA’s nonce
    4. The AP’s nonce
    5. The contents of the Association Request frame from the capability (inclusive) to the FILS ~~SIV~~ Session element (~~ex~~inclusive)
* The input keys, ~~the TAG,~~ the ciphertext, and the AAD shall be passed to the decrypt-and-verify operation specified in 11.9a.2.6.

If the output from 11.9a.2.6 returns a failure, authentication shall be deemed a failure. If the output returns plaintext, the Key-Auth from the decrypted Association frame shall be checked. If it is incorrect, authentication shall be deemed a failure. If authentication is deemed a failure, the KCK2, KEK2, KCK, KEK, ~~KMK~~, and TK, ~~PMK~~, ~~and all shared secrets~~ shall be irretrievably destroyed. If authentication is not deemed a failure, the AP shall check the Key-Auth field in the Key Confirmation element.

For FILS Authentication using a trusted third party, the AP shall construct a verifier as follows:

Key-Auth’ = HMAC-SHA256(KCK, NSTA | NAP | STA-MAC | AP-BSSID).

If Key-Auth’ differs from the Key-Auth field in the Key Confirmation element, authentication shall be deemed a failure.

For FILS Authentication without a trusted third party, the AP shall use the STA’s (certified) public key from the Public Key IE in the Association frame to verify the contents of the Key-Auth field of the Key Confirmation element. The specific technique for verification depends on the crypto-system used by the public key. If verification fails, authentication shall be deemed a failure.

If authentication is a failure, the KCK2, KEK2, KCK, KEK, TK, ~~PMK~~, ~~and all shared secrets~~ shall be irretrievably destroyed. Otherwise, the AP shall then construct a nascent 802.11 associate response frame confirming the selected ciphersuite and the FILS AKM, and containing the FILS KDE Container, and its own Key-Auth. ~~The FILS TAG element shall be set to zero.~~

For FILS authentication using a trusted third party, the Key Auth field of the Key Confirmation element in the Association Response shall be:

Key-Auth = HMAC-SHA256(KCK2, NAP | NSTA | AP-BSSID | STA-MAC).

For FILS Authentication without a trusted third party, the Key Auth field of the Key Confirmation element in the Association Response shall contain a digital signature using the AP’s private key, the specific construction of the digital signature depends on the crypto-system of the public/private keypair:

Key-Auth = Sig-AP(gAP | gSTA | NAP | NSTA | AP-BSSID | STA-MAC ).

Where Sig-AP indicates a digital signature using the AP’s private key, and where gSTA, gAP, NSTA, and NAP are the same as in the construction of the Association Request.

The 802.11 Association Response frame shall be protected as follows:

* The input key~~s~~ shall be the KEK2 ~~and KMK~~
* The input plaintext shall be the contents of the Association Request frame that follow the FILS ~~TAG~~ Session element
* The input AAD shall be:
  + 1. The AP BSSID
    2. The STA MAC
    3. The AP’s nonce
    4. The STA’s nonce
    5. The contents of the Association Response frame from the capability (inclusive) to the FILS ~~TAG~~ Session element (~~ex~~inclusive)
* The input key~~s~~, the plaintext, and the AAD shall be passed to the encrypt-and-authentication operation specified in 11.9a.2.5.
* ~~The output TAG shall be copied into the TAG field of the FILS TAG element~~
* The output ciphertext shall become the remainder of the Association Response frame that follows the FILS ~~TAG~~ Session element.

The resulting 802.11 Association Response frame shall be transmitted to the STA.

The STA shall protect the received 802.11 Association Response frame as follows:

* The input key~~s~~ shall be the KEK2 ~~and KMK~~
* ~~The tag shall be taken from the TAG field of the FILS TAG element~~
* The input ciphertext shall be the contents of the Association Response frame that follow the FILS ~~TAG~~ Session element
* The input AAD shall be:
  + 1. The AP BSSID
    2. The STA MAC
    3. The AP’s nonce
    4. The STA’s nonce
    5. The contents of the Association Response frame from the capability (inclusive) to the FILS ~~TAG~~ Session element (~~ex~~inclusive)
* The input key~~s~~, ~~the tag,~~ the ciphertext, and the AAD shall be passed to the decrypt-and-verify operation specified in 11.9a.2.6.

If the output from 11.9a.2.6 returns failure, authentication shall be deemed a failure. If the output returns plaintext, the Key-Auth from the decrypted Authentication frame shall be checked. If it is incorrect, authentication shall be deemed a failure. If authentication is deemed a failure, the KCK2, KEK2, KCK, KEK, ~~KMK~~, TK, ~~PMK~~, ~~and all shared secrets~~ shall be irretrievably destroyed. If authentication is not deemed a failure, the AP shall check the Key-Auth field in the Key Confirmation element.

For FILS Authentication using a trusted third party, the STA shall construct a verifier as follows:

Key-Auth’ = HMAC-SHA256(KCK2, NAP | NSTA | AP-BSSID | STA-MAC).

If Key-Auth’ differs from the Key-Auth field in the Key Confirmation element, authentication shall be deemed a failure.

For FILS Authentication without a trusted third party, the STA shall use the AP’s (certified) public key from the Public Key IE in the Association frame to verify the contents of the Key-Auth field of the Key Confirmation element. The specific technique for verification depends on the crypto-system used by the public key. If verification fails, authentication shall be deemed a failure.

If authentication is a failure, the KCK2, KEK2, KCK, KEK, ~~KMK~~,TK, ~~PMK~~, ~~and all shared secrets~~ shall be irretrievably destroyed. Otherwise, authentication succeeds. In that case, STA and AP shall irretrievably destroy the temporary keys KCK2 and KEK2 and both shall use the TK, KCK, and KEK generated in 11.9a.2.3 with the cipher indicated by the ciphersuite in the Association Request and Association Response.

***Modify section 11.6.3 as indicated:***

Remove the “TBD” line item in Table 11-9:

|  |  |  |  |
| --- | --- | --- | --- |
| AKM | Integrity Algorithm | Size of MIC | Key Wrap Algorithm |
| 00-0F-AC:6 | AES-128-CMAC | 16 | AES Key Wrap |
| ~~<ANA-12>~~ | ~~TBD~~ | ~~TBD~~ | ~~TBD~~ |

***Editorial note:*** *In other words, keep Table 11-9 of section 11.6.3 of 802.11-2012 as is.*