IEEE P802.11
Wireless LANs

|  |
| --- |
| 802.11PMKSA caching and MAC randomization  |
| Date: 2019-04-01 |
| Author(s): |
| Name | Company | Address | Phone | email |
| Thomas Derham | Broadcom |  |  | thomas.derham@broadcom.com |
| Nehru Bhandaru | Broadcom |  |  |  |
| Insun Song | Broadcom |  |  |  |

**Abstract**

This document provides a resolution for comment CID 2689. The document is based on REVmd D2.1.

R0: Initial draft

R1: Addressed review comments from preso at adhoc and subsequent offline comments. In the clarifications regarding PMKSA caching with FT Initial MD Assoc, changes include formalized definition of MPMK (already defined for FT-SAE) as the master PMK from which PMK-R0 is derived, which can be cached in PMKSA bound to AKM. Various clarifications and additions to fully define this behavior for all Auth/AKM cases.

R2 – Rebase on D2.2 (which contains some related approved changes), fix inconsistent text regarding inclusion of PMKID in M1 in non-PMKSA caching cases, fix inconsistent text in the case of using cached PMKSA from SAE with no match on AP side

R3 – Use new RSN Extension element for cap indication (based on 19/114r5)

R4 – Minor fixes from preso feedback

R5 – Removed changes related to FT (moved to separate doc)

**Discussion**

The recent 11aq amendment defines MAC address randomization behavior for a non-AP STA, whereby the STA might use a different randomized MAC address when sending probe requests and each time it associates with a network (ESS), in order to overcome privacy concerns resulting from persistent passive tracking of a STA each time it returns to a given location/network based on its MAC address identifier (which is sent in-the-clear in Address fields). The MAC address stays constant while the STA remains associated with the ESS (e.g. while roaming between APs in the ESS).

Given the increasing adoption of computationally-expensive authentication methods such as SAE, it is expected that PMKSA caching for long periods of time (hours, days or more) will become more prevalent.

Currently, 11aq text states that, if the STA uses PMKSA caching, it must return its MAC address to the MAC address that it used when establishing the PMKSA with the peer. In the general case, this is a reasonable rule because legacy peer (AP/Authenticator) implementations might be caching PMKs in a way that is indexed/linked to the STA’s/Supplicant’s MAC address.

However, it is suggested that such restriction is not required from a technical point of view – the peer (AP/Authenticator) can lookup a PMK based only on the PMKID identifier; the STA’s/Supplicant’s MAC address is not a constituent of the PMKSA definition. The fact that the PMK and/or PMKID might have been derived using the MAC address of the STA at the time does not preclude that cached PMK being used again when the STA’s MAC address has changed.

Therefore, this contribution proposes text for such capability, so that MAC randomization can be used in combination with PMKSA caching. Note that this does not impact the secrecy of the (cached) PMK, which remains a pairwise secret between the STA/Supplicant and the AP/Authenticator. This capability improves client privacy because the STA MAC address, which is sent in-the-clear in almost every frame sent to/from the STA, can be randomized each time the STA connects to the network.

It is noted that the PMKID itself, which is also sent in-the-clear but only in the initial association signaling, is static for the lifetime of the PMK and so could also be used as an identifier for tracking if that signaling is observed. Possible solutions for masking of such identifiers in (pre)association signaling can be handled in separate contributions.

The document also proposes new text and clarifications regarding PMKSA caching with FT Initial Mobility Domain Association (and some fixes in related areas). PMKSA caching with FT Initial Mobility Domain Association is ostensibly supported by the standard, e.g. references to PMKSA Caching in some FT AKM definitions, and this was also confirmed by REVmd motion related to CID 2211, however it is insufficiently specified in several aspects. The Master PMK (MPMK), currently defined for SAE as the “master” PMK from which the FT key hierarchy is defined, is formally defined as the basis for PMKSA caching with AKMs that use FT key management.

*

9.4.2.242 RSN Extension element (RSNXE)

***Instruct the editor to modify table as follows (Note to editor: this table is introduced by 114r5):***

|  |
| --- |
| Table 9-yyy Extended RSN Capabilities field  |
| Bit | Information | Notes |
| 0-3 | Field Length  | The length of the Extended RSN Capabilities field, in octets, minus 1, i.e., *n*-1. |
| 4 | Protected TWT Operations Support | The STA sets the Protected TWT Operations Support field to 1 when dot11ProtectedTWTOperationsImplemented is true, and sets it to 0 otherwise. See 10.48.1 (TWT overview). |
| 5 | PMKSA Caching with MAC Randomization | The STA sets the PMKSA Caching with MAC Randomization field to 1 when dot11PMKSACachingMACRandomizationActivated is true and sets it to 0 otherwise. |
| 6–(8×*n*-1) | Reserved |  |

|  |
| --- |
|  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* Requirements for support of MAC privacy enhancements(11aq)

***Instruct the editor to modify as follows:***

The non-AP STA connecting to an infrastructure BSS shall retain a single MAC address for the duration of its connection across an ESS. A PMKSA created as part of an RSNA will contain the MAC address of the AP used to create the PMKSA. Implementations of APs where dot11PMKSACachingMACRandomizationActivated is false or undefined might also bind a cached PMKSA to the non-AP STA’s MAC address. Therefore, a non-AP STA that supports PMKSA caching shall, if necessary, change its MAC address back to that value when attempting a subsequent association to the ESS using PMKSA caching, unless dot11PMKSACachingMACRandomizationActivated is true and the AP indicates support for PMKSA Caching with MAC Randomization in its RSN Extension element, in which case the non-AP STA may use a different MAC address.

* PMKSA

*Instruct the editor to modify as follows:*

When the PMKSA is the result of a successful IEEE 802.1X authentication, it is derived from the EAP authentication and authorization parameters provided by the AS. When the PMKSA is the result of a successful SAE authentication, it is generated as a result of the successful completion of the SAE exchange.

(M84)The PMKSA is created by the Supplicant’s SME when the EAP authentication or FILS authentication(11ai) completes successfully or the PSK is configured. The PMKSA is created by the Authenticator’s SME when the PMK is created from the keying information transferred from the AS in an(11ai) IEEE 802.1X authentication exchange, when the FILS authentication completes successfully(11ai), when the SAE exchange successfully completes, or when the PSK is configured. ~~When the negotiated AKM uses PMKID derivation with KCK as a parameter as defined in 12.7.1.3 (Pairwise key hierarchy), t~~The PMKID derived ~~from the KCK~~ during the initial 4-way handshake or FILS authentication exchange is not changed during the lifetime of this PMKSA.(M84)

NOTE – The PMKID does not change during the lifetime of the PMKSA even if the negotiated AKM uses PMKID derivation with one or more parameters that change during that lifetime (e.g. KCK, SPA). The keys derived when creating a PTKSA from a PMKSA use the current parameters at the time the PTKSA is created.

A PMKSA association is bidirectional. In other words, both parties use the information in the security association for both sending and receiving. The PMKSA is used to create the PTKSA. PMKSAs have a certain lifetime. The PMKSA consists of the following:

* PMKID, as defined in 12.7.1.3 (Pairwise key hierarchy). The PMKID identifies the security association.
* Authenticator’s MAC address; or, if the PMKSA was established by SAE authentication in an infrastructure BSS, the AP’s MAC address; or, if the PMKSA was established by SAE authentication in a PBSS or IBSS, the peer’s MAC address. For multi-band RSNA, the MAC address is associated with the operating band in use when the PMKSA is established.
* PMK
* Lifetime, as defined in 12.7.1.3 (Pairwise key hierarchy).
* AKMP.
* All authorization parameters specified by the AS or local configuration. This might include parameters such as the STA’s authorized SSID.
* Cache Identifier, if advertised by the AP in FILS Indication element(11ai).
* Cached PMKSAs and RSNA key management

***Instruct the editor to add the following paragraph to the end of this section:***

If an AP with dot11PMKSACachingMACRandomizationActivated true has cached a PMKSA, it shall assert possession of the PMKSA on reception of a (Re)Association Request frame or FILS Authentication frame from a non-AP STA indicating the matching PMKID and AKM as described above, irrespective of the MAC address used by that non-AP STA.

NOTE – An AP with dot11PMKSACachingMACRandomizationActivated false or undefined might not assert possession of a cached PMKSA if the non-AP STA that indicates the PMKID is using a different MAC address compared to when the PMKSA was established, or the AP might never cache PMKSAs at all.

***Instruct the editor to modify C.3 as follows:***

**C.3 MIB detail**

Dot11StationConfigEntry::=

 SEQUENCE {

......

dot11PMKSACachingMACRandomizationActivated TruthValue }

dot11PMKSACachingMACRandomizationActivated OBJECT-TYPE

 SYNTAX TruthValue

 MAX-ACCESS read-write

 STATUS current

 DESCRIPTION

"This is a control variable.

It is written by the MAC or an external management entity.

Changes take effect as soon as practical in the implementation.

This variable indicates whether support for PMKSA Caching with MAC randomization is supported by the STA.

.

 ::= { dot11StationConfigEntry <ANA>}

DEFVAL { false }