IEEE P802.11
Wireless LANs

|  |
| --- |
| Identifier Privacy Support |
| Date: May 22, 2020 |
| Author(s): |
| Name | Affiliation | Address | Phone | email |
| Nehru Bhandaru | Broadcom Inc. | 250 Innovation Drive, San Jose CA 95134 | +1 408 391 2159 | nehru.bhandaru@broadcom.com |
| Thomas Derham | Broadcom Inc. | 16340 W Bernardo Dr, San Diego CA 92127 |  | thomas.derham@broadcom.com |

Abstract

Privacy is becoming an increasingly important topic, and there is renewed interest in providing privacy for identifiers used in 802.11 protocol messages. Discussion related to privacy have arisen in TGm in various contexts, including SAE password identifiers and PMKIDs. This submission proposes a public-key based scheme for identifier privacy, which covers both these identifiers and is also extensible to protect other identifiers and quasi-identifiers in the future.

**References**

[1] IEEE P802.11-REVmd™/D3.2, March 2020

[2] SEC1 – Elliptic Curve Cryptography - <https://www.secg.org/sec1-v2.pdf>

[3] 11-20/0543r2 – Protected Password Identifiers for Privacy

[4] 11-20/0366r0 – MAC Privacy and PMKSA caching

[5] ETSI TS 133 501 V15.1.0 (2018-07) – 5G Security Architecture and Procedures

[6] IETF RFC 8492 – Secured Password Ciphersuites for TLS

[7] 11-19/0489r0 – Client Privacy Discussion

[8] IETF RFC 8018  - PKCS #5: Password-Based Cryptography Specification Version 2.1

**Revision Notes**

11-20-0746r0 – Initial Revision

11-20-0746r1 – Revision based on feedback on r0. We’d like to acknowledge helpful comments from Dan Harkins that enabled us to improve the submission.

* Update on downgrade protection
* An unauthenticated mechanism to retrieve **idpk**
* Update to table 9-43 presence of fields/elements in authentication frames
* Simplified Nonce construction
* Remove use of term pseudonym for identifiers used only once
* Cut and paste attack protection
* Update on computational/frame overhead
* Mechanism to minimize overhead by requesting **idpk** when needed using IDP MIC element Control field

**Introduction**

Privacy is an increasingly important topic and there is renewed interest in providing privacy for identifiers used in 802.11 protocol messages. For example, document 11-20/0543r2 proposes a mechanism to protect SAE password identifiers, while 11-20/0336r0 outlines a way to protect PMKIDs from being trackable. During discussion of the topic during recent TGm calls, some interest was expressed in a general mechanism for protecting identifier privacy. To that end, this submission proposes a public-key based scheme for identifier privacy, which initially supports protection of SAE password identifiers and PMKIDs. It is extensible to protect other identifiers and quasi-identifiers in the future.

*Comment 4731: “The Password Identifier element is included in the unprotected authentication frame. It may violate the privacy of users (household). For example, it exposes a group of devices and number of devices that are sharing the same password. Particularly, when these devices belongs to the same household (apartment) in an apartment building, it violates the privacy of users/residents.”*

*Commentor Proposes: “Delete the referenced subclause”*

The proposed changes in this submission lay out a framework and direction for identifier privacy as a (*Revise*) resolution for CID 4731.

**Discussion**

Various information elements sent in 802.11 frames might be used for tracking users. Recent discussions have focused on the SAE password identifier carried in SAE authentication frames, and the PMKID sent when PMKSA caching is used. There may be other information sent in 802.11 frames that contains other identifiers or that could act as quasi-identifiers, and may also need protection e.g. public key elements, ERP/FILS wrapped data element, possibly vendor specific elements etc. Some general discussion of the client privacy topic and options occurred in TGm (see 11-19-0489r0) without a specific proposal.

As highlighted in 11-20/0543r2, the introduction of new identifiers in 802.11, such as the SAE password identifier, can actually provide opportunities to improve client privacy compared to legacy approaches - e.g. SAE password identifiers can be used to map users to different VLANs, instead of using different SSIDs/VAPs - which cannot be easily protected due to the unprotected binding of SSID and BSSID being advertised in Beacon and/or Probe Response frames.

Some classical solutions to identifier privacy use temporary identifiers, aka pseudonyms. For example, in 3GPP access technology since GSM, a temporary identifier (TMSI) is used in place of the permanent identifier (IMSI) and is updated frequently. If the device connects to a 3GPP 3G/4G core over WLAN using EAP-AKA, another pseudonym or fast re-authentication temporary identifier can be used.

However, since in general the configuration of user/device identities across an 802.11 network is often decentralized (e.g. configured on each AP, as opposed to a central RADIUS server), temporary identifiers can be a less attractive solution since they can imply significant synchronization burden across APs in a network. The need for a stateless approach is also highlighted in 11-20/543r2. That said, temporary identities might still play a useful role in cases where the identifier is short-term (say a single protocol exchange) and its scope is pairwise between a STA and AP.

A second issue with the use of temporary identities is the lack of protection prior to a temporary identifier being established between the client and network. The nature of 802.11 deployments and use cases means that “first connections” to networks are quite common. Further, for some types of identifier and use cases, the identifier might contain explicit PII such as the user’s real name, or have meaning outside the context of the network, such as the user’s account ID with a network service provider.

A third issue with the use of temporary identifiers is the possibility of out-of-sync state between the client and network, e.g. due to failed delivery of an updated temporary identity or use of volatile storage for identifiers that does not survive device reboot. Workarounds for this issue often result in vulnerability to downgrade attack, e.g. EAP-AKA liberal peer mode.

3GPP has worked on enhancements to subscriber identity protection for 5G [5], in particular to mitigate active “IMSI catcher” downgrade attacks. The resulting solution encrypts the IMSI/SUPI using public-key based on ECIES [2]. A similar scheme is specified in IETF RFC 8492 [6] to protect identities.

**General**

The solution proposed below takes a similar approach, and has the following characteristics:

* Assumes that network public key is constant for the lifetime of the network
* Supports protection against passive and/or active attacks on identifiers in the “first connection” to the network
	+ Assumes STA obtains the network’s public key via some in-band or out-of-band mechanism prior to first connection. Mitigation of active attacks is possible if the public key is obtained via trusted means. This proposal includes an unauthenticated scheme using public action frames.
* Protection against both passive and active attacks on identifiers in subsequent connections to the network
	+ This proposal provides a protected mechanism for STA to request the network public key at association, this protects against active and passive attacks on subsequent connections to the network. Public key needs to be distributed only once and not at every association.
* Retries of failed attempts cannot be tracked by observing the protected identifier(s) alone. Integrity protection for status fields in responses.
* Minimizing state on AP as well as STA – similar to 11-20/0543r2
	+ STA stores a single public key associated with each network (SSID), e.g. as part of network profile
	+ AP stores a single ECC key pair for its BSS, same key pair configured on all APs in ESS, distribution across APs is out-of-scope
* Support for Password Identifier and PMK identifier privacy
* Extensible to support any identifier carried in IEs – privacy for other attributes avoiding fingerprinting.

**Privacy Downgrade**

An attacker might use an evil twin AP that sends Beacons without including the Identifier Privacy capability when the real AP does. Alternatively, an attacker might selectively block messages that have privacy protection in either direction and therefore cause the non-AP STA to fail the connection attempt when using identifier protection – say, by assuming an MITM position. Since the privacy public key is assumed constant for the lifetime of the network, once the non-AP STA has obtained an idpk for the network, it can assume it is valid for all APs in that network and implement a policy to avoid privacy downgrade simply by not falling-back to sending frames without privacy protection.

In the event that the non-AP STA makes multiple attempts to send a frame with privacy protection, one feature that is specific to this proposal is that the non-AP STA can use a different ephemeral key (along with a randomized MAC address) for each attempt, and hence make it harder for an attacker to correlate those attempts from the same STA. On the contrary, in schemes that use a temporary identifier - such as an identifier encrypted by the network, the same temporary identifier would be sent each time and an attacker might use this information to identify multiple attempts by the same non-AP STA in a targeted active attack to try to reveal its real identifier.

**Early downgrade detection in authentication exchanges**

Since fixed fields of (SAE) Authentication frames, including Status Code, are included in the MIC calculation, this can increase resistance to an active attack in which attacker simply injects a response frame with bad status code and might cause the protocol to fail.

**Denial of Service**

This submission does not provide additional denial of service protections. For example, an attacker can send a privacy protected frame to an AP, and force that AP to perform an ECDH operation which consists of one EC scalar multiplication. This is similar to 0543r2 which requires one EC scalar multiplication to generate and re-derive the PT value used w/ H2E method. Future work could consider additional denial of service protection based on proof-of-work concepts.

**Frame Overhead**

* Few IEs – IDP key element, IDP MIC element, IDP KDE
* Key element and MIC element have ASN.1 key in compressed form 59 octets for ECP-256 curve.
* IDP MIC element has additional fields – minimal length to protect 1 element id
	+ 59 + 25 = 84 octets for P-256
* Can consider alternative IANA id + octet string format instead of ASN.1 to reduce this to 35 octets if desired. That would bring the total to around 60 octets. The standard format in RFC 5480 using ASN.1 is preferable.
* Additional overhead is 1 octet for each element protected, and 1 additional octet for extension elements or 3 octets for vendor specific elements.
* The IDP public key need not be distributed on every 4-way, For ECP-256 this is distributed in KDE of 65 octets (59 + 6)

**Computational Overhead**

* Bulk of additional computation is one EC scalar operation to derive the symmetric key on STA and AP
* Symmetric key and DH key pairs are reused in a given exchange

**Slicing attack protection for password identifier privacy**

An attacker might attempt to track a user by correlating frame exchanges over time. For example, the attacker might record a temporary or encrypted identifier that was passively recorded from a previous exchange, and then act as an active MITM to insert the old protected identifier into the new exchange in an attempt to determine if the same real identifier is being used based on success or failure of the exchange. In document 0543r2, such attack is mitigated by using the encrypted password identifier for derivation of ‘PT’ that is used to generate PWE in the SAE protocol. In this submission, similar protection is provided by including the scalar values in SAE commit messages in the encrypted payload. By encrypting the Scalar values (rather than just including them in the MIC), this also prevents a passive observer from being able to derive the real PMKID.

**Proposed Changes to 11md Draft 3.2 – TGm Editor Instructions in red.**

***Add the following to 2. Normative References p152.51***

Standards for Efficient Cryptography Group, "SEC 1: Elliptic Curve Cryptography", version 2.0, May 2009, available from [www.secg.org](http://www.secg.org).

IETF RFC 8018  - PKCS #5: Password-Based Cryptography Specification Version 2.1

***Add the following to 3. Definitions, … p163.1***

**identifier privacy key**: a key used to protect the protect the privacy of identifiers in management frames that are not encrypted. (**idpk**)

**identifier privacy key element:** An information element with a key to protect identifier privacy (**idpke**)

**identifier privacy mic element:** An information element used to protect the integrity of private identifiers (**idpme**)

***Change* 5.1.2 *Security Services as follows* p301.41**

…

BIP provides message integrity and access control for group addressed robust Management frames.

During the authentication exchange, both parties exchange authentication information as described in

Clause 12 (Security) and Clause 13 (Fast BSS transition).

Identifier privacy service provides for protecting identifiable information exchanged in authentication and association messages that can be used for tracking a user or a device.

…

***Change Table 9-321—Extended RSN Capabilities field(#2715) 1462.23 as follows to add support for RSNXE capability for identifier protection***

|  |  |  |
| --- | --- | --- |
| … | … | … |
| ~~(M137)6–~~~~(8n – 1)~~ <ANA-ips> | ~~Reserved~~ Identifier privacy supported  |  |
| <ANA-ips> + 1 – (8xn – 1) | Reserved |  |

***Add to Table 9-94—Element IDs the IDPK Element 993.13***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| … | … | … |  |  |
| Identifier Privacy Key Element (see 9.4.2.xx (Identifier Privacy Key Element)) | 255 | <ANA-idpke> | no | yes |
| Identifier Privacy MIC element (see 9.4.2.yy (Identifier Privacy MIC element)) | 255 | <ANA-idpme> | yes | No |
| Reserved | 255 | ~~94~~ <ANA-privmac>+1 - 255 |  |  |

***At the end of 9.4.2 section on elements add the following elements p1465.51***

**9.4.2.xx Identifier Privacy Key Element**

The Identifier Privacy Key element is used to carry the privacy key used to protect identifiers in 802.11 protocol messages. The format of an Identifier Privacy Key element is shown in Figure 9-xxx (Identifier Privacy Key element format)

|  |  |  |  |
| --- | --- | --- | --- |
|  Element ID  | Length  | Element ID Extension | IDPK |

Octets: 1 1 1 variable

**Figure 9-xxx—** **Identifier Privacy Key element format**

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1 (General).

IDPK corresponds to an ECC public key encoded as DER of ASN.1 SubjectPublicKeyInfo according to IETF RFC 5480 in the compressed format.

**9.4.2.yy Identifier Privacy MIC Element**

The Identifier Privacy MIC element is used to carry the MIC related information used to protect identifiers in 802.11 protocol messages. The format of an Identifier Privacy MIC element is shown in Figure 9-yyy (Identifier Privacy MIC element format)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Element ID | Length | Element ID Extension | Control | Key Counter | ProtectedElement IDs Length | ProtectedElement IDs | Ephemeral Public Key Length | Ephemeral Public Key | MIC |

 Octets: 1. 1 1 1 1 1 variable 1 variable 16

**Figure 9-xxx—** **Identifier Privacy MIC element format**

The Element ID, Length, and Element ID Extension fields are defined in 9.4.2.1 (General).

The Control field consists of bits where

* Bit 0 is the Key Request subfield. When included by a non-AP STA, it is set to 1 to indicate **idpk** is requested and not set to 0 to indicate it not requested. Key Request subfield is reserved in frames from an AP.
* Bits 1 through 7 are reserved.

Key Counter field contains the value of the counter associated with the Ephemeral Public key

Protected Element IDs Length field contains the length of the Protected Element IDs field that follows.

Protected Element IDs field consists of one or more Element IDs whose data fields are protected using identifier privacy mechanisms. When Element ID is 255, the next octet is the corresponding Element ID extension. When Element ID is 221, the next three octets correspond to the vendor OUI. Element IDs corresponding to the Identifier Privacy MIC element and the Fragment element shall not be included in this field. When multiple elements matching a given Protected Element ID in the Protected Element IDs field are present in a frame, all of the matching elements in the frame are protected.

Ephemeral Public Key Length field is the length of the Ephemeral Public Key field that follows.

Ephemeral Public Key field is the ephemeral ECC public key that combined with the Identifier Privacy key of the AP using ECIES (see to derive the symmetric key used to protect the body of the elements specified in the MIC element. It is encoded as DER of ASN.1 SubjectPublicKeyInfo according to IETF RFC 5480 in the compressed format..

MIC field contains the MIC generated by the identifier privacy mechanism to protect the identifiers in the data fields of the elements in the frame specified in the Protected Element IDs field.

***Add optional elements at the end of authentication frame body in Table 9-42—Authentication frame body p884.41 – before the Last vendor specific element(s)***

|  |  |  |
| --- | --- | --- |
| <ANA-IDPME-Order> | Identifier Privacy MIC element  | The Identifier Privacy MIC element is present only incertain Authentication frames as defined in Table 9-43 (Presenceof fields and elements in Authentication frames) |

***Modify Table 9-43—Presence of fields and elements in Authentication frames*** ***– as follows***

|  |
| --- |
| * Presence of fields and elements in Authentication frames
 |
| Authentication algorithm | Authentication transaction sequence number | Status code | (#2528)Presence of fields and elements from order 4 onwards |
| Open System | 1 | Reserved | Not present |
| Open System | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | Not present |
| Open System | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) is present |
| Shared Key | 1 | Reserved | Not present |
| Shared Key | 2 | Any | The (#4279)Challenge Text element is present |
| Shared Key | 3 | Reserved | The (#4279)Challenge Text element is present |
| Shared Key | 4 | Any | Not present |
| FT | 1 | Reserved | The Mobility Domain element is present. The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true.Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection |
| FT | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | The Mobility Domain element is present if the Status Code field is 0. The Fast BSS Transition element and RSNEs are present if the Status Code field is 0 and dot11RSNAActivated is true.Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection |
| FT | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) is presentIdentifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection |
| FT | 3 | Reserved | The Mobility Domain element is present. The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true.The RIC element is optionally present.Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection |
| FT | 4 | Any | The Mobility Domain element is present if the Status Code field is 0. The Fast BSS Transition element and RSNEs are present if dot11RSNAActivated is true.The RIC element is optionally present if the Status Code field is 0. The TIE (reassociation deadline) is present if a RIC element is present.Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| SAE | 1 | Any | (Ed)The Scalar field(#2531) is present if the Status Code field is zero or 126(M137).(#2471)(Ed)The FFE field(#2531) is present if the Status Code field is zero or 126(M137).(#4726)(#2530)(Ed)When the hunting-and-pecking method is used to drive the PWE, the Anti-Clogging Token field(#2534) is present if the Status Code field is ANTI\_CLOGGING\_TOKEN\_REQUIRED(#4415)(M137) or if the Authentication frame is in response to a previous rejection with the Status Code field equal to ANTI\_CLOGGING\_TOKEN\_REQUIRED(#4415)(M137).(Ed)The Finite Cyclic Group field(#2531) is present if the Status Code field is zero, ANTI\_CLOGGING\_TOKEN\_REQUIRED(#4415), 77 or 126(M137).(M104)(#2530)(M41)(Ed)The Password Identifier element is optionally present if the Status Code field is zero, 123 or 126(M137)(Ed).The Rejected Groups element is (#4652)present if the Status Code (#4726)field is 126.(M137)(#4726)When the hash-to-element method is used to derive the PWE, the Anti-Clogging Token Container element is present if the Status Code field is ANTI\_CLOGGING\_TOKEN\_REQUIRED(#4415) or if the Authentication frame is in response to a previous rejection with the Status Code field equal to ANTI\_CLOGGING\_TOKEN\_REQUIRED(#4415).Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for Password Identifier or PMK Identifier protection. |
| SAE | 2 | Not REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | (#2530)The Send-Confirm field is present. (#2530)The Confirm field is present. |
| SAE | 2 | REJECTED\_WITH\_SUGGESTED\_BSS\_TRANSITION | One or more Neighbor Report element(s) are present |
| (11ai)FILS Shared Key authentication without PFS | 1 | Reserved | The RSNE is present.The MDE is present if the FILS authentication is used for FT initial mobility domain association.The FILS Nonce element is present.The FILS Session element is present.(Ed)The FILS Wrapped Data element is present.Identifier Privacy MIC element is optionally present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| (11ai)FILS Shared Key authentication without PFS | 2 | Status | The RSNE is present.The MDE and the FTE are present (#2530)if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association.The FILS Nonce element is present (#2530)if the Status Code field is 0.The FILS Session element is present (#2530)if the Status Code field is 0.The FILS Wrapped Data element is present (#2530)if the Status Code field is 0.The Association Delay Info element is present (#2530)if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.Identifier Privacy MIC element is present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| (11ai)FILS Shared Key authentication with PFS | 1 | Reserved | The Finite Cyclic Group field is present.The FFE field is present.The RSNE is present.(Ed)The MDE is present if the FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present.The FILS Session element is present.The FILS Wrapped Data element is present.Identifier Privacy MIC element is present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| (11ai)FILS Shared Key authentication with PFS | 2 | Status | The Finite Cyclic Group is present (#2530)if the Status Code field is 0.The FFE field is present (#2530)if the Status Code field is 0.The RSNE is present.The MDE and the FTE are present (#2530)if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present (#2530)if the Status Code field is 0.The FILS Session element is present (#2530)if the Status Code field is 0.The FILS Wrapped Data element is present (#2530)if the Status Code field is 0.The Association Delay Info element is present (#2530)if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.Identifier Privacy MIC element is present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| (11ai)FILS Public Key authentication | 1 | Reserved | The Finite Cyclic Group field is present.The FFE field is present.The RSNE is present.The MDE is present if the FILS authentication is used for FT initial mobility domain association. The FILS Nonce element is present.The FILS Session element is present.Identifier Privacy MIC element is present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |
| (11ai)FILS Public Key authentication | 2 | Status | The Finite Cyclic Group is present (#2530)if the Status Code field is 0.The FFE field is present (#2530)if the Status Code field is 0.The RSNE is present.The MDE and the FTE are present (#2530)if the Status Code field is 0 and FILS authentication is used for FT initial mobility domain association.The FILS Nonce element is present (#2530)if the Status Code field is 0.The FILS Session element is present (#2530)if the Status Code field is 0.The Association Delay Info element is present (#2530)if the Status Code field is 0 and the AP expects that the (Re)Association Response frame will be transmitted more than 1 TU after the (Re)Association Request frame.Identifier Privacy MIC element is present when using Identifier Privacy Service (11.xx Identifier Privacy Service) for PMKID protection. |

***Add optional elements at the end of association request frame body in Table 9-36—Authentication frame body p864.29 – before the Last vendor specific element(s)***

|  |  |  |
| --- | --- | --- |
| <ANA-IDPME-Order> | Identifier Privacy MIC element  | Identifier Privacy MIC element is optionally present in association request frames from non-AP STA that chooses to protect identifiers using Identifier Privacy Service (11.xx Identifier Privacy Service) |

***Add IDP MIC element as an optional element at the end of reassociation request frame body in Table 9-38—Reassociation request frame body p878.38 – before the Last vendor specific element(s)***

|  |  |  |
| --- | --- | --- |
| <ANA-IDPME-Order> | Identifier Privacy MIC element  | Identifier Privacy MIC element is optionally present in reassociation request frames from non-AP STA that chooses to protect identifiers using Identifier Privacy Service (11.xx Identifier Privacy Service) |

***Add IDP Key and IDP MIC elements as optional elements at the end of association response frame body in Table 9-37—Association Response frame body (continued) p867.40– before the Last vendor specific element(s)***

|  |  |  |
| --- | --- | --- |
| <ANA-IPDKE-Order> | Identifier Privacy Key element  | Identifier Privacy key element is optionally present in FILS association response frames when dot11IdentifierPrivacySupported is true on the AP |
| <ANA-IDPME-Order> | Identifier Privacy MIC element  | Identifier Privacy MIC element is optionally present in frames from an AP that protects identifiers using Identifier Privacy Service (11.xx Identifier Privacy Service) |

***Add IDP Key and IDP MIC elements as optional elements at the end of reassociation response frame body in Table*** 9-39—Reassociation Response frame body ***(continued) 874.17 – before the Last vendor specific element(s)***

|  |  |  |
| --- | --- | --- |
| <ANA-IPDKE-Order> | Identifier Privacy Key element  | Identifier Privacy key element is optionally present in FILS association response frames when dot11IdentifierPrivacySupported is true on the AP |
| <ANA-IDPME-Order> | Identifier Privacy MIC element  | Identifier Privacy MIC element is optionally present in frames from an AP that protects identifiers using Identifier Privacy Service (11.xx Identifier Privacy Service) |

***Add entry at the end of Table 9-363 (Public Action field values) as follows and update Reserved range accordingly: Public Action field value: <ANA>, Description: Identifier Privacy Public Key.***

***Add the following subclause to end of 9.6.7:***

9.6.7.x.x Identifier Privacy Public Key frame

The Identifier Privacy Public Key frame is transmitted by a STA to provide its identifier privacy public key to peer STAs and to request the peer’s identifier privacy public key. The format of the Public Key frame Action field is defined in Figure 9-887 (Public Key frame Action field format).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Category | Public Action | Identifier Privacy Public Key Frame Usage | IDPK(optional) |
| Octets: | 1 | 1 | 1 | variable |

**Figure 9-xxx - Identifier Privacy Public Key frame Action field format**

The Category field is defined in 9.4.1.11 (Action field).

The Public Action field is defined in 9.6.7.1 (Public Action frames).

The Identifier Privacy Public Key Frame Usage field is set to 0 (Request) when the frame is used to request a public key, or 1 (Response) when the frame is used to respond to a request. Values 2 to 255 are reserved.

When the Identifier Privacy Public Key Frame Usage field indicates Request, the IDPK field is not present.

When the Identifier Privacy Public Key Frame Usage field indicates Response, the IDPK field is present, and contains the identifier privacy public key of the STA that is sending this frame, encoded as DER of ASN.1 SubjectPublicKeyInfo according to IETF RFC 5480 in the compressed format.

The use of the Identifier Privacy Public Key frame is described in 12.6.xx (Identifier Privacy Protection).

***Add at the end of clause 11, subclause 11.xx p2536.40 as follows.***

**11.xx Identifier Privacy Service**

In an infrastructure BSS, Identifier Privacy Service provide protection for privacy of sensitive identifiers that may be used for tracking devices or users.

When dot11IdentifierPrivacySupported is true on a AP and the AP has an **idpk**, the AP shall advertise the capability in RSNXE by setting the Identifier Privacy supported field to 1. Otherwise it sets the field in RSNXE to 0.

When dot11IdentifierPrivacySupported is true on a non-AP STA, the STA shall advertise the capability in RSNXE by setting the Identifier Privacy supported field to 1. Otherwise it sets the field in RSNXE to 0.

When dot11IdentifierPrivacySupported is true on a non-AP STA, the non-AP STA *may* protect the privacy of the identifiers in *management frame elements* sent to an AP supporting Identifier Privacy using the **idpk** it receives using a mechanism specified in 12.6.xx (12.6.xx Identifier Privacy Protection)

***Add subclause 12.6.xx p2634.26 for identifier privacy***

**12.6.xx Identifier Privacy Protection**

**idpk** is an ECC public key that is shared among all of the STAs of an ESS. The corresponding private key **idpk-priv** is only shared among the APs of the ESS using a mechanism outside the scope of this specification. The ECC group chosen for **idpk** should be such that it is supported by all of the APs and non-AP STAs of the ESS.

An AP at which dot11IdentifierPrivacySupported is true shall provide the **idpk** to non-AP STAsin an RSNA by including it as part of PTKSA establishment by including an Identifier Privacy KDE in M3 of the 4-way handshake or IDPK element in the FILS association response frame to non-AP STAs in the BSS that have indicated the capability in RSNXE.

To allow identifier protection of the first interaction with the ESS, the **idpk** can also be distributed to non-AP STAs of an ESS using the Identifier Privacy Public Key action frame exchange, or in a manner that is out of scope for this specification.

The Identifier Privacy Public Key action frame exchange uses the Identifier Privacy Public Key action frame defined in 9.6.7.x.x, which is an unprotected Public Action frame. A STA at which dot11IdentifierPrivacySupported is true may send an Identifier Privacy Public Key action frame with Identifier Privacy Public Key Frame Usage subfield set to 0 (Request) to an AP that advertises support for Identifier Privacy Protection in RSNXE.

An AP at which dot11IdentifierPrivacySupported is true, on reception of an Identifier Privacy Public Key action frame with Identifier Privacy Public Key Frame Usage subfield equal to 0 (Request), shall respond by sending an Identifier Privacy Public Key action frame with Identifier Privacy Public Key Frame Usage subfield set to 1 (Response), containing the AP’s **idpk** in the IDPK field, to the requesting STA.

A non-AP STA at which dot11IdentifierPrivacySupported is true and has obtained the **idpk** for an ESS may use the key to protect the privacy of identifiers in management frames by including an Identifier Privacy MIC element in the frame. If the **idpk** is obtained by the non-AP STA as part of PTKSA establishment or via a trusted out-of-band mechanism, protection against active attacks on Identifier privacy is provided. If the **idpk** is obtained by the non-AP STA using an unprotected mechanism such as the Identifier Privacy Public Key action frame exchange, only protection against passive attacks on Identifier privacy is provided. The Identifier privacy MIC element is not included in any robust management frame.

When a non-AP STA *chooses* to protect identifiers in non-robust management frames, it generates a fresh ephemeral ECC key pair with the same ECC group parameters as that of **idpk** and derives a shared key *sk* (see 12.6.xx.1 Identifier Privacy encryption) to protect the privacy of selected identifiers. When the ephemeral key is created, the non-AP STA also initializes a 1-octet counter *sk-counter* with a value 0. The *sk-counter* value is incremented by 1 each time prior to using it to construct a Nonce to protect a frame.

To protect a frame, the non-AP STA constructs and includes in the frame an Identifier Privacy MIC element (see 9.4.2.yy Identifier Privacy MIC Element) that is constructed as follows

* Control field with Key Request bit set to 1 if the STA is requesting **idpk** from the AP. The STA may request **idpk** if it possesses an **idpk** that is currently not trusted.
* Key counter field is set to the *sk-counter* value.
* Protected Element IDs Length field is set to length in octets of the following Protected Element IDs field.
* Protected Element IDs field set corresponding to the elements containing private identifiers.
* Ephemeral Public key length field is set to the length of the ephemeral key
* Ephemeral Public key field is set to the ephemeral public key that was generated
* MIC field is set to the value of the cryptographic tag that is generated as specified in 12.6.xx.1 (12.6.xx.1 Identifier Privacy encryption)

When an AP or a non-AP STA at which dot11IdentifierPrivacySupported is true receives a management frame with Identify Privacy MIC element, it shall validate the element as follows.

* Generate the shared key as specified in 12.6.xx.2(12.6.xx.2 Identifier Privacy Key decryption)
* Decrypt the elements that include protected identifier fields as specified in 12.6.xx.2 (12.6.xx.2 Identifier Privacy decryption). Decryption process validates that the MIC field matches the computed cryptographic tag.

A non-AP STA that has included an Identifier Privacy MIC element in a frame and does not receive a valid Identifier Privacy MIC element in the corresponding response frame shall terminate further processing of the response frame. A non-AP STA that has included an Identifier Privacy MIC element in a frame with the Key Request subfield of the Control field of the element and does not receive the **idpk** in the subsequent 4-way handshake or FILS (re)association response, it shall deauthenticate. A non-AP STA that is Identifier Privacy capable (as indicated in RSNXE) and has not included the Identifier Privacy MIC element in any frame during the exchange but failed to receive the **idpk** from a Identifier Privacy capable AP in the subsequent 4-way handshake or FILS (re)association response, it shall deauthenticate.

If a non-AP STA receives an **idpk** from an AP in a 4-way handshake or FILS (re) association response, it shall compare the received **idpk** with any **idpk** it has received earlier in an untrusted exchange and discard the untrusted **idpk** if there is a mismatch. The mismatch may happen with an MITM attacker who may own the private key corresponding to the untrusted **idpk**.

An AP or non-AP STA shall terminate the processing of the management frame if the decryption fails. The decryption and validation shall occur prior to any of the services requiring the use of the contents of the elements that are protected.

A non-AP STA at which dot11IdentifierPrivacySupported is set to true upon successful (re)association with an AP that supports Identifier privacy shall protect any of the PMK identifiers in EAPOL-Key message M2 (see 12.7.6.3 4-way handshake message 2). The PMKID or PMKR1Name (FT) included in these messages contain a protected PMK Identifier generated as specified in 12.6.xx.4 (12.6.xx.4 Protected PMK identifier generation). The protected PMK identifier is used in computing the EAPOL MIC instead of the actual PMKID or PMKR1Name.

An AP at which dot11IdentifierPrivacySupported is true and has received a valid Identifier privacy MIC element in the preceding (re) association request shall protect any of the PMK identifiers in EAPOL-Key message M1 (see 12.7.6.2 4-way handshake message 1). The PMKID or PMKR1Name (FT) included in these messages contain the PMK identifier pseudonym generated as specified in 12.6.xx.4 (12.6.xx.4 Protected PMK identifier generation)

An AP and a non-AP STA shall decrypt the protected PMK identifier to obtain the real PMKID or PMKR1Name, immediately after the EAPOL MIC is validated. Only the real PMKID or PMKR1Name is used for further processing.

If an AP at which dot11IdentifierPrivacySupported is true received a valid Identifier Privacy MIC element in an authentication or (re) association request frame, it shall include the Identifier Privacy MIC element in the corresponding response frame. The Identifier Privacy MIC element (see 9.4.2.yy Identifier Privacy MIC Element) is constructed as follows

* Control field is set to 0
* Key counter field is set to the value in the Identifier Privacy MIC element from the non-AP STA.
* Protected Element IDs Length field is set to length in octets of the following Protected Element IDs field.
* Protected Element IDs field is set corresponding to the elements containing private identifiers.
* Ephemeral Public key length field is set to 0
* Ephemeral Public key field is empty
* MIC field is set to the value of the cryptographic tag that is generated as specified in 12.6.xx.1 (12.6.xx.1 Identifier Privacy encryption)

A non-AP STA desiring password identifier privacy includes the IDP MIC element in the frame in which a Password identifier element (9.4.2.216 (Password Identifier Element)) is present and includes the Element ID of Password identifier element in the Element IDs field of the IDP MIC element.

If and only if Identifier Privacy protection is used for a password identifier, in order to prevent an attacker from determining the identifier length, the non-AP STA shall pad the identifier field of the Password identifier element with a random number of octets before Identifier Privacy encryption. The length of the pad shall be between 1 and (254 – *l-pwd-id*) octets, where *l-pwd-id* is the length, in octets, of the unpadded identifier. An AP receiving the padded identifier shall remove the padding before use in SAE protocol after Identifier Privacy decryption. The padding scheme specified in IETF RFC 8018 shall be used.

A non-AP STA desiring PMK identifier privacy includes the IDP MIC element in the frame in which there is an element or a field containing a PMK identifier and includes the Element ID of that element in the Element IDs field of the IDP MIC element. The IDP MIC element for PMK identifier privacy may be included in

* (re)association request frames and FILS authentication frames where RSNE (9.4.2.24 (RSNE)) may contain PMK identifiers when PMKSA caching is used.
* Fast BSS Transition element (FTE) included in FT authentication frames (see 13.5 FT Protocol)
* SAE Authentication frames in which a Scalar field is present. The Scalar field is used to derive the PMKID when a PMKSA is established using SAE.

When an IDP MIC element is included in a frame, the AP shall include the Element ID of any element that contains a PMK identifier in the Element IDs field of the IDP MIC element. PMK identifiers may be included in (re)association response frames for FT and FILS authentication

**12.6.xx.1 Identifier Privacy encryption**

Use ECIES (see SEC 1: Elliptic Curve Cryptography § 5.1) to generate a symmetric key (sk) and encrypt contents of elements with protected identifiers as follows

* Generate Diffie-Hellman shared secret (ss) as follows
	+ On a non-AP STA, use the generated ephemeral *private* key and the **idpk** for the ESS.

On an AP, use the *private* key **idpk-priv** for the ESS and the ephemeral public key from the IDP MIC element’s Ephemeral key field from the non-AP STA

* If the security strength of the ECC group for **idpk** is > 128 then symmetric key length (*sk-len*) is 256, otherwise shared key length is 128
* Derive *sk* of length *sk-len* using KDF-SHA256-Length as the hash function i.e. Length is *sk-len*
	+ With Diffie-Hellman shared secret *ss* as the key
	+ “*Identifier Privacy key expansion*” as the label
	+ Concatenation of receiver MAC address and transmitted MAC address as the context.
* Select Cipher AES-GCM-128 or AES-GCM-256 based on whether sk-len is 128 or 256.
* Using the cipher protect the elements by applying GCM AEAD encryption as follows
	+ Additional authentication data (AAD) set to the concatenation of
		- If the management frame is of type authentication, frame body up to and including the Finite Cyclic Group field (Table 9-42—Authentication frame body)
		- If the management frame is an association response or a reassociation response, the frame body up to and including the AID field (9-37—Association Response frame body , Table 9-39—Reassociation Response frame body)
		- Contents of IDP MIC element excluding the MIC field.
	+ Construct the Nonce as specified in 12.6.xx.3 Identifier Privacy Nonce construction.
	+ Plaintext set to the concatenation of selected data fields in the frame in the order they appear in the frame. i.e. the encryption is done in place. Data field of an IE encompasses the body of the IE present in the information field of an element (Figure 9-145—Element format) or the vendor-specific-content field (Figure 9-290—Vendor Specific element format) if the IE is vendor specific. If a data field of an IE is included, all of the data fields of its Fragment elements (see 9.4.2.188 Fragment element and 10.28.11 Element fragmentation) that immediately follow (if any) are included. Selected data fields are the data fields for the elements specified by the Protected Element IDs of the Identifier Privacy MIC element in the frame. If the frame is an SAE Authentication frame from a non-AP STA, the Scalar field in the frame shall be included in the Plaintext.
	+ Encrypt the plaintext and place the tag in the MIC field of the IDP MIC element

**12.6.xx.2 Identifier Privacy decryption**

Use ECIES (see SEC 1: Elliptic Curve Cryptography § 5.1) to generate a symmetric key (sk) and decrypt contents of elements with protected identifiers as follows

* Generate Diffie-Hellman shared secret (ss) as follows
	+ On a non-AP STA, use the generated ephemeral *private* key and the **idpk** for the ESS.
	+ On an AP, use the *private* key **idpk-priv** for the ESS and the ephemeral public key from the IDP MIC element’s Ephemeral key field from the non-AP STA
* If the security strength of the ECC group for **idpk** is > 128 then symmetric key length (*sk-len*) is 256 bits, otherwise shared key length is 128 bits
* Derive *sk* of length *sk-len* using KDF-SHA256-Length as the hash function i.e. Length is *sk-len*
	+ With Diffie-Hellman shared secret *ss* as the key
	+ “*Identifier Privacy key expansion*” as the label
	+ Concatenation of receiver MAC address and transmitted MAC address as the context.
* Select Cipher AES-GCM-128 or AES-GCM-256 based on whether sk-len is 128 bits or 256 bits.
* Using the cipher protect the elements by applying GCM AEAD decryption as follows
	+ Additional authentication data (AAD) set to the concatenation of
		- If the management frame is of type authentication, frame body up to and including the Finite Cyclic Group field (Table 9-42—Authentication frame body)
		- If the management frame association response or reassociation response, frame body up to and including the AID field (9-37—Association Response frame body , Table 9-39—Reassociation Response frame body)
		- contents of IDP MIC element excluding the MIC field.
	+ Construct the Nonce as specified in 12.6.xx.3 Identifier Privacy Nonce construction.
	+ Ciphertext is set to the concatenation of selected data fields in the frame in the order they appear in the frame. i.e. the decryption is done in place. Data field of an IE encompasses the body of the IE present in the information field of an element (Figure 9-145—Element format) or the vendor-specific-content field (Figure 9-290—Vendor Specific element format) if the IE is vendor specific. If a data field of an IE is included, all of the data fields of its Fragment elements (see 9.4.2.188 Fragment element and 10.28.11 Element fragmentation) that immediately follow (if any) are included. Selected data fields are the data fields for the elements specified by the Protected Element IDs of the Identifier Privacy MIC element in the frame. If the frame is an SAE Authentication frame from a non-AP STA, the Scalar field in the frame shall be included in the Ciphertext.
	+ Decrypt the Ciphertext and if successful populate the IE data fields from which the Ciphertext was derived in order they appear in the frame. Otherwise indicate failure.

**12.6.xx.3 Identifier Privacy Nonce construction**

On a non-AP STA for protection of identifiers in management frames, construct the 12 octet Nonce value whose

* 11 MSB octets are set 0
* 1 LSB octets are set to the value of *s-counter*.

On an AP, for protection of identifiers in management frames, construct the 12 octet Nonce value whose

* 1 MSB octet is set to 0x80
* The next 10 MSB octets are set to 0
* 1 LSB octets set to the value of Key Counter received in the frame from the non-AP STA to which frame being constructed is the response.

For a given ephemeral key, the Nonce value shall not be repeated. A new ephemeral key can be generated by the non-AP STA prior to sending any frame that includes Identifier Privacy MIC element but shall be generated if the Nonce value was used with the ephemeral key.

**12.6.xx.4 Protected PMK Identifier generation**

Protected PMKID generation uses the same underlying AES counter mode used to protect the elements in the management frames. Counter generation specified below avoids any counter reuse.

On a non-AP STA, generate the 16-octet counter (*P-Counter*) is constructed as follows

* If the STA is not an AP, one octet of 0x40, otherwise on an AP, one octet of 0xc0
* Followed by 14 octets of 0 and one octet with a value of
	+ 1 for 4-way handshake message 1
	+ a value of 2 for PMKID in 4-way handshake message 2
	+ a value of 3 for PMKR1Name in 4-way handshake message 2

Compute the protected PMK identifier (PMK\_Identifier which is PMKID or PMKR1Name) as

 protected-PMK\_Identifier = AES-*sk-len* (*sk*, *P-Counter*) $⊕$ PMK\_Identifier

where

* *sk* is the shared key generated as part of Identifier Privacy encryption or decryption.
* *sk-len* is the bit length of *sk* – which is 128 or 256*.*
* *P-Counter* is the counter constructed as above
* PMK\_Identifier is the real PMK identifier

PMK identifier can be obtained from the protected identifier as

 PMK\_Identifier = AES-*sk-len* (*sk*, P-Counter) $⊕$ protected-PMK\_Identifier

***Modify table 12-9 p2652.18 in section 12.7.2 by adding idpk KDE***

 **Table 12-9—KDE selectors**

|  |  |  |
| --- | --- | --- |
|  **OUI** |  **Data type** |  **Meaning** |
|  00-0F-AC |  13 |  OCI KDE |
|  00-0F-AC |  14 |  BIGTK KDE |
|  00-0F-AC |  <ANA-idpk-kde> |  IDPK KDE |
|  00-0F-AC | <ANA-idpk-kde>+1 -255 |  Reserved |
| Other OUI or CID |  Any |  Vendor specific |

***Add at the end of section 12.7.3 IDPK KDE format***

The format of the IDPK KDE is shown in Figure 12-XX (IDPK KDE).

|  |
| --- |
|  IDPK  |

 Octets: (Length – 4)

 **Figure 12-XX—IDPK KDE format**

IDPK corresponds to an ECC public key encoded according to IETF RFC 5480

***Change in 12.7.6.4 4-way handshake message 3 p2664.10 as follows.***

—(#2715)The RSNXE that the Authenticator sent in its Beacon or Probe Response frame, if this

element is present in the Beacon or Probe Response frame that the Authenticator sent.

— Additionally, contains an IDPK KDE if and only if all of the following are true

* dot11IdentifierPrivacySupported is true on the Authenticator
* Supplicant has indicated the Identifier Privacy capability in RSNXE .
* Authenticator has not delivered idpk during the current 802.11 association.
* Either the Supplicant has not included an IDP MIC element in the association request for the current association or the Supplicant has included an IDP MIC element with Key Request subfield set to 1 in the Control field of the association request.

IDPK KDE contains the identifier privacy public key for the ESS.

***Add to MIB definitions p3816.33***

Dot11StationConfigEntry…

{

 …

 dot11IdentifierPrivacyEnabled TruthValue

}

…

***P3859.26***

dot11IdentifierPrivacySupported OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This is a control variable.

It is written by an external management entity.

Changes take effect as soon as practical in the implementation.

This variable indicates whether this STA supports identifier privacy in an RSNA."

DEFVAL {false}

::= { dot11StationConfigEntry 193 }