EAP Method Update
Internet-Draft
Intended status: Standards Track
Expires: 3 September 2026

G. Wang, Ed.
Z. Lei
Huawei Int. Pte Ltd

2 March 2026

Forward Secure Reauthentication in the Extensible Authentication Protocol Method for Authentication and Key Agreement (EAP-AKA')

draft-wang-emu-fs-reauth-00

Abstract

This draft specifies an update to RFC 9678, "Forward Secrecy Extension to the Improved Extensible Authentication Protocol Method for Authentication and Key Agreement (EAP-AKA' FS)", and its predecessors RFC 9048, RFC 5448, and RFC 4187. This update enables forward security of the Transient EAP Keys (TEKs) for protecting EAP packets, which are not in EAP-AKA' FS. Based on this extension, the executions of reauthentication after a full authentication will be unlinkable to each other and then the privacy of end users is enhanced. This udapte is optional to the above standards.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 3 September 2026.

Copyright Notice

Table of Contents

1. Introduction

EAP-AKA (Extensible Authentication Protocol method for 3rd Generation Authentication and Key Agreement) [RFC4187] is a secure authentication method used for mobile devices connecting to networks (like Wi-Fi) using their credentials from SIM/USIM cards. It enables mutual authentication and key exchange between the mobile devices and their mobile network operator. After that, communication data can be securely transmitted between them by using various key materials agreed in EAP-AKA.

EAP-AKA' (Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement) [RFC5448], introduces a new key derivation function, SHA-256 instead of SHA-1. This function is also used to bind the keys derived within the method to the name of the access network. This limits the effects of compromised access network nodes and keys.

Moreover, "Improved Extensible Authentication Protocol Method for 3GPP Mobile Network Authentication and Key Agreement (EAP-AKA')" [RFC9048] specifies the the protocol behavior for both 4G and 5G deployments using EAP-AKA'. For examples, how the Network Name field is constructed in the protocol; how EAP-AKA' use identifiers in 5G; how to define session identifiers and other exported parameters (including the case for fast reauthentication), and how to updates the requirements on generating pseudonym usernames and fast reauthentication identities to ensure identity privacy.

"Forward Secrecy Extension to the Improved Extensible Authentication Protocol Method for Authentication and Key Agreement (EAP-AKA' FS)" [RFC9678] enhances the forward security for the session keys generated as a part of the authentication run in EAP-AKA', by introducing ephemeral Diffie-Hellman key exchange. This prevents an attacker who has gained access to the long-term key from compromising session keys established in the past. However, as noted in Section 7.6 of [RFC9678], K_encr, the key for encrypting reauthentication pseudonym idenities, is not forward secure, as it is generated before ephemeral DH. Therefore, "an adversary compromising the long-term key would be able to link reauthentication protocol runs when pseudonyms are used, within a sequence of runs followed after a full EAP-AKA' authentication. No such linking would be possible across different full authentication runs. If the pseudonym linkage risk is not acceptable, one way to avoid the linkage is to always require full EAP-AKA' authentication."

However, as discussed in [RFC4187], reauthentication is much faster and then benefits to both mobile devices and the network operator. Having full EAP AKA' authentication defeats the purpose of fast reauthentication. This document specifies an update to enhance the forward security for TEKs (incluidng K_encr) in EAP-AKA' FS. Based on this, it is not feasible to link the executions of reauthentication within the session of a full authentication. When this extension is enabled, the privacy of mobile device users are protected against long-term key compromise. This udapte is applicable and optional to all standards specifed in [RFC9678], [RFC9048], [RFC5448], and [RFC4187].

This extension is also applicable and optional to the drafts specified in [I-D.ietf-emu-pqc-eapaka] and [I-D.ietf-emu-hybrid-pqc-eapaka], where the ephemeral DH key exchange is replaced by post-quantum (PQ) KEM and hybrid KEMs [RFC9794], respectively.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Readers are assumed familiar with the terms in EAP-AKA [RFC4187], EAP-AKA' [RFC5448] [RFC9048], and EAP-AKA' FS [RFC9678]. The implication of forward security is discussed in Sections 1 and 4.3 of [RFC9678], and the usage of reauthentication is discussed in Section 5 of [RFC4187], and Sections 6.5.4 and 6.5.5 of [RFC9678].

3. Linkable Reauthentication in EAP-AKA' FS

3.1. Review of EAP-AKA' FS

The normal process of EAP-AKA' FS is briefly reviewd in Figure 1, where AD denotes the 3GPP Authentication Database. Details can be found in Section 5 of [RFC9678].

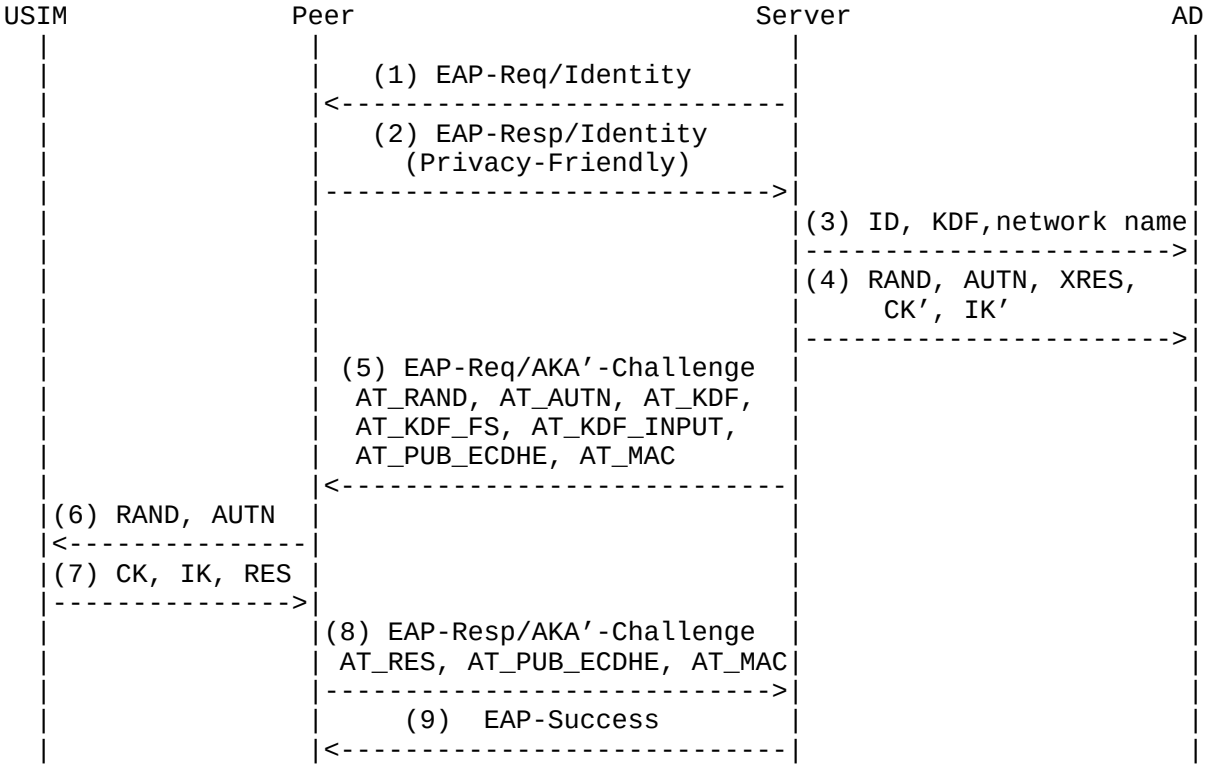

Figure 1. EAP-AKA' FS Authentication Process (Section 5 of RFC 9678)

Key materials are derived in EAP-AKA' FS as shown in Figure 2 (Section 6.3 of [RFC9678]). Note that the TEKs, consisting both K_encr and K_aut, are parts of MK (Master Key). However, MK itself is derived from IK' and CK' without the ephemeral SHARED_SECRET, obtained via running ephemeral DH key exchange. Therefore, both K_encr and K_aut are not forward secure, as they just rely on the security of the long-term key, shared by the peer's USIM and the mobile network operator's AD. Note that IK' and CK' are derived from this long-term key.

```
MK       = PRF'(IK'|CK',"EAP-AKA'"|Identity)
MK_ECDHE = PRF'(IK'|CK'|SHARED_SECRET,"EAP-AKA' FS"|Identity)
K_encr   = MK[0..127]
K_aut    = MK[128..383]
K_re     = MK_ECDHE[0..255]
MSK      = MK_ECDHE[256..767]
EMSK     = MK_ECDHE[768..1279]
```

Figure 2. Key Derivation in EAP-AKA' FS (Section 6.3 of RFC 9678)

In more detail, the ephemeral SHARED_SECRET is generated from ephemeral DH values available in two AT_PUB_ECDHE attributes, exchanged by the peer and server in Steps (5) and (8) in Figure 1. However, K_encr and K_aut are generated by the server after Step (4) and used in Step (5) to protect the info for the peer. And similiarly, they are generated after Step (7) by the peer and used to verify and decrypt ciphtertext sent in Step (5), and used in Step (8) to protect the info for the server. So, the ephemeral SHARED_SECRET is avaialbe later than when K_encr and K_aut are generated and used by the server and the peer. So, in case the long-term key is compromised, K_encr and K_aut will be compromised too.

3.2. Linkable Reauthentication Identities

Figure 3 is a brief review of the reauthentication procedure, which is specified in Section 5.4 of [RFC4187]. Here all attributes with '*' denote that they are encrypted using the encryption key K_encr, and encapsulated in the AT_ENCR_DATA attribute. For offering the peer a new reauthentication identity for the next run, the authenticator generates a pseudonym and uses K_encr to encrypt it in the optional attribute AT_NEXT_REAUTH_ID. At the same time, the authenticator uses integrity key K_aut to produce a MAC in the atttibute AT_MAC in Step (3).

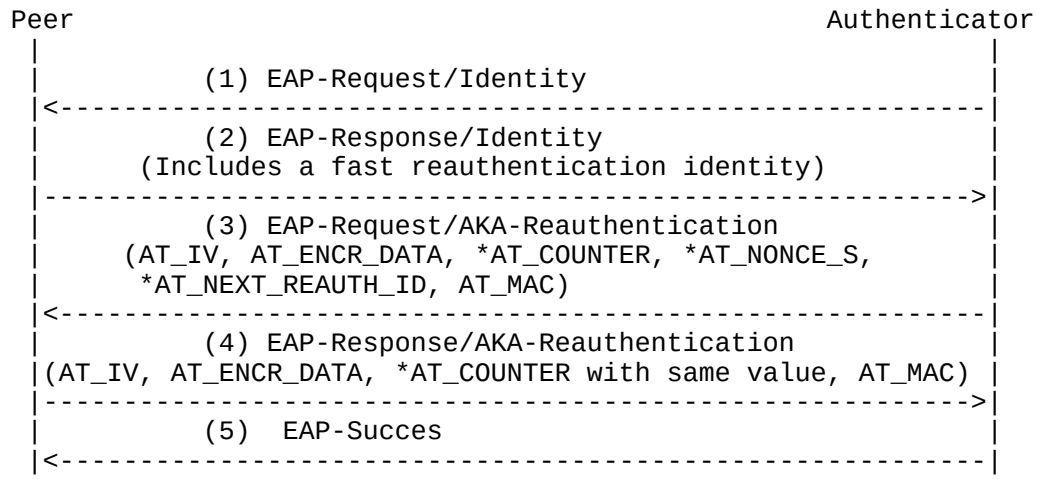

Figure 3. Reauthentication Procedure (Section 5.4 of RFC 4187)

As discussed above, K_encr and K_aut will be compromised if the long-term key is leaked. So, in such case, the attacker with access to the long-term key will be able to decrypt all reauthentication identities delivered from the server to the peer. When these identities are used for reauthentication, the attacker will be able to link these runs of reauthentication, even those reauthentication identities are pseudonyms generated by the server independently.

Morever, even without decrypting the reauthentication identities from the AT_NEXT_REAUTH_ID attributes, the attacker can also link two or more runs of reauthentication via using the integrity key K_aut. Namely, the attacker can check the AT_MAC attributes sent by either the authenticator in Step (3) or the peer in Step (4) to be sure that different runs belong to the same peer, if all these AT_MAC attributes are valid with respect to the given K_aut.

Therefore, to guarantee the privacy of mobile users running reauthentication with compromised long-term key, it is necessary to enhance the forward security of the TEKs, i.e, K_encr and K_aut.

4. Forward Secure Reauthentication

To enable the forward security of K_encr and K_aut, this document specifies a concrete key derivation process given in Figure 4. (EDITOR’S NOTE: Other variants are also available.)

```
MK       = PRF'(IK'|CK',"EAP-AKA'"|Identity)
MK_ECDHE = PRF'(IK'|CK'|SHARED_SECRET,"EAP-AKA' FS"|Identity)
K_encr   = MK[0..127]
K_aut    = MK[128..383]
K_encr'  = MK_ECDHE[0..127]
K_aut'   = MK_ECDHE[128..383]
K_re     = MK_ECDHE[384..639]
MSK      = MK_ECDHE[640..1060]
EMSK     = MK_ECDHE[1061..1633]
```

Figure 4. The Proposed Key Derivation Process

In this process, K_encr and K_aut are updated after completing full EAP-AKA' FS authentication in Figure 1 to forward secure K_encr' and K_aut', which are derived from IK', CK' and SHARED_SECRET, the ephemeral secret. And only K_encr' and K_aut' are used to protect the transmission and usage of reauthenticaton identities in reauthentication procedure in Figure 3.

The whole process will be elaborated later.

5. Security Considerations

Security condiderations will be added later.

6. IANA Considerations

At the time of writing, there are no IANA considerations that may need to be considered.

7. Acknowledgments

To be added later.

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <https://www.rfc-editor.org/info/rfc3748>.

[RFC4187] Arkko, J. and H. Haverinen, "Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)", RFC 4187, DOI 10.17487/RFC4187, January 2006, <https://www.rfc-editor.org/info/rfc4187>.

[RFC5448] Arkko, J., Lehtovirta, V., and P. Eronen, "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')", RFC 5448, DOI 10.17487/RFC5448, May 2009, <https://www.rfc-editor.org/info/rfc5448>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9048] Arkko, J., Lehtovirta, V., Torvinen, V., and P. Eronen, "Improved Extensible Authentication Protocol Method for 3GPP Mobile Network Authentication and Key Agreement (EAP-AKA')", RFC 9048, DOI 10.17487/RFC9048, October 2021, <https://www.rfc-editor.org/info/rfc9048>.

[RFC9678] Arkko, J., Norrman, K., and J. Preu Mattsson, "Forward Secrecy Extension to the Improved Extensible Authentication Protocol Method for Authentication and Key Agreement (EAP-AKA' FS)", RFC 9678, DOI 10.17487/RFC9678, March 2025, <https://www.rfc-editor.org/info/rfc9678>.

[FIPS203] National Institute of Standards and Technology, "FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard", Federal Information Processing Standards Publication , August 2024, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.pdf>.

9. Informative References

[I-D.ietf-emu-hybrid-pqc-eapaka] Banerjee, A. and T. Reddy.K, "Enhancing Security in EAP-AKA' with Hybrid Post-Quantum Cryptography", Work in Progress, Internet-Draft, draft-ietf-emu-hybrid-pqc-eapaka-01, 26 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-emu-hybrid-pqc-eapaka-01>.

[I-D.ietf-emu-pqc-eapaka]
Reddy.K, T. and A. Banerjee, "Post-Quantum Key Encapsulation Mechanisms (PQ KEMs) in EAP-AKA prime", Work in Progress, Internet-Draft, draft-ietf-emu-pqc-eapaka-01, 26 February 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-emu-pqc-eapaka-01>.

[RFC9794] Driscoll, F., Parsons, M., and B. Hale, "Terminology for Post-Quantum Traditional Hybrid Schemes", RFC 9794, DOI 10.17487/RFC9794, June 2025, <https://www.rfc-editor.org/info/rfc9794>.

Authors' Addresses

Guilin Wang (editor)
Huawei Int. Pte Ltd
9 North Buona Vista Drive, #13-01
The Metropolis Tower 1
SINGAPORE 138588
Singapore
Email: wang.guilin@huawei.com

Zhongding Lei
Huawei Int. Pte Ltd
9 North Buona Vista Drive, #13-01
The Metropolis Tower 1
SINGAPORE 138588
Singapore
Email: lei.zhongding@huawei.com