EMU working group
Internet-Draft
Obsoletes: 7170 (if approved)
Updates: 9427 (if approved)
Intended status: Standards Track
Expires: 11 July 2026

A. DeKok
InkBridge Networks

7 January 2026

Tunnel Extensible Authentication Protocol (TEAP) Version 2
draft-dekok-emu-teapv2-02

Abstract

This document defines the Tunnel Extensible Authentication Protocol (TEAP) version 2. It addresses a number of security and interoperability issues in TEAPv1 which was defined in [I-D.ietf-emu-rfc7170bis].

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-dekok-emu-teapv2/.

Discussion of this document takes place on the EMU Working Group mailing list (mailto:emu@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/emu/. Subscribe at https://www.ietf.org/mailman/listinfo/emu/.

Source for this draft and an issue tracker can be found at https://github.com/inkbridgenetworks/teapv2.git.

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 11 July 2026.

Table of Contents

1. Introduction

Tunnel Extensible Authentication Protocol (TEAP) version 1 was first defined in [RFC7170]. However, implementations of that specification were found to have limited interoperability, due to the complexity and under-specification of the cryptographic key deriviations that it defined.

TEAPv1 was updated and clarified in [I-D.ietf-emu-rfc7170bis]. That document described a large amount of potential functionality in the protocol, but also noted in [I-D.ietf-emu-rfc7170bis], Section 5.1 that only a small subset of that functionality was interoperable. In addition, the interoperable parts of the protocol security issues which could potentially allow on-path attackers control control over the data being transported inside of the TLS tunnel.

We do not review the full security issues with TEAPv1 here. Instead, we define TEAPv2, with new and simpler cryptographic key deriviations. These derivations address all of the known issues with TEAPv1.

1.1. Changes from TEAPv1

Most parts of TEAPv1 are unchanged. The message and TLV formats are the same, as are the derivations for the session_key_seed ([I-D.ietf-emu-rfc7170bis], Section 6.1), the Master Session Key (MSK) and the Extended Master Session Key (EMSK) ([I-D.ietf-emu-rfc7170bis], Section 6.4).

The Crypto-Binding TLV [I-D.ietf-emu-rfc7170bis], Section 4.2.13 format is also the same as for TEAPv1, even though its value is now calculated via a different derivation.

The main difference between TEAPv1 and TEAPv2 is in the cryptographic calculations. The changes between TEAPv1 and TEAPv2 simplify the key deriviations, and address issues seen with TEAPv1:

The Crypto-Binding TLV calculation in TEAPv1 had significant differences between theory and practice. This difference was due to complex derivations of multiple keys, some of which were defined ambiguously. This complexity and lack of clarity lead to interoperability problems. TEAPv2 simplifies the key deriviations, reduces the number of intermediate keys, and gives smaller (and therefore clearer) definitions for the derivations.

The result is a protocol which is simpler and more extensible.

1.2. Outline of this Document

This document largely follows the same outline as [I-D.ietf-emu-rfc7170bis]. Where changes from that document are made, the same section titles are used in order to ensure easy comparison of the documents. New sections are added, with new titles. For parts of TEAP which are not mentioned herein, this document makes no changes from [I-D.ietf-emu-rfc7170bis].

1.3. Terminology

Round

A round is one set of inner message exchanges. Each round finishes with an Interim-Result TLV and a Cryptographic-Binding TLV. Other TLVs may also be included in a round.

A round can include multiple inner message exchanges, e.g. EAP-TLS.

This term was used in [I-D.ietf-emu-rfc7170bis], but was not defined in that document.

2. Negotiation

TEAPv2 uses the same version negotiation method as is defined in [I-D.ietf-emu-rfc7170bis], Section 3.1, with the Version field set to two (2) for TEAPv2.

TEAPv2 MUST use TLS 1.3 or later. TEAPv2 MUST NOT use TLS-PSK.

3. Cryptographic Calculations

The crytographic calculations for TEAPv2 have been substantially simplified from those defined in [I-D.ietf-emu-rfc7170bis], Section 6.

3.1. TEAP Authentication Phase 1: Key Derivations

The session key seed is the same as defined in [I-D.ietf-emu-rfc7170bis], Section 6.2.1 for TEAPv2. That definition is reproduced here verbatim:

```
session_key_seed = TLS-Exporter(
                    "EXPORTER: teap session key seed",, 40)
```

3.2. Intermediate Compound Key Derivations

The intermediate key derivation in TEAPv2 proceeds via the following steps:

* Combine the seed with the MSK/EMSK from the current round. If MSK or EMSK is not available, the relevant field is set to zero. The resulting data is the "RoundSeed", which is used to seed the intermediate key derivations.

* At the conclusion of an inner round, call the TLS-Exporter() function ([RFC8446], Section 7.5) with the above data as the "context_value" in order to derive intermediate keying data.

* Split the resulting intermediate keying data into two subkeys. One subkey is used in the seed to the next round. The other subkey is the Compound MAC Key (CMK) which is used to calculate the Crypto-Binding TLV.

The following sections explain how the round seed is created (Section 3.2.1), how the seed is used to derive an intermediate key (Section 3.2.2), how the seed is updated after each round (Section 3.2.3), and finally how the master keys are generated from the final round seed (Section 3.4).

3.2.1. Intermediate Key Seeding

The intermediate compound key deriviations for TEAPv2 depend on the following RoundSeed structure. The RoundSeed structure is defined using the same syntax as is used for TLS [RFC8446]:

```
struct {
    opaque PrevRoundKey[40]
    opaque MSK[32];
    opaque EMSK[32]
 } RoundSeed
```

The RoundSeed structure fields have the following definitions:

PrevRoundKey

A key which ties the current round to the previous round.

MSK

The Master Session Key (MSK) from the inner authentication method.

EMSK

The Extended Master Session Key (EMSK) from the inner authentication method.

At the start of Phase 2, the first 40 octets of the session_key_seed MUST be copied to the PrevRoundKey field of the RoundSeed structure. All other fields MUST be set to zero.

The RoundSeed structure is updated after the DerivedKey structure (Section 3.2.2, below) is calculated. The update process is defined below in Section 3.2.3.

3.2.2. Key Derivation

After a successful inner round, a DerivedKey is calculated. The DerivedKey depends on the current value of RoundSeed via the following calculation.

```
DerivedKey = TLS-Exporter(
             "EXPORTER: TEAPv2 Inner Methods Compound Keys",
             RoundSeed, 72)
```

The DerivedKey is 72 octets in length, and is assigned to the following data structure:

```
struct {
    opaque RoundKey[40];
    opaque CMK[32]
} DerivedKey
```

The DerivedKey structure fields have the following definitions:

RoundKey

A key which is associated with this round.

This field is copied to the PrevRoundKey field of the RoundSeed structure.

CMK

The Compound MAC Key (CMK)

The CMK is mixed with with data from the TEAP negotiation to create the Compound-MAC ({#computing-compound-mac}) which is used in the Crypto-Binding TLV.

3.2.3. Updating RoundSeed

The RoundSeed structure MUST be updated after the DerivedKey structure has been calculated. Each field is updated via the process defined below.

Note that the RoundSeed value is updated at the end of each inner round, before the Crypto-Binding TLV is calculated. This update ensures that the final master key calculation (Section 3.4, below) uses a different value for the RoundSeed structure than was used in the last inner exchange.

PrevRoundKey

The RoundKey field from the DerivedKey structure is copied to the PrevRoundKey field.

MSK

If the inner round did not define an MSK, this field is set to all zeros.

If the inner method derives an MSK, then only the first 32 octets of that MSK are copied to this field.

EMSK

If the inner round did not define an EMSK, this field is set to all zeros.

If the inner method derives an EMSK, then only the first 32 octets of that EMSK are copied to this field.

3.3. Computing the Compound-MAC

The Compound-MAC used in the Crypto-Binding TLV is calculated via method as for TEAPv1. That definition is reproduced here verbatim:

Compound-MAC = the first 20 octets of MAC(CMK, BUFFER)

CMK is taken from the DerivedKey structure which was calculated for this round. The definition of BUFFER is the same as in [I-D.ietf-emu-rfc7170bis], Section 6.3 for TEAPv1.

For TEAPv2, only one CMK is derived for each inner message. This limitation also means that only one Compound-MAC is derived for the Crypto-Binding TLV ({#crypto-binding}).

3.4. EAP Master Session Key Generation

The final MSK and ESMK are generated via the following derivation:

```
MSK  = the first 64 octets of TLS-PRF(RoundSeed,
       "Session Key Generating Function")
EMSK = the first 64 octets of TLS-PRF(RoundSeed,
       "Extended Session Key Generating Function")
```

This construction ensures that the cryptographic binding requirements of [RFC6677] are satisfied, while using a much simpler key derivation than was done for TEAPv1.

3.5. TEAPv2 Message Format

The TEAPv2 message format is identical to that of TEAPv1 ([I-D.ietf-emu-rfc7170bis], Section 4.1) with only one change: the Ver field is set to "2", to indicate that this is TEAPv2.

3.6. TEAPv2 TLVs

The TEAPv2 TLV format and TLV definitions are identical to that for TEAPv1 ([I-D.ietf-emu-rfc7170bis], Section 4.2), with only the changes and additions noted below.

3.6.1. Crypto-Binding TLV

The format of the TEAPv2 Crypto-Binding TLV is the same as for TEAPv1 ([I-D.ietf-emu-rfc7170bis], Section 4.2.13), with the following changes:

* The Version field MUST set to two (2).

* The Received-Ver field MUST be set to two (2), to indicate TEAPv2.

* The Flags field MUST have value 2, to indicate that only the MSK Compound-MAC is present.

* the Nonce field is set to a random value. There is no need to set the least significant bit to zero or one. If the least significant bit is set to a particular value, it has no impact on the protocol.

* The ESMK Compound-MAC field is not used. It SHOULD be set to zeros by the sender. The receiver MUST ignore it.

* The MSK Compound-MAC field is calculated as described above in Section 3.3.

4. Security Considerations

5. IANA Considerations

6. References

6.1. Normative References

[BCP14] 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/rfc/rfc8174>.

[I-D.ietf-emu-rfc7170bis] DeKok, A., "Tunnel Extensible Authentication Protocol (TEAP) Version 1", Work in Progress, Internet-Draft, draft-ietf-emu-rfc7170bis-22, 28 May 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-emu-rfc7170bis-22>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[RFC9427] DeKok, A., "TLS-Based Extensible Authentication Protocol (EAP) Types for Use with TLS 1.3", RFC 9427, DOI 10.17487/RFC9427, June 2023, <https://www.rfc-editor.org/rfc/rfc9427>.

6.2. Informative References

[KAMATH] Palekar, R. H. and A., "Microsoft EAP CHAP Extensions", June 2007.

[RFC6677] Hartman, S., Ed., Clancy, T., and K. Hoeper, "Channel-Binding Support for Extensible Authentication Protocol (EAP) Methods", RFC 6677, DOI 10.17487/RFC6677, July 2012, <https://www.rfc-editor.org/rfc/rfc6677>.

[RFC7170] Zhou, H., Cam-Winget, N., Salowey, J., and S. Hanna, "Tunnel Extensible Authentication Protocol (TEAP) Version 1", RFC 7170, DOI 10.17487/RFC7170, May 2014, <https://www.rfc-editor.org/rfc/rfc7170>.

Author’s Address

Alan DeKok
InkBridge Networks
Email: alan.dekok@inkbridge.io