Network Working Group
Internet-Draft
Intended status: Experimental
Expires: February 7, 2021

E. Ingles
University of Murcia
D. Garcia-Carrillo
Odin Solutions S.L.
R. Marin-Lopez
University of Murcia

August 6, 2020

EAP method based on EDHOC Authentication
draft-ingles-eap-edhoc-00

Abstract

This document describes a proposal of an EAP method based on the EDHOC authentication protocol .

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 February 7, 2021.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

EDHOC [I-D.selander-lake-edhoc] is a new protocol for autentication and key derivation that has been proposed as an alternative in IoT to provide a secure exchange in an end-to-end fashion. This key material can be futher used to run other protocols such as OSCORE, as well as providing key material to any other protocol that needs pre-shared key material to secure the communications. Provides authentication and key material generation, which are basic pillars to the design of an EAP method. And indeed the most important thing is that it is lightweight and designed for IoT. In addition, the EDHOC implementation that exists on the device can be reused to establish OSCORE Security Associations (SAs) for the authentication process. EAP is a protocol that allows to implement different authentication mechanims, provides a framework for key management and has integration with AAA infrastructures. For these reasons, this new EAP method will allow the different applications and use cases to take advantage of EAP.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Protocol Overview

2.1. The EAP-EDHOC Conversation

The EAP-EDHOC conversation typically starts with the negotiation of EAP by the EAP authenticator and the EAP peer. The EAP authenticator typically sends an EAP-Request/Identity packet to the EAP peer, to

which the EAP peer answers with an EAP-Response/Identity. This last messages contains the peer's user-Id.

From this point on, the authenticator MAY act as a forwarder of the EAP messages between the EAP peer and the server, if the pass-through mode is used, receiving the EAP packets from the peer, encapsulating them for transmission to the EAP server that will act as Authentication Server (AS).

Once the EAP server receives the peer's Identity, it MUST respond with an an EAP-Request packet with EAP-Type=EAP-EDHOC and the prefered EDHOC authentication credential mode in decresing order, or no data if only one method is allowed or used. The EAP-EDHOC conversation will then begin, with the peer sending an EAP-Response packet with EAP-Type=EAP-EDHOC. The data field of that packet will encapsulate the "EDHOC Message 1".

The EAP server will then respond with an EAP-Request packet with EAP-Type=EAP-EDHOC. The data field of this packet will encapsulate "EDHOC Message 2" message. To this message, the EAP peer will send the and EAP-Response message containing the "EDHOC Message 3" message.

In the case where the EDHOC mutual authentication is successful, the conversation will appear as follows:

```
+--------+                                                  +-------+
|  EAP   |                                                  |  EAP  |
|  Peer  |                                                  | AuthN |
+--------+                                                  +-------+
  |                          EAP-Request/Identity             |
  | <---------------------------------------------------+ |
  |                                                         |
  |   EAP-Response/Identity (ID)                            |
  | +---------------------------------------------------> |
  |                                          EAP-Request/    |
  |                                    EAP-Type=EAP-EDHOC    |
  |                                             (RPK,PSK)    |
  | <---------------------------------------------------+ |
  |   EAP-Response/                                         |
  |   EAP-Type=EAP-EDHOC                                    |
  |   (EDHOC message 1)                                     |
  | +---------------------------------------------------> |
  |                                          EAP-Request/    |
  |                                    EAP-Type=EAP-EDHOC    |
  |                                     (EDHOC message 2)    |
  | <---------------------------------------------------+ |
  |   EAP-Response/                                         |
  |   EAP-Type=EAP-EDHOC                                    |
  |   (EDHOC message 3)                                     |
  | +---------------------------------------------------> |
  |                                                         |
  |                                          EAP-Success     |
  | <---------------------------------------------------+ |
  +                                                         +
```

Figure 1: Overview EDHOC exchange

2.1.1. Identity

It is RECOMMENDED to use NAIs in the Identity Response as identities. While

3. Identity Verification

The identity provided in the EAP-Response/Identity is not authenticated by EAP-EDHOC, hence SHALL NOT be used for authorization or accounting purposes. The authenticator and the EAP server MAY examine the identity presented in EAP-Response/Identity for routing and EAP method selection.

4. Key Hierarchy

EDHOC uses HKDF RFC 5869 [RFC5869] to derive keys. HKDF-Extract is used for deriving fixed-length uniformly pseudorandom keys (PRK) from ECDH shared secrets. HKDF-Expand is used for deriving additional output keying material (OKM) from the PRKs.

The derivation proceeds as follows:

PRK = HKDF-Extract(salt, IKM)

Where:

HKDF-Extract = RFC5869 HKDF function

salt = PSK when using symmetric keys, and the empty byte string when using asymmetric keys

IKM (input keying material) = the ECDH shared secret

Figure 2 illustrates the EDHOC Key Hierarchy.

In EAP-EDHOC, the MSK, EMSK, and Initialization Vector (IV) are derived from the PRK via a hash function. This ensures that the EDHOC PRK cannot be derived from the MSK, EMSK, or IV unless the hash function is defeated. Since the MSK and EMSK are derived from the EDHOC PRK, if the EDHOC PRK is compromised then the MSK and EMSK are also compromised.

EAP-EDHOC derives exported keying material and parameters as follows:

Type-Code = 0XFF

Key_Material = HKDF-Expand(EDHOC PRK, "EAP-EDHOC encryption", 128)

MSK = Key_Material(0,63)

EMSK = Key_Material(64,127)

IV = HKDF-Expand(EDHOC PRK, "EAP-EDHOC IV", 64)

Session-Id = Type-Code || Method-Id

Method-Id = HKDF-Expand(EDHOC PRK, "EAP_EDHOC_Method-Id", 64)

Where:

Key_Material(S,F) = Octets S through F inclusive of the key material.

```
                                  | PRK
                                  V
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
        |                        MSK, EMSK                          |
        |                label == "EAP-EDHOC encryption"            |
        +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
          |                            |                       |
          | MSK(0,63)                  | EMSK(0,63)            | IV (0,63)
          |                            |                       |
          |                            |                       |
          V                            V                       V
```

Figure 2: EAP-EDHOC Key derivation

5. IANA considerations

TBD.

6. Security Considerations

TBD.

7. Acknowledgements

This work is possible due the EU Project IoTCrawlwer under grant agreement n.779852, the EU Project INSPIRE-5Gplus under grant agreement n.871808 and to the pre-doctoral grant Industrial PhD DI-16-08432 granted to ODIN Solutions S.L

8. Normative References

[I-D.selander-lake-edhoc]
Selander, G., Mattsson, J., and F. Palombini, "Ephemeral Diffie-Hellman Over COSE (EDHOC)", draft-selander-lake-edhoc-01 (work in progress), March 2020.

[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>.

[RFC2548] Zorn, G., "Microsoft Vendor-specific RADIUS Attributes", RFC 2548, DOI 10.17487/RFC2548, March 1999, <https://www.rfc-editor.org/info/rfc2548>.

[RFC5869] Krawczyk, H. and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)", RFC 5869, DOI 10.17487/RFC5869, May 2010, <https://www.rfc-editor.org/info/rfc5869>.

Authors' Addresses

Eduardo Ingles-Sanchez
University of Murcia
Campus de Espinardo S/N, Faculty of Computer Science
Murcia 30100
Spain

Email: eduardo.ingles@um.es

Dan Garcia-Carrillo
Odin Solutions S.L.
Poligono Industrial Oeste, C/ Peru, 5
Alcantarilla, Murcia 30820
Spain

Email: dgarcia@odins.es

Rafael Marin-Lopez
University of Murcia
Campus de Espinardo S/N, Faculty of Computer Science
Murcia 30100
Spain

Phone: +34 868 88 85 01
Email: rafa@um.es