Workgroup: Network Working Group
Internet-Draft:
draft-ietf-ipsecme-ikev2-auth-announce-03
Published: 14 April 2023
Intended Status: Standards Track
Expires: 16 October 2023
Authors: V. Smyslov
ELVIS-PLUS

Announcing Supported Authentication Methods in IKEv2

Abstract

This specification defines a mechanism that allows the Internet Key Exchange version 2 (IKEv2) implementations to indicate the list of supported authentication methods to their peers while establishing IKEv2 Security Association (SA). This mechanism improves interoperability when IKEv2 partners are configured with multiple different credentials to authenticate each other.

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 16 October 2023.

Table of Contents

1. Introduction

The Internet Key Exchange version 2 (IKEv2) protocol, defined in [RFC7296], performs authenticated key exchange in IPsec. IKEv2, unlike its predecessor IKEv1, defined in [RFC2409], doesn't include a mechanism to negotiate an authentication method that the peers would use to authenticate each other. It is assumed that each peer selects whatever authentication method it thinks is appropriate, depending on authentication credentials it has.

This approach generally works well when there is no ambiguity in selecting authentication credentials. The problem may arise when there are several credentials of different type configured on one peer, while only some of them are supported on the other peer. Another problem situation is when a single credential may be used to produce different types of authentication tokens (e.g. signatures of different formats). Emerging post-quantum signature algorithms may bring additional challenges for implementations, especially if so called hybrid schemes are used (e.g. see [I-D.ounsworth-pq-composite-sigs]).

This specification defines an extension to the IKEv2 protocol that allows peers to announce their supported authentication methods, thus decreasing risks of SA establishment failure in situations when there are several ways for the peers to authenticate themselves.

2. Terminology and Notation

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.

3. Protocol Details

When establishing IKE SA each party may send a list of authentication methods it supports and is configured to use to its peer. The sending party may additionally specify that some of the authentication methods are only for use with the particular trust anchors. Upon receiving this information the peer may take it into consideration while selecting an algorithm for its authentication if several alternatives are available.

3.1. Exchanges

If the responder is willing to use this extension, it includes a new status type notify SUPPORTED_AUTH_METHODS in the IKE_SA_INIT response message. This notification contains a list of authentication methods supported by the responder.

```
Initiator                              Responder
-----------                            -----------
HDR, SAi1, KEi, Ni -->
                                  <-- HDR, SAr1, KEr, Nr, [CERTREQ,]
                                    [N(SUPPORTED_AUTH_METHODS)(...)]
```

Figure 1: The IKE_SA_INIT Exchange

If the initiator doesn't support this extension, it ignores the received notification as an unknown status notify. Otherwise, it may send the SUPPORTED_AUTH_METHODS notification in the IKE_AUTH request message, with a list of authentication methods supported by the initiator.

```
Initiator                              Responder
-----------                            -----------
HDR, SK {IDi, [CERT,] [CERTREQ,]
[IDr,] AUTH, SAi2, TSi, TSr,
[N(SUPPORTED_AUTH_METHODS)(...)] }  -->
                                  <-- HDR, SK {IDr, [CERT,]
                                         AUTH, SAr2, TSi, TSr }
```

Figure 2: The IKE_AUTH Exchange

If the initiator is configured to use Extensible Authentication Protocol (EAP) for authentication in IKEv2 (see Section 2.16 of [RFC7296]), then it SHOULD NOT send the SUPPORTED_AUTH_METHODS notification.

Since the responder sends the SUPPORTED_AUTH_METHODS notification in the IKE_SA_INIT exchange, it must take care that the size of the response message wouldn't grow too much so that IP fragmentation takes place. If the following conditions are met:

* the SUPPORTED_AUTH_METHODS notification to be included is so large, that the responder suspects that IP fragmentation of the resulting IKE_SA_INIT response message may happen;

* both peers support the IKE_INTERMEDIATE exchange, defined in [RFC9242] (i.e. the responder has received and is going to send the INTERMEDIATE_EXCHANGE_SUPPORTED notification);

then the responder may choose not to send actual list of the supported authentication methods in the IKE_SA_INIT exchange and instead ask the initiator to start the IKE_INTERMEDIATE exchange for the list to be sent in. In this case the responder includes SUPPORTED_AUTH_METHODS notification containing no data in the IKE_SA_INIT response.

If the initiator receives the empty SUPPORTED_AUTH_METHODS notification in the IKE_SA_INIT exchange, it means that the responder is going to send the list of the supported authentication methods in the IKE_INTERMEDIATE exchange. If this exchange is to be initiated anyway for some other reason, then the responder MAY use it to send the SUPPORTED_AUTH_METHODS notification. Otherwise, the initiator MAY start the IKE_INTERMEDIATE exchange just for this sole purpose by sending an empty IKE_INTERMEDIATE request. The initiator MAY also indicate its identity (and possibly the perceived responder's identity too) by including the IDi payload (possibly along with the IDr payload) into the IKE_INTERMEDIATE request. This information could help the responder to send back only those authentication methods, that are configured to be used for authentication of this particular initiator. If these payloads are sent, they MUST be identical to the IDi/IDr payloads sent later in the IKE_AUTH request.

If the responder has sent any CERTREQ payload in the IKE_SA_INIT, then it MUST re-send the same payload(s) in the IKE_INTERMEDIATE response containing the SUPPORTED_AUTH_METHODS notification if any of the included Announcements has a non-zero Cert Link field (see Section 3.2.2 and Section 3.2.3). This requirement allows peers to have a list of Announcements and a list of CAs in the same message, which simplifies their linking (note, that this requirement is

always fulfilled for the IKE_SA_INIT and IKE_AUTH exchanges). However, if for any reason the responder doesn't re-send CERTREQ payload(s) in the IKE_INTERMEDIATE exchange, then the initiator MUST NOT abort negotiation. Instead, the initiator MAY either link the Announcements to the CAs received in the IKE_SA_INIT response, or MAY ignore the SUPPORTED_AUTH_METHODS notification entirely.

If multiple IKE_INTERMEDIATE exchanges take place during IKE SA establishments, it is RECOMMENDED that the responder use the last IKE_INTERMEDIATE exchange (the one just before IKE_AUTH) to send the list of supported auth methods. However, it is not always possible for the responder to know how many IKE_INTERMEDIATE exchanges the initiator will use. In this case the responder MAY send the list in any IKE_INTERMEDIATE exchange. If the initiator sends IDi/IDr in an IKE_INTERMEDIATE request, then it is RECOMMENDED that the responder sends back the list of authentication methods in the response.

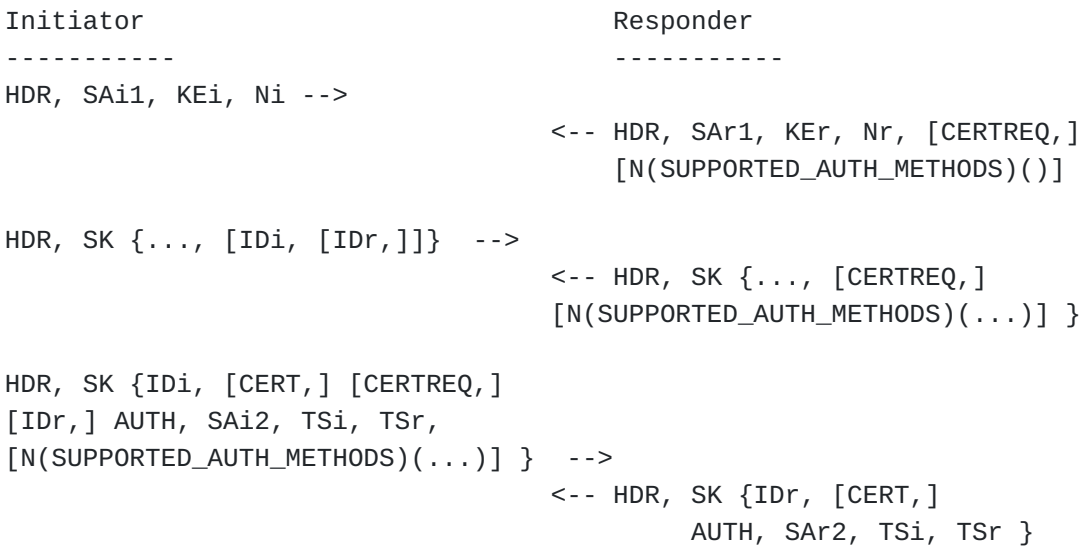

Figure 3: Using the IKE_INTERMEDIATE Exchange for sending auth methods

Note, that sending the SUPPORTED_AUTH_METHODS notification and using information obtained from it is optional for both the initiator and the responder.

3.2. SUPPORTED_AUTH_METHODS Notify

The format of the SUPPORTED_AUTH_METHODS notification is shown below.

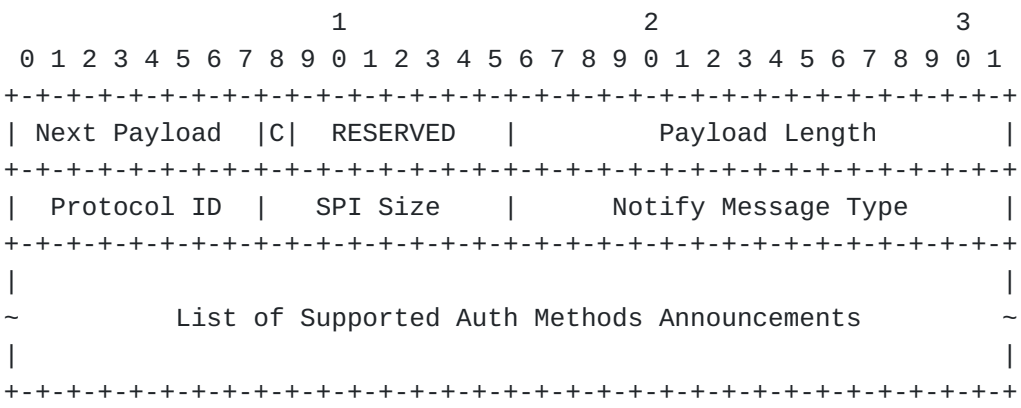

Figure 4: SUPPORTED_AUTH_METHODS Notify

The Notify payload format is defined in Section 3.10 of [RFC7296]. When a Notify payload of type SUPPORTED_AUTH_METHODS is sent, the Protocol ID field is set to 0, the SPI Size is set to 0, meaning there is no SPI field, and the Notify Message Type is set to <TBA by IANA>.

The Notification Data field contains the list of supported authentication methods announcements. Each individual announcement is a variable-size data blob, which format depends on the announced authentication method. The blob always starts with an octet containing the length of the blob followed by an octet containing the authentication method. Authentication methods are represented as values from the "IKEv2 Authentication Method" registry defined in [IKEV2-IANA]. The meaning of the remaining octets of the blob, if any, depends on the authentication method and is defined below. Note, that for the currently defined authentication methods the length octet fully defines both the format and the semantics of the blob.

If more authentication methods are defined in future, the corresponding documents must describe the semantics of the announcements for these methods. Implementations MUST ignore announcements which semantics they don't understand.

3.2.1. 2-octet Announcement

If the announcement contains an authentication method that is not concerned with public key cryptography, then the following format is used.

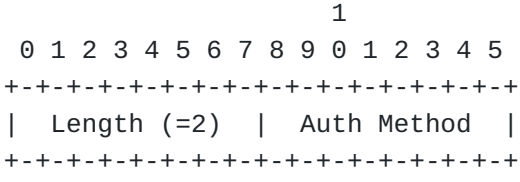

Figure 5: Supported Authentication Method

*Length - Length of the blob, must be 2 for this case.

*Auth Method - Announced authentication method.

This format is applicable for the authentication methods "Shared Key Message Integrity Code" (2) and "NULL Authentication" (13). Note, that authentication method "Generic Secure Password Authentication Method" (12) would also fall in this category, however it is negotiated separately (see [RFC6467] and for this reason there is no point to announce it via this mechanism. See also Section 4.

3.2.2. 3-octet Announcement

If the announcement contains an authentication method that is concerned with public key cryptography, then the following format is used. This format allows to link the announcement with a particular trust anchor from the Certificate Request payload.

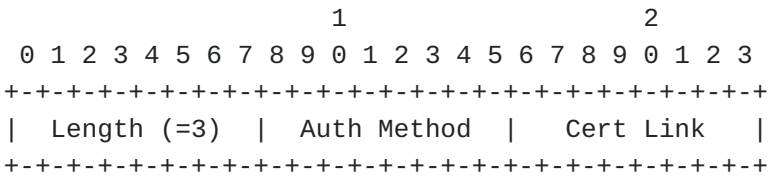

Figure 6: Supported Authentication Method

*Length - Length of the blob, must be 3 for this case.

*Auth Method - Announced authentication method.

*Cert Link - Links this announcement with particular CA.

If the Cert Link field contains non-zero value N, it means that the announced authentication method is intended to be used only with the N-th trust anchor (CA certificate) from the Certificate Request payload(s) sent by this peer. If it is zero, then this authentication method may be used with any CA. If multiple CERTREQ payloads were sent, the CAs from all of them are treated as a single list for the purpose of the linking. If no Certificate Request payload were receives, the content of this field MUST be ignored and treated as zero.

This format is applicable for the authentication methods "RSA Digital Signature" (1), "DSS Digital Signature" (3), "ECDSA with SHA-256 on the P-256 curve" (9), "ECDSA with SHA-384 on the P-384 curve" (10) and "ECDSA with SHA-512 on the P-512 curve" (11). Note however, that these authentication methods are currently superseded

by the "Digital Signature" (14) authentication method, which has a different announcement format, described below.

3.2.3. Multi-octet Announcement

The following format is currently used only with the "Digital Signature" (14) authentication method.

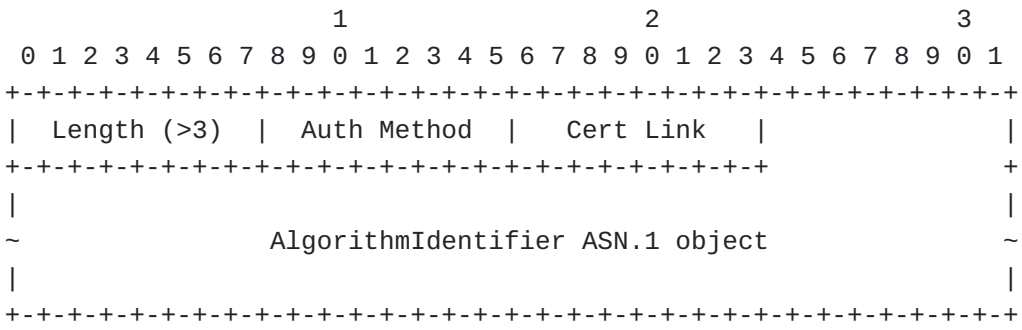

Figure 7: Supported Authentication Method

*Length - Length of the blob, must be greater than 3 for this case.

*Auth Method - Announced authentication method, at the time of writing this document only value 14 ("Digital Signature") is allowed.

*Cert Link - Links this announcement with particular CA; see Section 3.2.2 for details.

*AlgorithmIdentifier ASN.1 object - DER-encoded AlgorithmIdentifier ASN.1 object.

The "Digital Signature" authentication method, defined in [RFC7427], supersedes previously defined signature authentication methods. In this case the real authentication algorithm is identified via AlgorithmIdentifier ASN.1 object. Appendix A in [RFC7427] contains examples of Commonly Used ASN.1 Objects.

4. Interaction with IKE Extensions concerning Authentication

Generally in IKEv2 each party independently determines the way it authenticates itself to the peer. In other words, authentication methods selected by the peers need not be the same. However, some IKEv2 extensions break this rule.

The prominent example is [RFC6467], (Secure Password Framework for Internet Key Exchange Version 2), which defines a framework for using Password-authenticated key exchanges (PAKE) in IKEv2. With

this framework peers negotiate using one of PAKE methods in the IKE_SA_INIT exchange - the initiator sends a list of supported PAKE methods in the request and the responder picks one of them and sends it back in the response.

If peers negotiate PAKE for authentication, then the selected PAKE method is used by both initiator and responder and no other authentication methods are involved. For this reason there is no point to announce supported authentication methods in this case. Thus, if the peers choose to go with PAKE, they MUST NOT send the SUPPORTED_AUTH_METHODS notification.

5. Security Considerations

Security considerations for IKEv2 protocol are discussed in [RFC7296]. It is believed that this extension doesn't add new vulnerabilities to the protocol.

6. IANA Considerations

This document defines a new Notify Message Types in the "Notify Message Types - Status Types" registry:

<TBA>	SUPPORTED_AUTH_METHODS

7. Acknowledgments

The author would like to thank Paul Wouters for his valuable comments and proposals for protocol improvement.

8. References

8.1. 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>.

[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>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7427] Kivinen, T. and J. Snyder, "Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)", RFC 7427,

DOI 10.17487/RFC7427, January 2015, <https://www.rfc-editor.org/info/rfc7427>.

[RFC9242] Smyslov, V., "Intermediate Exchange in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 9242, DOI 10.17487/RFC9242, May 2022, <https://www.rfc-editor.org/info/rfc9242>.

[IKEV2-IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", <http://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml#ikev2-parameters-7>.

8.2. Informative References

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, DOI 10.17487/RFC2409, November 1998, <https://www.rfc-editor.org/info/rfc2409>.

[RFC6467] Kivinen, T., "Secure Password Framework for Internet Key Exchange Version 2 (IKEv2)", RFC 6467, DOI 10.17487/RFC6467, December 2011, <https://www.rfc-editor.org/info/rfc6467>.

[I-D.ounsworth-pq-composite-sigs] Ounsworth, M., Gray, J., and M. Pala, "Composite Signatures For Use In Internet PKI", Work in Progress, Internet-Draft, draft-ounsworth-pq-composite-sigs-08, 13 March 2023, <https://datatracker.ietf.org/doc/html/draft-ounsworth-pq-composite-sigs-08>.

Author's Address

Valery Smyslov
ELVIS-PLUS
PO Box 81
Moscow (Zelenograd)
124460
Russian Federation

Phone: +7 495 276 0211
Email: svan@elvis.ru