

Workgroup: Network Working Group  
Internet-Draft:  
draft-ietf-ipsecme-ikev2-auth-announce-10  
Published: 18 April 2024  
Intended Status: Standards Track  
Expires: 20 October 2024  
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 credentials of different type 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 20 October 2024.

### **Copyright Notice**

Copyright (c) 2024 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 Revised BSD License text as described in

Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

- [1. Introduction](#)
- [2. Terminology and Notation](#)
- [3. Protocol Details](#)
  - [3.1. Exchanges](#)
  - [3.2. SUPPORTED\\_AUTH\\_METHODS Notify](#)
    - [3.2.1. 2-octet Announcement](#)
    - [3.2.2. 3-octet Announcement](#)
    - [3.2.3. Multi-octet Announcement](#)
- [4. Interaction with IKEv2 Extensions concerning Authentication](#)
- [5. Security Considerations](#)
- [6. IANA Considerations](#)
- [7. Acknowledgments](#)
- [8. References](#)
  - [8.1. Normative References](#)
  - [8.2. Informative References](#)
- [Appendix A. Examples of Announcing Supported Authentication Methods](#)
  - [A.1. No Need to Use the IKE\\_INTERMEDIATE Exchange](#)
  - [A.2. With Use of the IKE\\_INTERMEDIATE Exchange](#)
- [Author's Address](#)

## 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. SA establishment failure between peers may arise when there are several credentials of different types 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). Since IKEv2 requires that each peer uses exactly one authentication method and doesn't provide means for peers to indicate to the other side which authentication methods they support, it is possible that in these situations the peer that supports wider range of authentication methods (or authentication token formats) improperly selects the method (or format) which is not supported by the other side.

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. For this purpose this specification introduces a new Notify Message Type SUPPORTED\_AUTH\_METHODS. The Notify payload with this Notify Message Type is utilized to convey the supported authentication methods of the party sending it. The sending party may additionally specify that some of the authentication methods are only for use with the particular trust anchors. The receiving party may take this information into consideration when selecting an algorithm for its authentication (i.e., the algorithm used for calculation of the AUTH payload) if several alternatives are available. To simplify the receiver's task of linking the announced authentication methods with the trust anchors, the protocol ensures that the SUPPORTED\_AUTH\_METHODS notification is always co-located with the CERTREQ payload in the same message.

### 3.1. Exchanges

The initiator starts the IKE\_SA\_INIT exchange as usual. If the responder is willing to use this extension, it includes a new notification SUPPORTED\_AUTH\_METHODS in the IKE\_SA\_INIT response message. This notification contains a list of authentication methods supported by the responder, ordered by their preference.

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.

Regardless of whether the notification is received, if the initiator supports and is willing to use this extension, it includes the SUPPORTED\_AUTH\_METHODS notification in the IKE\_AUTH request message, with a list of authentication methods supported by the initiator, ordered by their preference.

```
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

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 both of 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. This would allow using IKE fragmentation [\[RFC7383\]](#) for long messages (which cannot be used in the IKE\_SA\_INIT exchange), thus avoiding IP fragmentation. 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 SHOULD 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 Announcements containing links to CAs.

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.

```

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

HDR, SK {..., [IDi, [IDr,]]} -->
                                     <-- HDR, SK {..., [CERTREQ,]
                                     [N(SUPPORTED_AUTH_METHODS)(...)] }

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

```

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. If multiple SUPPORTED\_AUTH\_METHODS notifications are included in a message, all their announcements form a single ordered list, unless overridden by other extension (see [Section 4](#)).

### 3.2. SUPPORTED\_AUTH\_METHODS Notify

The format of the SUPPORTED\_AUTH\_METHODS notification is shown below.

```

          1                2                3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Next Payload |C|  RESERVED   |           Payload Length          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Protocol ID  |  SPI Size    |     Notify Message Type         |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|
~          List of Supported Auth Methods Announcements          ~
|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

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>.

Notification data 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. 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 the future, the corresponding documents must describe the semantics of the announcements for these methods. Implementations MUST ignore announcements whose 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.

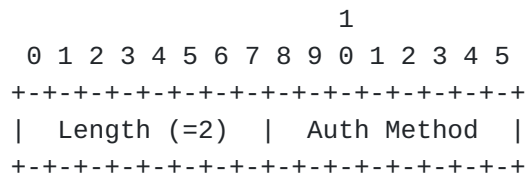


Figure 5: Supported Authentication Method

\*Length - Length of the blob in octets, 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 linking the announcement with a particular trust anchor from the Certificate Request payload.





Figure 7: Supported Authentication Method

\*Length - Length of the blob in octets, 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 - the AlgorithmIdentifier of PKIX (see Section 4.1.1.2 of [\[RFC5280\]](#)), encoded using distinguished encoding rules (DER) [\[X.690\]](#).

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 IKEv2 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.

If peers are going to use Multiple Authentication Exchanges [\[RFC4739\]](#), then they MAY include multiple SUPPORTED\_AUTH\_METHODS notifications instead of one, each containing authentication methods appropriate for each authentication round. The notifications are included in the order of the preference of performing authentication rounds.

## 5. Security Considerations

Security considerations for IKEv2 protocol are discussed in [RFC7296]. Security properties of different authentication methods varies. Refer to corresponding documents, listed in [IKEV2-IANA] for discussion of security properties of each authentication method.

Announcing authentication methods gives an eavesdropper additional information about peers' capabilities. If a peer advertises NULL authentication along with other methods, then active attacker on the path can encourage peers to use NULL authentication by removing all other announcements. Note, that this is not a real "downgrade" attack, since authentication methods in IKEv2 are not negotiated and in this case NULL authentication should be allowed by local security policy.

Similarly, if an attacker on the path can break some of the announced authentication methods online, then the attacker can encourage peers to use one of these weaker methods by removing all other announcements, and if this succeeds, then perform person-in-the-middle attack.

## 6. IANA Considerations

This document defines a new Notify Message Type in the "IKEv2 Notify Message Status Types" registry referencing this RFC:

<TBA>           SUPPORTED\_AUTH\_METHODS     [RFCXXXX]

## 7. Acknowledgments

The author would like to thank Paul Wouters for his valuable comments and proposals. The author is also grateful to Daniel Van Geest, who made proposals for protocol improvement. Reese Enghardt and Rifaat Shekh-Yusef contributed to the clarity of the document.

## 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>>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <<https://www.rfc-editor.org/info/rfc5280>>.

[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>>.

[X.690] "ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", July 2002.

[IKEV2-IANA] IANA, "Internet Key Exchange Version 2 (IKEv2) Parameters", <<https://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xhtml#ikev2-parameters-12>>.

## 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>>.

[RFC4739] Eronen, P. and J. Korhonen, "Multiple Authentication Exchanges in the Internet Key Exchange (IKEv2) Protocol",

RFC 4739, DOI 10.17487/RFC4739, November 2006, <<https://www.rfc-editor.org/info/rfc4739>>.

[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>>.

[RFC7383] Smyslov, V., "Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation", RFC 7383, DOI 10.17487/RFC7383, November 2014, <<https://www.rfc-editor.org/info/rfc7383>>.

[I-D.ounsworth-pq-composite-sigs] Ounsworth, M., Gray, J., Pala, M., and J. Klaußner, "Composite ML-DSA for use in Internet PKI", Work in Progress, Internet-Draft, draft-ounsworth-pq-composite-sigs-13, 4 March 2024, <<https://datatracker.ietf.org/doc/html/draft-ounsworth-pq-composite-sigs-13>>.

## Appendix A. Examples of Announcing Supported Authentication Methods

This appendix shows some examples of announcing authentication methods. This appendix is purely informative; if it disagrees with the body of this document, the other text is considered correct. Note that some payloads that are not relevant to this specification may be omitted for brevity.

### A.1. No Need to Use the IKE\_INTERMEDIATE Exchange

This example illustrates the situation when the SUPPORTED\_AUTH\_METHODS notify fits into the IKE\_SA\_INIT message and thus the IKE\_INTERMEDIATE exchange is not needed. In this scenario the responder announces that it supports the "Shared Key Message Integrity Code" and the "NULL Authentication" authentication methods. The initiator informs the responder that it supports only the "Shared Key Message Integrity Code" authentication method.

Initiator	Responder
-----	-----
	IKE_SA_INIT
HDR, SAI1, KEi, Ni -->	<-- HDR, SAR1, KEr, Nr, N(SUPPORTED_AUTH_METHODS( PSK, NULL))
	IKE_AUTH
HDR, SK {IDi, AUTH, SAI2, TSi, TSr, N(SUPPORTED_AUTH_METHODS(PSK))} -->	<-- HDR, SK {IDr, AUTH, SAR2, TSi, TSr}

### A.2. With Use of the IKE\_INTERMEDIATE Exchange

This example illustrates the situation when the IKE\_INTERMEDIATE exchange is used. In this scenario the responder announces that it supports the "Digital signature" authentication method using the RSASSA-PSS algorithm with CA1 and CA2 and the same method using the ECDSA algorithm with CA3. The initiator supports only the "Digital signature" authentication method using the RSASSA-PSS algorithm with no link to a particular CA.

Initiator

-----

IKE\_SA\_INIT

HDR, SAi1, KEi, Ni,  
N(SIGNATURE\_HASH\_ALGORITHMS) -->

Responder

-----

<-- HDR, SAR1, KEr, Nr,  
CERTREQ(CA1, CA2, CA3),  
N(SIGNATURE\_HASH\_ALGORITHMS),  
N(SUPPORTED\_AUTH\_METHODS())

IKE\_INTERMEDIATE

HDR, SK {..., IDi]} -->

<-- HDR, SK {...,  
CERTREQ(CA1, CA2, CA3),  
N(SUPPORTED\_AUTH\_METHODS(  
SIGNATURE(RSASSA-PSS:1),  
SIGNATURE(RSASSA-PSS:2),  
SIGNATURE(ECDSA:3)))}

IKE\_AUTH

HDR, SK {IDi, CERT, CERTREQ(CA2),  
AUTH, SAi2, TSi, TSr,  
N(SUPPORTED\_AUTH\_METHODS(  
SIGNATURE(RSASSA-PSS:0)))} -->

<-- HDR, SK {IDr, CERT,  
AUTH, SAR2, TSi, TSr}

#### Author's Address

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

Phone: [+7 495 276 0211](tel:+74952760211)

Email: [svan@elvis.ru](mailto:svan@elvis.ru)