IKEv2 Optional SA&TS Payloads in Child Exchange
draft-kampati-ipsecme-ikev2-sa-ts-payloads-opt-02

Abstract

This document describes a method for reducing the size of the Internet Key Exchange version 2 (IKEv2) exchanges at time of rekeying IKE SAs and Child SAs by removing or making optional of SA & TS payloads. Reducing size of IKEv2 exchanges is desirable for low power consumption battery powered devices. It also helps to avoid IP fragmentation of IKEv2 messages.

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 May 7, 2020.

Copyright Notice

Copyright (c) 2019 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 Internet Key Exchange protocol version 2 (IKEv2) specified in [RFC7296] is used in the IP Security (IPsec) architecture for the purposes of Security Association (SA) parameters negotiation and authenticated key exchange. The protocol uses UDP as the transport for its messages, which size varies from less than one hundred bytes to several kilobytes.

According to [RFC7296], the secret keys used by IKE/IPSec SAs should be used only for a limited amount of time and to protect a limited amount of data. When the lifetime of an SA expires or is about to expire, the peers can rekey the SA to reestablish a new SA to take the place of the old one.
For security gateways/ePDG in 4G networks and cRAN/Cloud in 5G networks, they will support more than 100,000 IKE/IPSEC tunnels. So on an average, for every second there may be hundreds or thousands of IKE SAs and Child SAs that are rekeying. This takes huge amount of bandwidth, packet fragmentation and more processing resources. For these devices, these problems can be solved by introducing the solution described in this document.

This is also useful in Internet of Things (IoT) devices which utilizing lower power consumption technology. The appendix A of [I-D.mglt-6lo-diet-esp-requirements] gives some estimate data. For these devices, reducing the length of IKE/Child SA rekeying messages can save the bandwidth consumption. At the same time, it can also save the computing processes by less payload are included.

Most devices don’t prefer to change cryptographic suites frequently. By taking this advantage the SA and TS payloads can be made optional at the time of rekeying IKE SAs and Child SAs. In such situation, only a new SPI value is needed to create the new IKE SA and Child SA. So a new Notify payload which contains the needed SPI value can be sent instead of the SA and TS payloads.

In case of rekeying IKE SAs, the SA payloads can be optimized if there is no change of cryptographic suites. In case of rekeying Child SAs, the SA and TS payloads can be optimized if there is no change of cryptographic suites and ACL configuration.

2. Conventions Used in This Document

2.1. 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.

3. Protocol Details

This section provides protocol details and contains the normative parts.

Before using this new optimization, the IPSec implementation who supports it has to know that the peer also supports it. Without the support on both sides, the optimized rekeying messages sent by one peer may be unrecognizable for the other peer. To stop this failure from happening, the first step is to negotiate the support of this optimization between the two peers. There are two specific rekeying
SAs cases: rekeying IKE SAs and rekeying Child SAs. After the negotiation, it’s up to the initiator to decide at which case to optimize the rekeying messages. The initiator can optimize the rekeying message payloads in both cases, or just in one case. The responder can react based on the received rekeying message.

3.1. Negotiation of Support for Optimizing Optional Payload at Rekeying IKE SAs and Child SAs

The initiator indicates its support for optimizing optional payloads at rekeying IKE SAs and Child SAs by including a Notify payload of type MINIMAL_REKEY_SUPPORTED in the IKE_AUTH request message. By observing the MINIMAL_REKEY_SUPPORTED notification in the received message, the responder can deduce the initiator’s support of this extension. If the responder also supports this extension, it includes the MINIMAL_REKEY_SUPPORTED notification in the corresponding response message. After receiving the response message, the initiator can also know the support of this extension of the responder side.

The IKE_AUTH message exchange in this case is shown below:

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

If the responder doesn’t support this extension, it MUST ignore the MINIMAL_REKEY_SUPPORTED notification sent by the initiator and MUST NOT respond error to the initiator. With no MINIMAL_REKEY_SUPPORTED notification in the response message, the initiator can deduce that the responder doesn’t support this extension. In this case, the IKE SAs and Child SAs rekeyings happen as the usual way without the optimizations defined in this document.

The IKE_AUTH message exchange in this case is shown below:

Initiator                               Responder
---------------------------------------------------------------
HDR, SK {IDi, [CERT,] [CERTREQ,]
[IDr,] AUTH, SAi2, TSi, TSr,
N(MINIMAL_REKEY_SUPPORTED)} -->
<-- HDR, SK {IDr, [CERT,] AUTH,
SAr2, TSi, TSr}
3.2. Payload Optimization at Rekeying IKE SAs

The payload optimization at rekeying IKE SAs MUST NOT be used unless both peers have indicated their support of this extension by using the negotiation method described in Section 3.1. If the initiator decides to optimize the payloads at the time of rekeying IKE SAs, then it includes the SA_UNCHANGED notification in its CREATE_CHILD_SA exchange message. If the initiator decides not to do the optimization, then it just sends the rekeying request message as the original way, the rekeying is conducted as [RFC7296] defined.

3.2.1. Rekeying IKE SAs When No Change of Initiator and Responder’s Cryptographic Suites

At the time of rekeying an IKE SA, when the initiator determines there is no change on its cryptographic suites since this IKE SA was created or last rekeyed, it MAY send the SA_UNCHANGED notification payload instead of the SA payloads in the rekeying request message. In this SA_UNCHANGED notification, it contains the initiator’s new Security Parameter Index (SPI) used for creating the new IKE SA.

After receiving the initiator’s rekeying request message with the SA_UNCHANGED notification and no SA payloads, the responder knows that the initiator wants to optimize the rekeying payload. Then when it determines that there is also no change in its cryptographic suites, the responder MAY send the rekeying respond message to the initiator with the SA_UNCHANGED notification payload instead of the SA payloads. In this SA_UNCHANGED notification, it contains the responder’s new SPI used for creating the new IKE SA.

According to the initiator’s new SPI and the responder’s new SPI, the initiator and the responder can rekey the IKE SA on both sides.

The CREATE_CHILD_SA message exchange in this case is shown below:

Initiator                         Responder
--------------------------------------------------------------------
HDR, SK {N(SA_UNCHANGED), Ni, KEi} -->
                            <-- HDR, SK {N(SA_UNCHANGED), Nr, KEr}

The initiator sends a SA_UNCHANGED notification payload, a Nonce payload and a Diffie-Hellman value in the KEi payload. A new initiator SPI is supplied in the SPI field of the SA_UNCHANGED notification payload.

The responder replies (using the same Message ID to respond) with a SA_UNCHANGED notification payload, a Nonce payload and a Diffie-
3.2.2. Rekeying IKE SAs When Responder’s Cryptographic Suites Changed

At the time of or before rekeying IKE SAs, the responder’s cryptographic suites may be changed while there is no change of initiator’s cryptographic suites. New cryptographic suites may be added to the responder, or some outdated cryptographic suites may be deleted from the responder. In this situation, the initiator MAY send the SA_UNCHANGED notification payload instead of the SA payloads in the CREATE_CHILD_SA request message at the time of rekeying IKE SAs.

If the responder decides to continue using the previously negotiated cryptographic suite to rekey the IKE SA, it MAY send the SA_UNCHANGED notification payload in the CREATE_CHILD_SA response message, then the rekeying is conducted like the way described in Section 3.2.1.

If the responder decides to re-negotiate the cryptographic suite, it MUST send NO_PROPOSAL_CHOSEN notification payload in the CREATE_CHILD_SA response message. After receiving this error notification, the initiator MUST retry the CREATE_CHILD_SA exchange with the SA payloads. Then the rekeying is conducted in the original way defined in [RFC7296]. The CREATE_CHILD_SA message exchange in this case is shown below:

Initiator                         Responder
--------------------------------------------------------------------
HDR, SK {N(SA_UNCHANGED), Ni, KEi} -->  HDR, SK {N(NO_PROPOSAL_CHOSEN),  
                                           Nr, KEr}  
                                           <--- HDR, SK {SA, Ni, KEi}  
                                           <--- HDR, SK {SA, Ni, KEi}

Besides, if the responder only supports the Child SA rekeying optimization and doesn’t support the IKE SA rekeying optimization, it can also follow the way described above, i.e., it MUST send NO_PROPOSAL_CHOSEN notification payload in the CREATE_CHILD_SA response message when receiving the SA_UNCHANGED notification at the time of rekeying IKE SAs.
3.3. Payload Optimization at Rekeying Child SAs

The payload optimization at rekeying Child SAs MUST NOT be used unless both peers have indicated their support of this extension by using the negotiation method described in Section 3.1. If the initiator decides to optimize the payloads at the time of rekeying Child SAs, then it includes the SA_TS_UNCHANGED notification in its CREATE_CHILD_SA exchange message. If the initiator decides not to do the optimization, then it just sends the rekeying request message as the original way, the rekeying is conducted as [RFC7296] defined.

This SA_TS_UNCHANGED notification MUST be included in a CREATE_CHILD_SA exchange message when there is no SA and TS payloads included. The new Child SA is created with the SPI value in the SA_TS_UNCHANGED notification.

3.3.1. Rekeying Child SAs When No Change of Initiator and Responder’s Cryptographic Suites and ACL Configuration

At the time of rekeying Child SAs, the initiator MAY send the SA_TS_UNCHANGED notification payload instead of the SA and TS payloads when there is no change in its cryptographic suites and ACL configuration since last negotiation. After receiving the initiator’s request message with the SA_TS_UNCHANGED notification, the responder MAY respond to the initiator with the SA_TS_UNCHANGED notification payload instead of the SA and TS payloads if there is also no change in its cryptographic suites and ACL configuration since last negotiation.

At the time of rekeying a Child SA, when the initiator determines there is no change in its cryptographic suites and ACL configuration since this Child SA was created or last rekeyed, it MAY send the SA_TS_UNCHANGED notification payload instead of the SA and TS payloads in the rekeying request message. In this SA_TS_UNCHANGED notification, it contains the initiator’s new Security Parameter Index (SPI) used for creating the new Child SA.

After receiving the initiator’s rekeying request message with the SA_TS_UNCHANGED notification and no SA and TS payloads, the responder knows that the initiator wants to optimize the rekeying payload. Then when it determines that there is also no change in its cryptographic suites and ACL configuration, the responder MAY send the rekeying respond message to the initiator with the SA_TS_UNCHANGED notification payload instead of the SA and TS payloads. In this SA_TS_UNCHANGED notification, it contains the responder’s new SPI used for creating the new Child SA.
According to the initiator’s new SPI and the responder’s new SPI, the initiator and the responder can rekey the Child SA on both sides.

The CREATE_CHILD_SA message exchange in this case is shown below:

**Initiator**                         **Responder**
----------------------------------------
 HDR, SK {N(REKEY_SA), N(SA_TS_UNCHANGED),
         Ni, [KEi,] } -->

                        <<< HDR, SK {N(SA_TS_UNCHANGED),
                        Nr, [KEr,]}

This SA_TS_UNCHANGED notification MUST be included in a CREATE_CHILD_SA exchange message when there is no SA and TS payloads included at the time of rekeying Child SAs. When the SA_TS_UNCHANGED notification payload is included, the SA and TS payloads MUST NOT be included.

### 3.3.2. Rekeying Child SAs When Responder’s Cryptographic Suites or ACL Configuration Changed

At the time of or before rekeying Child SAs, the responder’s cryptographic suites or ACL configuration may be changed while there is no change of initiator’s cryptographic suites and ACL configuration. In this situation, the initiator MAY send the SA_TS_UNCHANGED notification payload instead of the SA and TS payloads in the CREATE_CHILD_SA request message at the time of rekeying Child SAs.

If the responder decides to continue using the previously negotiated cryptographic suite and Traffic Selectors to rekey the Child SA, it MAY send the SA_TS_UNCHANGED notification payload in the CREATE_CHILD_SA response message, then the rekeying is conducted like Section 3.3.1.

If the responder decides to re-negotiate the cryptographic suite or Traffic Selectors, it MUST send NO_PROPOSAL_CHOSEN notification payload in the CREATE_CHILD_SA response message. After receiving this error notification, the initiator MUST retry the CREATE_CHILD_SA exchange with the SA and TS payloads. Then the rekeying is conducted in the original way defined in [RFC7296]. The CREATE_CHILD_SA message exchange in this case is shown below:
Besides, if the responder only supports the IKE SA rekeying optimization and doesn’t support the Child SA rekeying optimization, it can also follow the way described above, i.e., it MUST send NO_PROPOSAL_CHOSEN notification payload in the CREATE_CHILD_SA response message when receiving the SA_TS_UNCHANGED notification at the time of rekeying Child SAs.

4. Payload Formats

4.1. MINIMAL_REKEY_SUPPORTED Notification

The MINIMAL_REKEY_SUPPORTED notification is used by the initiator and responder to inform their ability of optimizing optional payload at the time of rekeying IKE SAs and Child SAs to the peers. It is formatted as follows:

```
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(=0)| SPI Size (=0) |      Notify Message Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Protocol ID (1 octet) - MUST be 0.
- SPI Size (1 octet) - MUST be 0, meaning no SPI is present.
- Notify Message Type (2 octets) - MUST be <Need to get value from IANA>, the value assigned for the MINIMAL_REKEY_SUPPORTED notification.

This notification contains no data.
4.2. SA_UNCHANGED Notification

The SA_UNCHANGED notification is used to replace the SA payloads at the time of rekeying IKE SAs when there is no change of cryptographic suites in initiator or responder. It is formatted as follows:

```
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 (=8) |      Notify Message Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Security Parameter Index (SPI)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Protocol ID (1 octet) - MUST be 1.
- SPI Size (1 octet) - MUST be 8.
- Notify Message Type (2 octets) - MUST be <Need to get value from IANA>, the value assigned for the SA_UNCHANGED notification.
- SPI (8 octets) - Security Parameter Index. The initiator sends initiator SPI. The responder sends responder SPI.

4.3. SA_TS_UNCHANGED Notification

The SA_TS_UNCHANGED notification is used to replace the SA payloads and TS payloads at the time of rekeying Child SAs when there is no change of cryptographic suites and ACL configuration in initiator or responder. It is formatted as follows:

```
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 (=4) |      Notify Message Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Security Parameter Index (SPI)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Protocol ID (1 octet) - MUST be either (2) to indicate AH or (3) to indicate ESP.
- SPI Size (1 octet) - MUST be 4.
o Notify Message Type (2 octets) - MUST be <Need to get value from IANA>, the value assigned for the SA_TS_UNCHANGED notification.

o SPI (4 octets) - Security Parameter Index. The initiator sends initiator SPI. The responder sends responder SPI.

5. IANA Considerations

This document defines two new Notify Message Types in the "IKEv2 Notify Message Types - Status Types" registry. IANA is requested to assign codepoints in this registry.

<table>
<thead>
<tr>
<th>NOTIFY messages: status types</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINIMAL_REKEY_SUPPORTED</td>
<td>TBD</td>
</tr>
<tr>
<td>SA_UNCHANGED</td>
<td>TBD</td>
</tr>
<tr>
<td>SA_TS_UNCHANGED</td>
<td>TBD</td>
</tr>
</tbody>
</table>

6. Security Considerations

TBD

7. References

7.1. Normative References


7.2. Informative References

Authors’ Addresses

Sandeep Kampati
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: sandeepkampati@huawei.com

Meduri S S Bharath
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: MeduriS.Bharath@huawei.com

Wei Pan
Huawei Technologies
101 Software Avenue, Yuhuatai District
Nanjing, Jiangsu
China

Email: william.panwei@huawei.com