Network Working Group V. Smyslov
Internet-Draft ELVIS-PLUS
Updates: 4555, 6311 (if approved) May 30, 2017
Intended status: Standards Track
Expires: December 1, 2017

Responder Initiated IP Addresses Update in MOBIKE
draft-smyslov-ipsecme-ikev2-r-mobike-00

Abstract

IKEv2 Mobility and Multihoming Protocol (MOBIKE) allows peers to update their IP addresses without re-establishing IKE and IPsec Security Associations (SAs). In the MOBIKE protocol it is the Initiator of the IKE SA, who is responsible for selecting new SA addresses and for initiating the IP addresses update procedure. This document presents an extension to the MOBIKE protocol that allows the Responder to initiate the update.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Internet Key Exchange protocol version 2 (IKEv2), specified in [RFC7296], is a key part of the IP Security (IPsec) architecture. It allows peers to perform authenticated key exchange, which results in establishing IKE Security Association (IKE SA) and to create a data protection channels called IPsec Security Associations (IPsec SAs). In original IKEv2 the IKE and IPsec SAs are established between the IP addresses used in IKEv2 negotiation. The IKEv2 Mobility and Multihoming Protocol (MOBIKE), specified in [RFC4555], extends the IKEv2 functionality by allowing peers to dynamically change IP addresses of the established SAs without the need to re-establish these SAs.

The main use case for the MOBIKE protocol is a remote access user that travels and moves from one from one IP address to another without re-establishing existing SAs with the VPN gateway. However, the MOBIKE also supports more complex scenarios when VPN gateway is multihomed and its addresses may change over time.

In the MOBIKE it is the Initiator (e.g. the remote access client) who is responsible for detecting the working IP addresses pairs and for deciding which pair to use. In other words, the Responder (e.g. the VPN gateway) plays a passive role and could neither initiate the IP address update process nor tell the Initiator which IP address is

preferred to use. This limitation makes use of complex scenarios less efficient and decreases the value of MOBIKE protocol.

For example, if the VPN gateway is a load sharing cluster where each node has its own IP address, then the cluster must be able to move SA between nodes depending on their current load. Currently Redirect Mechanism for IKEv2 [RFC5685] can accomplish this task, however it requires IKE SA to be re-established, that is very inefficient. Another possible solution is to use IKE SA Cloning along with the MOBIKE (see [RFC7791] for scenario description), but the limitation of the MOBIKE protocol makes this problematic. Obviously, the client has insufficient information to select when and to which of cluster IP addresses to move an SA to and the VPN gateway has no means to provide the client with this information.

This specification extends the MOBIKE protocol by adding ability for the Responder to ask the Initiator for IP address update and to provide it with the new IP address to use.

2. Terminology and Notation

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 [RFC2119].

In this document the term "Initiator" means the party who originally initiated the first IKE SA (in a series of possibly several rekeyed IKE SAs), and "Responder" means the other party. This is consistent with a way these terms are used in [RFC4555]. Note, that in [RFC7296] the terms "original initiator" and "original responder" mean the party, who initiated (or responded to) the latest IKE SA in a series of possibly several rekeyed IKE SAs.

3. Protocol Overview

The MOBIKE protocol is designed in such a way, that it is the IKE SA Initiator, who is responsible for performing the actions concerned with the selecting of a working IP addresses pair and for initiating an IP addresses update exchange. Usually the Initiator selects an IP addresses pair by continuously probing different pairs and choosing the working one. If several pairs work then the choice between them is arbitrary. The Responder cannot influence the process of selecting and cannot ask the client to immediately switch to a particular gateway's address. As a result the process of selection a new pair takes substantial time and may ends up with a suboptimal path. Moreover, in case the Responder isn't multihomed (and thus doesn't provide the Initiator with a list of additional IP

addresses), the change of its IP address cannot be handled by the MOBIKE.

Obviously, this limitation comes from the fact that there might be middleboxes on the path (like Network Address Translators (NAT) or firewalls) that might disallow IP packets to come from VPN gateway to the client unless the client first contacts the VPN gateway. For example, the client might reside behind a dynamic NAT that creates a mapping when IP packet first come from the client to the gateway. If the gateway tries to send an IP packet to the client from different IP address, the packet would be dropped since the NAT box has no corresponding mapping.

This specification provides the following solution to the described problem. When the Responder decides that its end of existing SA should be switched from its original IP address IP_R1 to a new address IP_R2, it initiates an INFORMATIONAL exchange containing a new notification SWITCH_TO_IP_ADDRESS, that contains IP_R2. The request message of this exchange is sent from IP_R1 address, so that an existing middlebox mappings are used and the message can reach the Initiator. However, the response message is sent to a newly presented IP_R2 address, so that a new middlebox mappings are created. Once the Initiator completes exchange containing SWITCH_TO_IP_ADDRESS notification, it immediately initiates standard MOBIKE procedure for updating SA addresses by starting the INFORMATIONAL exchange containing UPDATE_SA_ADDRESSES notification.

4. Protocol Description

4.1. Capability Advertising

According to [RFC4555], the peers must exchange MOBIKE_SUPPORTED notifications in the IKE_AUTH exchange before they can use the MOBIKE protocol. If the Initiator supports this specification and is willing to use it, then it MUST include a single octet 0x52 ('R') in the notification data of the MOBIKE_SUPPORTED notification sent to the Responder. There is no need for the Initiator to know whether the Responder supports this specification or not, so the MOBIKE_SUPPORTED notification sent by the Responder has an empty notification data.

Note, that [RFC4555] specifies that MOBIKE_SUPPORTED notification must contains no data when sending and the content of the notification data must be ignored while parsing. So, So, if the Responder doesn't support this specification, it will just ignore the content of the MOBIKE_SUPPORTED notification and will use MOBIKE without this extension.

```
   (IP_I1:500 -> IP_R1:500)
   HDR, SAi1, KEi, Ni,
            N(NAT_DETECTION_SOURCE_IP),
            N(NAT_DETECTION_DESTINATION_IP)  -->

                             <--  (IP_R1:500 -> IP_I1:500)
                                  HDR, SAr1, KEr, Nr,
                                       N(NAT_DETECTION_SOURCE_IP),
                                       N(NAT_DETECTION_DESTINATION_IP)

   (IP_I1:4500 -> IP_R1:4500)
   HDR, SK { IDi, CERT, AUTH,
            SAi2, TSi, TSr,
            N(MOBIKE_SUPPORTED('R')) }  -->

                             <--  (IP_R1:4500 -> IP_I1:4500)
                                  HDR, SK { IDr, CERT, AUTH,
                                            SAr2, TSi, TSr,
                                            N(MOBIKE_SUPPORTED),
                                            N(ADDITIONAL_IP4_ADDRESS) }
```

4.2. Responder Initiated IP Address Update

If the Initiator advertised its support for this specification during the initial exchange as described in Section 4.1, then the Responder is free to initiate IP Address Update request at any time. If the Initiator doesn't indicate its support for this extension, then the Responder MUST NOT initiate IP Address Update request. The IP Address Update request NUST NOT be initiated by the Initiator, the Responder MUST take no action if it receives such a request (apart from sending an empty response message to complete the exchange).

It is up to the Responder to decide when to initiate an IP Address request and what new address to include into it. Some of the possible reasons are:

- o Responder's IP address is changed due to Network Interface Card (NIC) reconfiguration

- o Responder is multihomed and wishes to switch SA to a different IP address

- o Responder is a cluster and wishes to move SA to a different node having its own IP address

The Responder requests the Initiator to update SA Address by initiating the INFORMATIONAL exchange containing a new status type notification SWITCH_TO_IP_ADDRESS. The notification data of this

notification contains a new IP address the Responder requests the Initiator to use for the IKE SA and its Child SAs. Note, that the exchange request message MUST be sent using old SA addresses. In the example below the SA was established using IP_I1 and IP_R1 addresses for the Initiator and Responder respectively, and the Responder wishes to change the address of its end of the SA to IP_R2. So, it initiates the INFORMATIONAL exchange from IP_R1 address containing the SWITCH_TO_IP_ADDRESS notification with IP_R2 address. However, since the response message should come on a new address (IP_R2), at this point the Responder MUST be able to receive packets on the IP address it included in the SWITCH_TO_IP_ADDRESS notification.

```
                      <--  (IP_R1:4500 -> IP_I1:4500)
                           HDR, SK { N(SWITCH_TO_IP_ADDRESS(IP_R2)) }
```

Since the request is sent using old SA addresses, it is expected to pass through the middleboxes and reach the Initiator because it must use existing mappings.

Upon receiving the SWITCH_TO_IP_ADDRESS notification the Initiator extracts its content and makes a decision whether the received IP address is appropriate for the SA. If the received IP address is among the addresses previously received from the Responder in ADDITIONAL_IP4_ADDRESS or ADDITIONAL_IP6_ADDRESS notifications, then it is definitely appropriate for the SA. Otherwise local policy must be consulted to decide whether the received IP is appropriate. If the address is considered inappropriate, then the Initiator MUST complete the exchange by sending an empty message to an old address (IP_R1) and continue to use this address. It is RECOMMENDED that the Initiator immediately initiates Liveness Check exchange to ensure that the Responder is able to operate using old address.

```
(IP_I1:4500 -> IP_R1:4500)
HDR, SK {}  -->
```

If the Initiator decides that the received address is appropriate, it completes the exchange by sending an empty response message to the newly received address (IP_R2). Since the response message to the new Responder's address flows in the original direction (from the Initiator to the Responder), it should create new mappings in middleboxes, thus allowing further communication between them. After the response message is sent the Initiator MUST immediately initiate an IP address update procedure according to the MOBIKE specification by sending the INFORMATIONAL exchange request message containing the UPDATE_SA_ADDRESSES notification. See [RFC4555] for details. As a result, the remote IP address of the SA is changed from IP_R1 to IP_R2. Note that only the IP address is changed, the port remains the same.

```
(IP_I1:4500 -> IP_R2:4500)
HDR, SK {}  -->

(IP_I1:4500 -> IP_R2:4500)
HDR, SK { N(UPDATE_SA_ADDRESSES),
          N(NAT_DETECTION_SOURCE_IP),
          N(NAT_DETECTION_DESTINATION_IP),
          N(COOKIE2) }  -->

                          <--  (IP_R2:4500 -> IP_I1:4500)
                               HDR, SK { N(NAT_DETECTION_SOURCE_IP),
                                    N(NAT_DETECTION_DESTINATION_IP),
                                    N(COOKIE2) }
```

The Responder MUST NOT change IP address of the SA until it receives the UPDATE_SA_ADDRESSES notification from the Initiator. Note, that there is no need for the Responder to perform Return Routability check once the addresses are updated since it itself requested to change IP address of the SA and it successfully received a response from the Initiator sent to the new address. However, depending on the Responder's policy, the Return Routability check MAY be performed.

If the Responder doesn't receive a response message on a request containing the SWITCH_TO_IP_ADDRESS notification after several retransmissions, then it means that either request or response message cannot use the new path and pass through the middleboxes. In this case the Responder's behavior depends on whether it advertised additional IP addresses before and whether old SA address is still available.

If old SA address is unavailable and no alternative addresses were advertised before, then the IKE SA and all associated Child SAs MUST be torn down. Otherwise the SA MAY be kept in an anticipation that the Initiator after some time detects the old IP address failure itself and performs IP addresses update.

4.2.1. High Availability Cluster Scenario

In case the VPN gateway is a cluster consisting of several nodes each having its own IP address, both Load Sharing (LS) and High Availability (HA) goals may be achieved. For the purposes of HA the nodes share an IKE SA state while only one of them communicate with the IKE SA peer at any given time. Of the active node fails, the other nodes detect this fact and select a new active node for the SAs the failed node served. The selected node must then instruct the failed node peers to switch their SAs to a new IP address using this specification.

Since some exchanges might be in progress when the active node fails, some special measures must be taken to ensure that the IKE SA state is synchronised between the new active cluster node and the client. Protocol Support for High Availability of IKEv2/IPsec [RFC6311] describes the necessary measures. In particular, the new active node initiates the INFORMATIONAL exchange containing the IKEV2_MESSAGE_ID_SYNC notification and optionally the IPSEC_REPLAY_COUNTER_SYNC notification. [RFC6311] states that no other payload must be included in this exchange. However, in case the IP address of the new active node differs from the IP address of the failed active node it is necessary to combine the IKEV2_MESSAGE_ID_SYNC and the SWITCH_TO_IP_ADDRESS notifications in one exchange. So, this specification updates [RFC6311] in this regard: if HA cluster nodes have different IP addresses then in case of failover the request to synchronize Message IDs and the request to change IP address MUST be sent together in the same INFORMATIONAL exchange.

```
                      <--  (IP_R1:4500 -> IP_I1:4500)
                           HDR, SK { N(SWITCH_TO_IP_ADDRESS(IP_R2))
                                     N(IKEV2_MESSAGE_ID_SYNC),
                                     [N(IPSEC_REPLAY_COUNTER_SYNC)] }

(IP_I1:4500 -> IP_R2:4500)
HDR, SK { N(IKEV2_MESSAGE_ID_SYNC) }  -->
```

Once this exchange is completed the client MUST immediately perform an IP address update procedure according to the MOBIKE specification as described in Section 4.2.

5. Payload Formats

5.1. MOBIKE_SUPPORTED Notification

The MOBIKE_SUPPORTED Notification is defined in [RFC4555], Section 4.2.1 with the Notify Message Type 16396. This definition requires the notification data to be empty while sending and to be ignored when notification is received.

This document updates the definition from [RFC4555]. Exchange Initiator sets the notification data of the MOBIKE_SUPPORTED Notification to a single octet 0x52 ('R') to indicate that this specification is supported.

5.2. SWITCH_TO_IP_ADDRESS Notification

The Notify Message Type for this notification is <TBA by IANA>. The notification data contains new Responder's IP address.

For IPv4, the notification data is 4 octets long and is defined as follows:

```
                     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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 New Responder's IPv4 Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

For IPv6, the notification data is 16 octets long and is defined as follows:

```
                     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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                 New Responder's IPv6 Address                  |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Protocol ID and SPI Size fields are set to zero.

6. Security Considerations

This specification is an extension of the MOBIKE protocol, so the Security Considerations described in [RFC4555] are applied.

7. IANA Considerations

This document defines new Notify Message Types in the "IKEv2 Notify Message Types - Status Types" registry:

```
  <TBA>       SWITCH_TO_IP_ADDRESS
```

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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", RFC 4555, DOI 10.17487/RFC4555, June 2006, <http://www.rfc-editor.org/info/rfc4555>.

[RFC6311] Singh, R., Ed., Kalyani, G., Nir, Y., Sheffer, Y., and D. Zhang, "Protocol Support for High Availability of IKEv2/IPsec", RFC 6311, DOI 10.17487/RFC6311, July 2011, <http://www.rfc-editor.org/info/rfc6311>.

[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, <http://www.rfc-editor.org/info/rfc7296>.

8.2. Informative References

[RFC5685] Devarapalli, V. and K. Weniger, "Redirect Mechanism for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5685, DOI 10.17487/RFC5685, November 2009, <http://www.rfc-editor.org/info/rfc5685>.

[RFC7791] Migault, D., Ed. and V. Smyslov, "Cloning the IKE Security Association in the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 7791, DOI 10.17487/RFC7791, March 2016, <http://www.rfc-editor.org/info/rfc7791>.

Author's Address

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

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