

IPSECME Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 5, 2010

D. Migault  
Orange Labs R&D  
Sep 2009

IPsec mobility and multihoming requirements : Problem statement  
draft-mglt-ipsec-mm-requirements-00.txt

## Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on March 5, 2010.

## Copyright Notice

Copyright (c) 2009 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 in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

## Abstract

Currently IPsec mobility is the purpose of MOBIKE [[RFC4555](#)] which is the IKEv2 multihoming and mobility extension. More specifically, MOBIKE mobility support provides the ability to change the outer IP

address of a tunnel mode Security Association. On the other hand MOBIKE multihoming support provides the ability of a peer to inform its correspondent that alternate IP addresses may be used if the current IP address does not work any more. This draft presents requirements to extend mobility and multihoming facilities. This includes the use of simultaneous IP addresses as well as other IPsec mode like transport mode.

## Table of Contents

<a href="#">1.</a>	<a href="#">Requirements notation</a>	<a href="#">4</a>
<a href="#">2.</a>	<a href="#">Introduction</a>	<a href="#">4</a>
<a href="#">3.</a>	<a href="#">Terminology</a>	<a href="#">4</a>
<a href="#">4.</a>	<a href="#">Mobility Scenarios</a>	<a href="#">5</a>
<a href="#">4.1.</a>	<a href="#">Move and Re-connect Scenario</a>	<a href="#">6</a>
<a href="#">4.2.</a>	<a href="#">Pre-connect and Move Scenario</a>	<a href="#">6</a>
<a href="#">5.</a>	<a href="#">Multihoming Scenarios</a>	<a href="#">7</a>
<a href="#">5.1.</a>	<a href="#">Different Multihoming, Different Layers</a>	<a href="#">7</a>
<a href="#">5.2.</a>	<a href="#">Asymmetric Communications</a>	<a href="#">8</a>
<a href="#">5.3.</a>	<a href="#">Multihoming Scenario : Simultaneous IP Addresses</a>	<a href="#">9</a>
<a href="#">5.4.</a>	<a href="#">Multihoming Scenario : Alternate IP addresses</a>	<a href="#">11</a>
<a href="#">6.</a>	<a href="#">The Case of IKEv2</a>	<a href="#">11</a>
<a href="#">7.</a>	<a href="#">Multihoming for Mobility actions</a>	<a href="#">13</a>
<a href="#">8.</a>	<a href="#">Mobility and Multihoming : complementary actions</a>	<a href="#">13</a>
<a href="#">9.</a>	<a href="#">Related work</a>	<a href="#">15</a>
<a href="#">9.1.</a>	<a href="#">IPsec</a>	<a href="#">15</a>
<a href="#">9.2.</a>	<a href="#">IKEv2</a>	<a href="#">16</a>
<a href="#">9.3.</a>	<a href="#">MOBIKE</a>	<a href="#">18</a>
<a href="#">9.4.</a>	<a href="#">MIPv6</a>	<a href="#">19</a>
<a href="#">9.5.</a>	<a href="#">MIPv6 and MOBIKE</a>	<a href="#">20</a>
<a href="#">9.6.</a>	<a href="#">SHIM6</a>	<a href="#">20</a>
<a href="#">9.7.</a>	<a href="#">SCTP</a>	<a href="#">20</a>
<a href="#">9.8.</a>	<a href="#">mtcp</a>	<a href="#">20</a>
<a href="#">9.9.</a>	<a href="#">HIP</a>	<a href="#">20</a>
<a href="#">10.</a>	<a href="#">Mobility / Multihoming with IKEv2/IPsec mechanisms</a>	<a href="#">23</a>
<a href="#">10.1.</a>	<a href="#">Possible Mobility Scenarios</a>	<a href="#">23</a>
<a href="#">10.1.1.</a>	<a href="#">Analyze</a>	<a href="#">23</a>
<a href="#">10.1.2.</a>	<a href="#">Requirements</a>	<a href="#">24</a>
<a href="#">10.2.</a>	<a href="#">Possible Multihoming Scenarios</a>	<a href="#">24</a>
<a href="#">10.2.1.</a>	<a href="#">Analyze</a>	<a href="#">24</a>
<a href="#">10.2.2.</a>	<a href="#">Requirements</a>	<a href="#">25</a>
<a href="#">10.3.</a>	<a href="#">MOBIKE and our Scenarios</a>	<a href="#">25</a>
<a href="#">10.3.1.</a>	<a href="#">Analyze</a>	<a href="#">25</a>
<a href="#">10.3.2.</a>	<a href="#">Requirements</a>	<a href="#">26</a>

<a href="#">11.</a>	Mobility Rejected by the Responder? . . . . .	<a href="#">27</a>
<a href="#">12.</a>	Scope and Restrictions . . . . .	<a href="#">28</a>
<a href="#">13.</a>	Acknowledgments . . . . .	<a href="#">29</a>
<a href="#">14.</a>	Security Considerations . . . . .	<a href="#">30</a>
<a href="#">15.</a>	IANA Considerations . . . . .	<a href="#">30</a>
<a href="#">16.</a>	References . . . . .	<a href="#">30</a>
<a href="#">16.1.</a>	Normative References . . . . .	<a href="#">30</a>
<a href="#">16.2.</a>	Informative References . . . . .	<a href="#">31</a>
	Author's Address . . . . .	<a href="#">31</a>

## [1.](#) Requirements 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](#)].

## [2.](#) Introduction

This draft presents the scenarios we consider. For sake of simplicity, we spitted the scenarios into three different categories by considering mobility-only scenarios with non multihomed peers, multihoming-only scenarios without mobility, and then scenarios where mobility and multihoming are combined together. The latter category is in fact more a discussion on interaction between mobility and multihoming.

Secondly, we present a state of the art of different protocols related to IPsec, mobility and multihoming. This section presents briefly the protocols, and specific points related either to mobility, multihoming or IPsec. This section should be considered more like a key note section than a normative section.

At last, we derive the requirements to fit our scenarios. From the scenarios section and the state of the art section, we present the existing mechanisms that best match our scenarios, and figure out what is required to match them completely. Thus we consider the mobility scenarios, then the multihoming scenarios, and for each of them derive requirements. We added a special section with MOBIKE

[[RFC4555](#)]. In fact the scenarios of this draft are using the IPsec transport mode, and MOBIKE [[RFC4555](#)] is the IKEv2 [[RFC4306](#)] extension that deals with mobility and multihoming with the tunnel mode. We point out how MOBIKE is related and what takes to MOBIKE so that it matches our scenarios.

We assume the reader is familiar with IPsec [[RFC4301](#)], IKEv2 [[RFC4306](#)] and with MOBIKE [[RFC4555](#)].

### [3.](#) Terminology

- Mobile Node (MN) : In this draft the Mobile Node is the peer that performs the mobility action. The Mobile Node does not have to be understood as in MIPv6.

Migault

Expires March 5, 2010

[Page 4]

---

Internet-Draft

IPsec with mobility and multihoming

Sep 2009

- Initiator : The Initiator is the peer that initiates the exchange. It sends a message to the Responder. It is important to note that if two peers are connected, the Initiator of one exchange can be the Responder of another exchange. When a mobility action is performed then the Initiator is also the Mobile Node.
- Responder : The Responder is the peer receiving a message. The message is sent from the Initiator.
- Alternate IP Address : The alternate IP address of a peer is the IP address a peer is not currently using but that might be used latter. An alternate IP address should used only if the current IP address does not work anymore.

### [4.](#) Mobility Scenarios

This section shows mobility scenarios that motivated this draft. They consider two peers directly connected between each other. The communication is protected using IPsec with a transport mode -- the tunnel mode has already be considered in [[RFC4555](#)].

Mobility scenarios with transport mode do not provide seamless

mobility -- at least without multihoming considerations -- and so is not transparent to the application. Thus it should not be used with any applications. In fact, we consider two peers directly connected with a single IP address -- i.e. not multihomed -- and their connection is protected by an IPsec Security Association using the transport mode. A mobility action requires the Mobile Node (MN) to change its IP address, to inform the other peer so that the already negotiation can still be used with the new IP address. This interrupts the communication and data lost might occurs. This kind of mobility can then only apply to certain use cases.

Typical examples can be UDP connections protected via IPsec. The peer decides to move, and packets that do not reach their destination are lost. The faster the connection is reestablished the fewer packets we loose. Another example can be TCP connections with very few packets transfer compared to the connection time. Web surfing is a good example for this case. Suppose an end user is connected, for example to a web server, and the communication is protected by IPsec using the transport mode. The time between sending a request and receiving the web page is very low compared to the time the end user spend on the web server to read the data. Furthermore end users are used to reload a web page when the download is not performed correctly. When the peer decides to move, if a web page is being downloaded, the connection is broken, and once mobility is performed, the user reloads the page. On the other hand, if the peer moves while the end user reads a web page, than the end user does not even

notice the mobility. Of course this mobility can be used if mobility actions are not too frequent, and if the peer changes its IP address every second then it is highly probable the end user will be affected by the mobility. In other words, walking speed is probably the considered speed for mobile and non-multihomed peers.

End users connected for example to a ftp server for heavy file transfer is not a use case that matches the mobile and non multihomed peer. In that specific case, unless the application deals with connection interruption, the transfer needs to be restarted each time a the peer change its IP address.

#### 4.1. Move and Re-connect Scenario

This scenario considers two peers connected via an IPsec Security

Association using transport mode, and using only one IP address. One peer loses its connectivity, manages to attach on another network, to get a new IP address, and send inform the other peer the communication can go on a new IP address. Signaling messages are sent via the IKEv2 protected channel, since this channel has already been negotiated. The MN updates its Security Associations before sending the message to the other peer. When the other peer receives the message, it checks the new IP address matches the local policies, the IPsec Security Policies, eventually performs some Return Routability Check before updating the Security Association Database. Once the SAD are updated on both peers, the communication can go on. The previously negotiated IPsec / IKEv2 security parameters remain the same. Particularly, authentication do not need to be replayed.

In this scenario, the MN has no guarantee that the new IP address will match either the Security Policies or the local policies of the peer. If the new IP address is not accepted, a CREATE\_CHILD\_SA or an IKE\_INIT exchange MAY be performed.

#### [4.2.](#) Pre-connect and Move Scenario

This scenario considers two peers connected via an IPsec Security Association using transport mode. Each peer only uses one IP address. Unlike in the previous scenario, the Mobile Node (MN) knows the IP address it will use before performing the mobility action. How the MN knows the IP address is beyond the scope of this paper, and we call it the new IP address as opposed to the current IP address. The MN checks with the other peer if a mobility action can be performed with this new IP address. This involves checking the local policies, the Security Policy Database.

As mentioned in the multihoming section, the MN may eventually, before moving, informs the other peer that the new IP address is an

alternate IP address. If the MN is multihomed checking the new IP address can eventually include reachability test with the Return Routability Check exchange. In this scenario this test cannot be performed since the MN has only one interface.

The MN moves as described in the previous scenario. It informs the Responder it has changed its IP address.

Motivation for preparing the mobility action are :

- Helps the MN to choose the next IP address to use : The Responder can check the IP address matches the local policies and send an error message if the IP address does not match the local policies. With an error message, the MN knows that using this IP address will require to re-negotiate an IPsec Security Association. If the MN can choose between a set of IP address then it might choose an IP address that does not require negotiating a new IPsec Security Association.
- Fasten Mobility action : When the MN moves, it can either send a message with the old IP address or the new IP address. If the MN knows the new IP address does not matches local policies of the Responder, than it might directly initiate an IKEv2 negotiation. If the MN knows the IP address matches the Responder's local policies, then it might avoid such a negotiation.

## [5.](#) Multihoming Scenarios

### [5.1.](#) Different Multihoming, Different Layers

Multihoming is the ability of a peer to handle with more than one IP address. It can have different meanings, and this section defines the different multihoming we consider in this paper. Multihoming depends on the layer that deals with multihoming. We consider two layers in this paper the Networks or the Application layers. Networks Layers are all layers below the application layer. Multihoming can also consider different strategies. We consider the use of Simultaneous IP Addresses (SM) as opposed to the use of Alternate IP Addresses (AM). Alternate IP address SHOULD NOT be used unless the current IP address does not work anymore.

- Multihomed Application : An application is multihomed if it can take advantage of multiple IP addresses. If the application considers SM, then the peer has to deal with multiple IP addresses, for example by having multiple interfaces. If the application considers AM, then the peer can be singled homed. If application deals directly with multihoming, there MUST be a



provide multihoming information to the other peer. This channel can be established by the application itself, but application can also decide to use an already existing channel, or protocol. In fact an application could use the IKEv2 channel to carry multihoming information. This would require in that case a communication channel between IKEv2 and the application.

- Multihomed Networks Layer : Multihoming can also be transparent to the application. SHIM6 [[RFC5533](#)] or HIP [[RFC4423](#)], [[RFC5201](#)] are protocols that provide an additional layer between the IP and the application layer. As a result the application only sees one fix IP address. This IP address might be a non routable IP address, and application packets can be routed using different IP addresses. In those cases, the networks layers deals with multihoming. We say Network(s) to include the transport layer. In fact it is still not clear which layer should consider multihoming, and multipath tcp are looking at the transport layer. So do SCTP. Unless the application requires a specific multihoming strategy, Multihomed Networks Layer brings simplicity for application developers. On the other hand the application completely relies on the network layer multihoming management facilities.
- Simultaneous IP addresses Multihoming (SM): We call Simultaneous IP Addresses Multihoming (SM) the ability to use more than one IP address. This means that an application sends / receives data from / to multiple IP addresses. If Multihoming is considered at the Network layer, then it means a peer receives / sends data from two distinct IP addresses. Of course we mean routable IP addresses! Usual motivations for SM are bandwidth aggregation, data duplication, traffic engineering, interface selection...
- Alternate IP addresses Multihoming (AM): We call Alternate IP Address Multihoming (AM) the ability to have alternate IP addresses, that is to say IP addresses that SHOULD be used only if the current IP address is not working anymore. The main motivation for using the AM mode is to enhance reachability. Providing multiple alternate IP addresses also enhance connectivity. If the peer provides more than one IP address, and the connection is lost, then the other peer can choose the best IP address.

## [5.2.](#) Asymmetric Communications

This section considers use cases when peers communication benefit from multihoming. We need to consider the following definitions :

- A peer is multihomed : if it has multiple IP addresses.
- A peer supports multihoming : if it has all multihoming facilities, that is to say if has the ability to deal with different IP address at the "network layer" -- that is to say network or transport layer.
- An application supports multihoming : if the application handles multihoming properly. We assume in this paper that when an application handles with multihoming, network layer multihoming abilities are skipped.

When an application supports multihoming and one of the host is multihomed, then the communication can take advantage of multihoming capabilities. When an application does not support multihoming, to take advantage of the multihoming, both peers network layer MUST support multihoming and one of the host MUST be multihomed. For simplicity, we also assume that applications do not follow the client server model with BYPASS security policy. The table below sums up cases when the communication between applications can take advantage of multihoming.

Multihoming supported layer						
Application	Peer A		Peer B		Benefit from	
	Network	Host	Network	Host	multihoming	
X	*	X	*	*	X	
X	*	*	*	X	X	
X	*	-	*	-	-	
-	X	-	X	-	-	
-	X	*	X	X	X	
-	X	X	X	*	X	
-	*	*	-	*	-	
-	-	*	*	*	-	

When do we benefit from Multihoming

### 5.3. Multihoming Scenario : Simultaneous IP Addresses

The purpose of using simultaneous IP addresses is to associate a peer with a pool of IP addresses. Each IP address of the pool can be used to reach that peer, and peer A can use simultaneous IP addresses while communication with peer B.

among the different IP addresses is beyond the scope of the draft. This draft considers that if peer A decides to use simultaneously multiple IP addresses, then Security Associations between peer A and peer B MUST be set so that traffic can use both of those IP addresses. In the same perspective, if peer A changes its pool of IP address Security Association between peer A and B MUST be updated.

Thus the use case to consider here, is peer A and peer B are communicating, and their communication is protected by a transport mode Security Association. Peer A has multiple interfaces and proceeds to an attachment procedure on different networks. Once the attachment procedure is over, peer A has multiple IP addresses, and wants to benefit from them in its communication with peer B. We assume that this communication can take advantage of the multihoming facilities, and that peer A and peer B set their IPsec Security Associations.

In our case, setting the IPsec Association means that the already configured SA is being associated multiple IP addresses. Multihoming action consists then to add or remove IP addresses associated to that specific Security Association.

When a peer adds an IP address to a given Security Association, it sends a message to inform the other peer. If the IP address matches the local and Security Policies, then it is added to specified SA(s). If the IP address does not match either the local policies or the Security Policies, or both of them, then an error is returned. One can notice that there is no need to check that the IP address matches the local policies or the Security Policies before proceeding to the multihoming action. Event if the IP address is refused, the communication is not affected. This was not the case with mobility.

When a peer wants to remove an IP address from an existing SA, it sends a message to the other peer. The other peer updates its IPsec databases.

Note : On an IPsec point of view, adding or removing one IP address from Security Associations does not present any difficulties. Nevertheless one should consider its consequences on the IP traffic.

Adding an IP address to a given SA do not affect the traffic. On the other hand removing one IP address from an SA might affect the communication between the peers. For now, ULP handle multihoming without any considerations of IPsec -- except for HIP. Once the transport or the 3.5 layer agrees on removing the IP address, then IPsec databases can be updated, and the IP address removed from the SA(s). One possibility is to initiate an IKEv2 exchange. Another possibility is the ULP proceed directly to the IPsec update without proceeding the IKEv2 exchange. This would at least avoid one

exchange. There is also another way to consider how the different layers can interact between each other. We can consider that mobility and multihoming signalization should be protected and uses an IKEv2 channel. In that case, the multihoming or mobility action could be transmitted to the upper layers.

#### 5.4. Multihoming Scenario : Alternate IP addresses

A peer provides an Alternate IP Address to the other peer so that if the peer is not anymore reachable on one of the currently used IP addresses, than it might still be reachable on one of the Alternate IP Address. An Alternate IP Address should not be used in addition to the currently IP addresses.

Suppose peer B reaches peer A through 2 IP addresses, and peer A has provided a pool of Alternate IP Addresses to peer B. Peer A happens not to be reachable on one of the IP address. Peer B can still reach peer A through one IP address. It is up to peer B's local policies to define whether or not it should add and use an Alternate IP Address.

The IPsec layer does not deal directly with reachability statement. This means that Alternate IP Addresses MUST be provided either to the application, or the entity that is in charge of the multihoming. Thus Alternate IP Addresses are useful for IKEv2 as an application or if IKEv2 is used as a channel to carry multihoming information that are provided to the Upper Layer Protocol.

The Alternate IP Address mechanism for the IKEv2 application is described in MOBIKE [[RFC4555](#)].

## 6. The Case of IKEv2

IKEv2 is a special case. On the one hand that is a regular application and it has its own strategy to handle multihoming. On the other hand IKEv2 is the application we consider to carry mobility and multihoming information. Most of the time this information is expected to affect other applications connections then IKEv2's connections. In other words, IKEv2 is to be considered as a secure channel that is used to carry signaling information.

This paper is not especially focused on how IKEv2, as an application deals with mobility and multihoming. The scope of this paper is to elaborate on how peers can deal with mobility and multihoming. In that sense peers need to exchange messages which affect the Security Associations of their communications. Such messages are exchanged via the IKEv2 channel. Thus we MUST define what kind of message need

Migault

Expires March 5, 2010

[Page 11]

---

Internet-Draft

IPsec with mobility and multihoming

Sep 2009

to be exchanged. On the other hand, to be exchanged, we need to provide mechanisms so that the IKEv2 channel can also survive to mobility and multihoming actions. In that sense we MUST also specify how IKEv2 deals with mobility and multihoming.

As an application IKEv2 has specific IPsec security policies. Packets are not filtered and are sent by the network layer to the IKEv2 application. IKEv2 binds the message to its IKE\_SA thanks to the SPI. This means that by using different IP addresses, one peer SHOULD be able to reach the other. The way IKEv2 works allows the Initiator to use multiple IP addresses, it will receive the answer from the Responder on the same IP address used for the query.

Nevertheless we cannot say IKEv2 is using the Simultaneous IP Address Multihoming. IKEv2 works like a server and send the response to source IP address of the query. In that sense peer A can use various IP addresses to send IKEv2 message to peer B. But peer B will always use the IP address associated to the IKE\_SA to reach peer B.

For now, IKEv2 even with the MOBIKE extension associates only one IP address to the IKE\_SA. So IKEv2 even with the MOBIKE extension do not consider Simultaneous IP Addresses Multihoming. If it would, then it would mean that peer A and peer B could have been associated a pool of IP address. Each time one of the peer want to reach the other it would be able to choose one the IP address of the pool.

More specifically, this is different from receiving one packet on one IP address and sending the response on another IP address.

Considering the previous example, where peer A uses different IP addresses that are not associated to the IKE\_SA. Automatically adding those IP address to the IKE\_SA is a bad idea. In fact it would not consider that peer A may use IP address it may be not or it does not want to be reach with. This means that multihoming and IKEv2 SHOULD NOT be performed automatically.

If one peer changes its IP address, it MUST explicitly inform the IKEv2 application. MOBIKE [[RFC4555](#)] deals with this by sending an UPDATE\_SA\_ADDRESSES message, that indicates the IP address of the sender has changed. This is what mobility occurs, but for now IKEv2 uses only one IP address, so adding or removing an IP address is not considered.

The way IKEv2 considers multihoming is described in MOBIKE [[RFC4555](#)], and uses the Alternate IP address Multihoming. Alternate IP addresses are sent from one peer to the other by using ADDITIONAL\_IP4\_ADDRESS or ADDITIONAL\_IP6\_ADDRESS Notify Payloads.

## [7.](#) Multihoming for Mobility actions

Until now, mobility and multihoming have been treated as different actions, but one can easily see that multihoming can also be used for a seamless mobility. This section analyzes when mobility is performed with multihoming actions.

Multihoming provides the ability to perform a smooth transition from one IP address to another IP address without interrupting the traffic. One peer is connected to the other via a pool of IP addresses. When the peer moves, it can get a new pool of IP addresses. If the intersection between the two pools is not void, then a smooth transition can occur by using both multihoming and eventually mobility mechanisms.

Suppose peer A and peer B are communicating using a pool of IP addresses. Peer A moves and is attached to other different networks. It proceeds to a multihoming add action and add all new IP addresses.

When some of the old IP addresses become less reliable -- less power on the signal for example -- then peer A decides to remove them from the communication. To perform this action peer A has two possibilities. It can use a multihoming remove action or proceed to a mobility action replacing the IP address to be removed by newly acquired IP address.

On a cross layer point of view, as long as it is coordinated, it does not make any difference whereas the peer update or remove IP addresses. This means that ULP MUST be involved. In fact, ULP can proceed to the IKEv2 exchanges once the multihoming actions has been performed at the IP and transport layer. On the other hand, IPsec multihoming signalization can also be forwarded to ULP. In both cases, proceeding to mobility or multihoming remove message does not change anything.

On IPsec point of view it is recommended to perform multihoming remove action. In fact multihoming remove message are expected to be shorter than mobility message. Mobility action may need to specify the replacing and the replaced IP address whereas multihoming remove message only require the IP address to be removed.

## 8. Mobility and Multihoming : complementary actions

The actions we consider in this paper are Mobility and Multihoming(s). The previous section exposes how mobility can be performed with multihoming actions. So why do we need mobility? This section exposes the differences between mobility and multihoming.

Some of the different uses between Multihoming and Mobility are listed below :

- Mobility (or update) action can, in some cases, be replaced by two Multihoming actions (add followed by a delete). Thus the main advantage of Mobility is that only one request is required whereas two are required with the use of Multihoming action.
- Multihoming can be used to perform mobility action. Considering the IPsec layer only is not the most efficient way, but nothing can prevent a peer from doing it. On a cross layer point of view, this might require the peer or the application

- supports multihoming.
- Multihoming does not necessarily affect the current communication. An application can communicate via one pair of IP address and make independent tests with another pair to check for example if the new IP address matches the local policies, or is still reachable. Such tests can be useful before proceeding to a Mobility action for example. The Mobility has a best effort approach. It changes the IP address, if that's possible. If that is not possible, and the peer has to change its IP address, than the connection is interrupted. This best effort approach can be mitigated with the mobility checks.
  - If an IPsec connection is broken, Mobility has a specific mechanism to re-establish this connection. This can happen when a host has to move, initiates a Mobility action that fails. It then has to wait to get a new IP address that enables the Mobility action. Multihoming SHOULD NOT be used for a Security Association that is not in an active state. In fact implementations might refuse to add an IP address to an non-active SA. This is beyond the scope of IPsec management, since IPsec databases are independent of the transport layer, and thus does not have any state regarding to the transport layer. Nevertheless, with Multihoming and Mobility, we believe mobility / multihoming / IPsec management will occur at the same level, and thus IPsec will be handled in conjunction of connections states.
  - Mobility is very Initiator-centric. The Initiator informs the Responder that an IP address is no more in use and another one MUST be used instead. Multihoming can be used to check an IP address matches a connection without affecting the current connection, it can also provide an IP address that MAY be used by the Responder. The Initiator can send a set of IP addresses to inform the Responder that in case the current connection fails. This is the purpose of Alternate IP Addresses. The Responder MUST only use them when at least one of the current IP address does not work. The Responder can then choose which IP address it prefers to use. How the Responder choose the IP

addresses to reach the Initiator is beyond the scope of this document, but the decision can be based on Responders local policies as well as information provided by the Initiator on the IP addresses (like preferences for example.).



## [9.](#) Related work

This section is not normative and only stands here for clarification.

### [9.1.](#) IPsec

[RFC4301] describes the IPsec architecture and the three associated databases : the Security Association Database (SAD), the Security Policy Database (SPD) et the Peer Authorization Database (PAD).

[Section 5.1](#) describes how outbound packets MUST be processed. For any outbound packet, a SPD lookup occurs first, then a SPD-Cache lookup is performed. The SPD-Cache defines whereas the packet should be BYPASS, DISCARD or PROTECT with the associated index of the SAD. The packet is then sent to a forwarding function that sends the packet on the wire or performs a loop back to the protected interface in the case of nested SA.

[Section 5.2](#) describes how inbound packets are handled. If packet are not IPsec protected, then a SPD-I lookup is performed and the SPD-I defines whether the packet should be DISCARD or BYPASS. If the inbound packet seems to be IPsec protected with protocol AH or ESP, then an SAD lookup is performed. The ESP/ AH process is done according to the SAD entry. Once performed, traffic selector MUST match the packet header's. If no match is found, then the packet MUST be discarded. After ESP/AH process, checking the traffic selectors can be performed by a SPD lookup. [[RFC4301](#)] mentions on p.61.

"This processing includes using the packet's SPI, etc., to look up the SA in the SAD, which forms a cache of the SPD for inbound packets (except for cases noted in Sections [4.4.2](#) and 5)."

The SA contains all security material to perform the IPsec processing. The SPD lookup is required to check that SAD are coherent with the SPD, that defines the Security Policy of the system, and can be changed.

SAD lookup is defined in [section 4.1](#) and MUST consider the longest match. The lookup considers then a match with the SPI, source and destination address, if no match occurs then a match for SPI and

destination address is considered. If no match occurs, then only the SPI match is considered.

[Section 4.4.3](#) provides a description and interaction between the PAD and other IPsec databases. The PAD is involved during the IKE\_AUTH exchange, and provides instruction on which IKE\_ID have to be considered and how IKE\_ID should be authenticated. During the CREATE\_CHILD\_SA, the peers are authenticated but the PAD provides instruction on how the SPD should be lookup, that is to say either considering the IKE\_ID or the Traffic Selector as an entry to the SPD. An interesting thread provides clarification on the PAD <http://www.nabble.com/PAD-and-IKEv2-td13123521.html#a13130457>

"The PAD is an artifact of the description of the processing model, but implementations will need something like it, because the SPD by itself does not provide enough information to IKE (one possible implementation might be to augment the SPD with data that would belong in the PAD in the nominal model). The PAD does two crucial things: it describes how to authenticate peers, and it specifies constraints on the traffic selectors that peers will be allowed in their child SA proposals. --Nico"

"The PAD gives a mapping/relation/binding between certain pieces of information. It's a local matter how this mapping/relation/binding is realized. I'm aware of at least one implementation, where the PAD is implemented as a table/database. -- Christian"

[Section 6](#) provides a description on how ICMP interacts with IPsec. When ICMP message are not protected, it is recommended for network administration purpose to accept and response to them.

In our case, during mobility or multihoming action, a Security Association is derived from an existing Security Association. We MUST first check with the PAD is the selector can be used by the ID, then check what Security Policy of the system requires for this new IP address associated to this Identity. If the current SA matches the security policy, then the new SA can be derived. Otherwise, a new SA MAY be negotiated.

## [9.2.](#) IKEv2

[RFC4306] defines IKEv2 and [\[RFC4718\]](#) provides clarification mainly for implementation purposes. When an IKE negotiation is initiated, it starts with an IKE\_INIT exchange. The IKE\_INIT exchange aims at defining a secure channel for IKE negotiation and management of SA. All actions such as creating a child SA, rekeying an IKE SA, rekeying child SA is performed through the CREATE\_CHILD\_SA exchange. The

two exchanges : the IKE\_SA\_INIT that establish an IKE\_SA, and the IKE\_AUTH which authenticates the peers. The IKE\_AUTH exchange is also used to create the first CHILD\_SA. This is mainly to avoid another exchange that would introduce more network latency. When an SA is created either via the IKE\_AUTH or CREATE\_IKE\_SA exchange, Traffic Selectors (TS) are specified. Such selectors are used to match the SPD to negotiate and create the SA(s). TS range can be narrowed by the Responder. The Responder can also accept for the given SA some subsets of the TSi. There are some situations where two different SA SHOULD be created rather than a single SA. In that case, the Responder should send a NOTIFY message with ADDITIONAL\_TS\_POSSIBLE. It indicates that additional selectors would be accepted but would require a separate SA ([section 2.9](#) and [section 3.10.1 RFC4306](#)).

COOKIES exchange can be used as a way to test return routability verification. COOKIES are sent in Notify Payloads. Routing verification can be done at two different levels. COOKIE can check the IKE\_SA is still valid, and the host is still reachable with the new IP address. When the peer changes its IP address, a successful COOKIE exchange means peers are still reachable and the IKE\_SA is still valid. If ICMP should not be used to check reachability, combination of the two tests can lead to the following conclusion : peers are not IP reachable, peers have no more valid IKE\_SA, peers are IP reachable with valid IKE\_SA channel. Peer can have no more valid IKE\_SA channel when for example, the new network filters IKE traffic.

CREATE\_CHILD\_SA exchange is used to rekey an existing SA, rekey an IKE\_SA, rekey a CHILD\_SA, or create new CHILD\_SA. As specified in [section 2.8 of \[RFC4306\]](#), CREATE\_CHILD\_SA exchange is optional and implementations MUST NOT support this exchange. [\[RFC4718\]](#) describes the exchanged payloads in the following cases :

We provide the exchanges only for illustration purposes, and complete description are provided in [\[RFC4718\]](#).

Initiator  
-----

Responder  
-----

```

HDR, SK {SA, Ni, [KEi]} -->
<-- HDR, SK {SA, Nr, [KEr]}

Rekeying IKE_SA

```

Migault

Expires March 5, 2010

[Page 17]

Internet-Draft

IPsec with mobility and multihoming

Sep 2009

```

Initiator                               Responder
-----
HDR, SK {N(REKEY_SA), [N+], SA,
Ni, [KEi], TSi, TSr} -->
<-- HDR, SK {[N+], SA, Nr,
[KEr], TSi, TSr}

```

Rekeying CHILD\_SA

```

Initiator                               Responder
-----
HDR, SK {[N+], SA, Ni, [KEi],
TSi, TSr} -->
<-- HDR, SK {[N+], SA, Nr,
[KEr], TSi, TSr}

```

Creating New CHILD\_SA

[Section 3.11 in \[RFC4306\]](#) describes a DELETE payload that enables to delete SA. SA are identified by their SPI.

Reauthentication without generating a new CHILD\_SA is described in [\[RFC4478\]](#).

IKEv2 is the negotiation protocol peers use to setup Security Associations. This application needs to check coherence between the IPsec databases. The negotiation protocol is designed in a query / response manner, and it allows extensions such as MOBIKE [\[RFC4555\]](#).

### [9.3.](#) MOBIKE

MOBIKE is defined in [\[RFC4555\]](#). It provides an extension of IKEv2

for mobility and multihoming. MOBIKE considers mobility and multihoming in one specific scenario : peer is connected to its home network using a IPsec tunnel mode, the peer is using only one interface, and the peer changes the address of the tunnel. MOBIKE also considers some cases of NAT, and it is recommended to run MOBIKE on port 4555.

The mobility initiative is performed by the client. This is the only case the peer can try to reach the other peer without being notified of any mobility action, is when the peers cannot reach each other. Since only one IP address is used at a time, this active IP address is always the one in the IKEv2 message header.

The main messages involved in MOBIKE are the MOBIKE\_SUPPORTED Notify message to indicate the peer supports MOBIKE. The

ADDITIONAL\_IP4\_ADDRESS and ADDITIONAL\_IP6\_ADDRESS Notify message enable an Initiator to provide the Responder the IP addresses that he might be used. The Responder does not have the ability to choose which IP address is going to be used to reach the Initiator except if the Initiator cannot be reached with its former IP address. In that specific case, the Responder can try addresses from the list. By providing different IP addresses, the Responder has the possibility to choose which IP address fits best its local policies to reach the Initiator.

On the other hand, the Initiator can also get a list of IP address and choose which one best fits its local policies. The mechanisms used to select the IP addresses are beyond the scope of MOBIKE, but MOBIKE provides means for the Initiator to redirect the traffic to another IP address. Mobility is performed with the UPDATE\_SA\_ADDRESSES Notify payload. Since only one IP address is used at a time, the address to be considered is the one inside the IKEv2 IP header. There is no need to provision the IP address before performing the mobility. MOBIKE also provides a COOKIE2 Notify payload that provides return routability check. Whether this check should be performed or not and when it is defined by local policies.

#### [9.4.](#) MIPv6

MIPv6 is described in [[RFC3775](#)] or in [[I-D.ietf-mext-rfc3775bis](#)], and the tunneling technique is specified in [[RFC2473](#)]. Its main purpose

is to provide a permanent IP address, which is usually hosted in the DNS : the Home of Address (HoA). When the Mobile Node (MN) cannot have its HoA as its active IP address, it uses a Care of Address (CoA) and sets up a tunnel between the MN and the Home Agent (HA) with the CoA and the IP address of the HA. This tunnel enables packets with HoA as IP destination to be routed to the HA and tunneled to the MN. In return, the HA routes packets with HoA as a source address to the CN. Mobile IP requires IPsec to secure the messages between the MN and the HA. IPsec and MIPv6 is specified in [\[RFC3776\]](#) and in [\[RFC4877\]](#). The Security Association are negotiated between the HoA and the HA. Signaling messages with MIPv6 are identified by the protocol selector. To match the SAD, a special action MUST be performed for inbound and outbound Binding Update (BU) packets : lookup in SPD and SAD MUST be done by replacing the CoA by the HoA, which is mentioned in the routing header extension. This is to make the SA independent of the CoA. On the other hand the IKE\_SA is negotiated with the CoA. When changing CoA, one can indicate with the K bit, that the used IP address in the header is a new CoA and the IKE\_SA MUST be updated.

Routing optimization is described in [\[RFC3775\]](#), and enables message to be directly routed to the CN rather than going through the HA.

#### [9.5.](#) MIPv6 and MOBIKE

The purpose of MIPv6 is to provide the MN a permanent IP address. A tunnel is created between the MN and the HoA. When the CoA changes, a Binding Update (BU) with a K bit asks the Tunnel to be updated. There are different flows to consider : The tunnel between the CoA and the HA, the MIPv6 signaling channel, and the IPsec signaling channel. The tunnel between the CoA and the HA might be secured with IPsec, but this is not required. The MIPv6 signaling channel is secured with IPsec in transport mode. The associated Security Associations are based on the HoA and the HA IP address. The IPsec signaling channel (IKEv2) is secured by the IKE\_SA. This is the value that is updated with the K bit. The new IP address to be used is provided by the BU message.

MOBIKE uses the UPDATE\_SA\_ADDRESSES message as an update request. The new IP address to consider is found in the IKE message. Since MOBIKE works only in tunnel mode, updating the SA requires changing not its selector, but one parameter of the SA, that is to say the

outer IP address of the tunnel. The IKE\_SA is also the SA that needs to be updated by changing the traffic selectors. The UPDATE\_SA\_ADDRESSES is sent with the new IP address, which means even with a new unknown IP address the packet will be analyzed.

#### [9.6.](#) SHIM6

To be done.

#### [9.7.](#) SCTP

To be done.

#### [9.8.](#) mtcp

To be done.

#### [9.9.](#) HIP

Host Identity Protocol (HIP) is defined in [[RFC4423](#)] and [[RFC5533](#)]. The mobility and multihoming extension is described in [[RFC5206](#)]. HIP is taking advantage of the separation of Host Identifier (HI) and the locator (LOC). This means that the application are presented a Host Identity Tag (HIT) which is independent from the IP addresses. HIP requires an ESP Security Association between the HITs, and [[RFC5202](#)] describes how the ESP association is established and maintained. Security Association are negotiated between HIs or HITs, packets are sent using IP addresses that are respectively bound to HITs. [[RFC5202](#)] provides two examples on how the ESP processing

of HIP packet can rely on standards compliant IPsec implementations.

- Transport : The processing is represented by figure "ESP processing : Transport". In that case, the HIP layer maintains the SAD and SPD with IP addresses that are associated with the different HITs. The HIP layer proceeds to replacement of HITs by IP addresses.
- Tunnel : The ESP processing is representing by figure "ESP processing : Tunnel". The tunnel mode refers to the BEET mode described in [[I-D.nikander-esp-beet-mode](#)]. It works like in the tunnel mode except that the inner header is removed.

```

+-----+ +
| Application | | HIT_s - HIT_d
+-----+ o
      |      u
+-----+ t
|   ULP   | b HIT_s - HIT_d
+-----+ o checksum(HIT)
      |      u
+-----+ n HIT_s -> IP_s   HIP <-> {IP1, IPi, IPn}
|   HIP   | d HIT_d -> IP_d
+-----+ |
      |      p SAD, SPD based on IP addresses
+-----+ a
|   IPsec  | c
+-----+ k ESP processing
      |      e
+-----+ t send IP packet
|   IP    | | on wire
+-----+ v

```

ESP processing : Transport

```

+-----+ +
| Application | | HIT_s - HIT_d
+-----+ o
      |      u
+-----+ t

```

^  
|  
|  
i  
n



	ULP		b	HIT_s - HIT_d	b
+-----+			o	checksum(HIT)	o
			u		u
+-----+			n		n
	IPsec		d	SAD SPD based on HITs	d Check IP, tunnel header
				ESP processing	ESP processing
	BEET mode		p	HIT_s -> IP_s	p IP_s -> HIT_s
			a	HIT_d -> IP_d	a IP_d -> HIT_d
			c		c
+-----+			k		k
			e		e
+-----+			t	send IP packet	t
	IP			on wire	
+-----+			v		

#### ESP processing : Tunnel

HIP provides mobility and multihoming facilities. Security association are negotiated between HIT, and the HIP layer binds the HIT with the corresponding IP address table and more specifically updates the corresponding SAD and SPD.

HIP Provides the transport mode facilities MOBIKE is missing. In fact a HIP communication involves a IPsec transport mode, and signaling messages to add or change IP addresses. Nevertheless, the goal of this paper is to provide IPsec facilities for Multihoming and mobility so that an host can rely on an IPsec security. This means that the IPsec layer should support any possible IPsec configuration due to mobility and multihoming scenarios. More precisely, we expect that mobility and multihoming can be performed by an IPsec host using the IPsec tunnel mode or the IPsec transport mode. Furthermore we would like IPsec to keep the IP granularity. This means that a multihomed host can have different Security Association depending on the path. In other words, we are looking for finer granularity than HIP, and Security Association should still be negotiated by peers on a per "selector" base. With HIP Security Associations are negotiated between HITs. Although next versions of HIP might supports those facilities, we are looking to extend the already existing mobility and multihoming IKEv2 extension. Developments SHOULD consider providing a neat API that might be used by other Upper Layer Protocols

## 10. Mobility / Multihoming with IKEv2/IPsec mechanisms

This section describes how mobility and multihoming can be considered with IPsec and IKEv2 mechanisms, how they differ from mechanisms exposed in the Scenarios description, as well as how those differences impact mobility and multihoming. In this section we only consider transport mode between the Initiator and the Responder.

### 10.1. Possible Mobility Scenarios

#### 10.1.1. Analyze

The move and reconnect scenario consists of the Initiator changing its IP address, creating the corresponding Security Association, and deleting the previous one.

[Section 1.3](#), 1.17 and 1.18 of [\[RFC4306\]](#) provide descriptions of rekeying exchanges. Different configurations include creation of a new SA, rekeying an already existing SA and rekeying an IKE\_SA. In our case, the Initiator wants to notify the IP address has changed. We do not rekey an already existing SA since the traffic selectors are changed. The required action is to create a new CHILD\_SA with the corresponding traffic selectors. The Initiator rekey request MUST omit the REKEY\_SA Notify Payloads that identifies the SA to be rekeyed. Once the new SA has been created, then the Initiator MUST delete the old CHILD\_SA, which is done thanks to the Delete Notify Payload described in [section 3.11 of \[RFC4555\]](#).

Current IPsec / IKEv2 do not provide any mechanisms so that the Initiator can check if an IP address matches SPD and local policies of the peer. Reachability tests can be provided through the Return Routability Check mechanism of MOBIKE described in [\[RFC4555\]](#), but the test requires that the tested IP address is used to send the Notify Payload. In other words, this mechanism can only be useful if the Initiator is Multihomed.

As a conclusion :

For a single homed Initiator : The only scenario to be considered is the "Move and Reconnect". The Initiator proceeds to a CREATE\_CHILD\_SA, followed by a DELETE exchange. This means that the mobility action requires two message exchanges to be performed.

For a multi homed Initiator : The "Move and Reconnect" scenario can be considered similarly as for a singled homed Initiator. The multihomed Initiator can proceed to the "Preconnect and Move" scenario, in the sense that it can check reachability with the COOKIE2 mechanism.

### 10.1.2. Requirements

Requirements on the Mobility and Multihoming extension are :

- The Initiator MUST be able to UPDATE a Security Association, that is to say, change the IP address of an SA.
- The Initiator MUST be able to check before the mobility action is performed whether or not the new IP address matches the SPD or the local policies. This means the tested IP address for the mobility can be specified as an argument and MUST NOT be in the Initiator IP address of the IP header.
- When a Responder receives a request to check the validity of a mobility action it MUST be able to provide the Initiator relevant pieces of information why the IP address is not accepted. Possible reasons might be the IP address does not match the local policies, or the SPD.
- Mobility MUST be performed with a single message exchange.

### 10.2. Possible Multihoming Scenarios

At the current time, multihoming is only considered with MOBIKE [[RFC4555](#)]. Multihoming is considered for the IKEv2 application, which means that IKEv2 supports multihoming. Furthermore IKEv2 supports multihoming means that the Initiator provides the Responder a range of IP address the Responder might use if the current IP address happens to be unreachable.

This section does not deal on how IKEv2 deals with multihoming. This paper exposes how IKEv2 can deal with multihoming of Security Association. More specifically, how an Initiator can assigned one or more IP address to a given Security Association, and how can it removes one or more IP address of a Security Association.

#### 10.2.1. Analyze

As in the "Possible Mobility Scenarios", current IPsec suite mechanisms provides the Initiator the ability to create new independent Security Association. With Multihoming this would mean that the number IKE context would be extremely high. If the Initiator has  $m$  IP addresses, and the Responder  $n$  IP addresses, this would lead to the creating of  $2mn$  CHILD\_SA. With the possibility to add and remove IP address to an already existing Security

Association, the number of IKEv2 CHILD\_SA is 2.

As mentioned in the definition section of multihoming, ADD or REMOVE an IP address is only one aspect of Multihoming. The other aspect is to provide alternate IP address to a given Security Association. This means that the Initiator SHOULD be able to inform the Responder

of alternate IP address that should be used if the Initiator is not reachable with the current IP addresses.

#### [10.2.2.](#) Requirements

Requirements on the Mobility and Multihoming extension are :

- The Initiator MUST be able to ADD an IP address to a given Security Association.
- The Initiator MUST be able to REMOVE an IP address to a given Security Association.
- For REMOVE or ADD action the Initiator MUST be able to check if the action is possible on the Responder side. This means that Responder MUST be able to provides information whether or not the action can be performed or not. If the action cannot be performed, then, the Responder MUST send information such as the IP address does not match the SPD or the local policies.
- The Initiator MUST be able to provide alternate IP addresses for a given Security Association.

#### [10.3.](#) MOBIKE and our Scenarios

MOBIKE is already designed to deal with mobility and multihoming. The purpose of this section is to list the functionalities that MOBIKE already has and those that MOBIKE is missing. This section points out MOBIKE properties and how there should be expanded to match our requirements

##### [10.3.1.](#) Analyze

- Transport mode : Mobility extension for transport mode requires changes between IKEv2 and IPsec. MOBIKE is already proposing one solution for tunnel mode mobility. The main difference with transport and tunnel mode is that in the tunnel mode the SPD is defined according to the inner IP addresses, when the

outer addresses are changed, only the SAD is changed. In the transport mode, SAD AND SPD MUST be changed. The PAD MUST also be checked to see if new SA can be created.

- Multiple addresses : MOBIKE considers only one IP address can be used at a time. The peer has established an IKE\_SA to secure IKE transactions. Sending an UPDATE\_SA\_ADDRESSES requires some checks, and then creating new SAD entries, checking Return Routability before deleting the old ones. Since peers are using only one IP addresses, mobility parameters are not explicitly mentioned. In fact, the address to be replaced is the one the Initiator used to have to carry the tunneled packets. The replacing IP address is the address located in the IP header of the IKE message. With a multihomed Initiator,

the Initiator MUST have the ability to explicitly specify the new IP address and the IP address to be changed. The Initiator MUST also have ways to specify which SA or IKE\_SA the change applies.

- Alternate IP addresses : MOBIKE already provides a mechanism to provide alternate IP addresses to the Responder. Those alternate IP addresses only concerns the IKEv2 application, and does not apply to the CHILD\_SAs. The Initiator SHOULD be able to specify alternate IP addresses to the different Security Associations. The main motivation for such functionalities is to provide IKEv2 the ability to communicate any information related to mobility or multihoming to Upper Layer Protocols.
- NAT : MOBIKE deals with NAT, and we would like to take advantage of this work, even though we do not consider NAT.
- Who decides the mobility or multihoming action : MOBIKE actions concerns only the client IP addresses. This means that the owner of the different IP address can choose to proceed to change the IP address. The only situation that excepts this rule is when a peer is not reachable, in that case, the other peer may try to use alternate IP address provided previously by the peer. We keep using this philosophy, the concerned peer should be the only one that should proceed to a mobility action.
- The IP the be used : MOBIKE is using only one single IP address at a time. When sending an UPDATE\_SA\_ADDRESSES, MOBIKE does not specify the address to be used. The considered address is the one provided in the IKE packet header. As explained in the multiple address bullet, this rule cannot be applied anymore

when more than one IP address is being used.

### [10.3.2.](#) Requirements

Requirements on the Mobility and Multihoming extension are :

- Mobility and Multihoming MUST be done with IPsec in transport and in tunnel mode.
- Mobility and Multihoming action, i.e. UPDATE, REMOVE, ADD action can explicitly specifies which IP address is to be ADDED, REMOVED, UPADTEd, and which IP address MUST replaced. Replacing IP address SHOULD NOT be necessary the one in the IP header, and the replaced IP address cannot be the "one" previously used by the Initiator.
- The Initiator MUST be able to provide alternate IP addresses associated to specific SA

Migault

Expires March 5, 2010

[Page 26]

---

Internet-Draft

IPsec with mobility and multihoming

Sep 2009

- Ways MUST be provided to the Initiator so that it can explicitly specifies the SA or IKE\_SA the action applies to. The IKE\_SA and CHILD\_SA is not the only value.

## [11.](#) Mobility Rejected by the Responder?

This section describes when a Mobility request might be rejected. When do the new IP address might be rejected by the local policies or SPD?

When one peer moves from one public network to another public network, we believe in most cases the Security Policy will remain the same on both IP addresses. This means that Security Policies might change when one a peer moves private IP address to a public IP address. This leads to consider the following three cases :

- Case 1 : Both peers are on the private network. Peers have established a connection within a private network, and one peer moves to the public network.
- Case 2 : One peer is in the public network, the other in the

- private network, one peer moves from the public network to the private network.
- Case 3 : One peer is in the public network, the other peer is in the private network and the peer moves from the private to the public network.

For topological reasons, if the peers are both within the private network and one of the peer wants to move and use a public IP address instead, specific mechanisms must be used to enable the reachability between the two hosts. Such mechanisms SHOULD involve a security gateway or a Home Agent. MIPv6 for example provides means not to break the communication and establish a tunnel between the Care of Address and the Home Agent [[RFC3775](#)]. Although MIPv6 enables the connection not to be broken, MIPv6 negotiations are required to set up the tunnel. Using MIPv6 does not change the Security Association between the peers, the Security Association is established between the private IP addresses (eventually HoA). In most cases private networks are considered as trusted network, and no Security Association will be established between the two private IP addresses. The Security Association will be negotiated between the new public IP address and the gateway to setup a secure tunnel. Since aside mechanisms MUST be provided to provide reachability of the peers, this case does not match the Move-and-reconnect scenario as specified in this section.

Case 2 and case 3 suppose that one peer is in a public network, the other peer is in a private network, and a connection is established

between them. Reachability between the two different networks can be provided by NAT, MIPv6, or different proxy / gateway MUST be used. We can then suppose there is an IKEv2 and a communication channel established between those two peers. Case 2 and cases 3 differs from case 1 because the connection between the peers is already established between heterogeneous networks. Case 2 supposes the peer on the public network is moving to the private network. It is highly possible that the Security Association across public and private network will be "more secured" than required Security Association set for communications within the private network. Unless internal network policy for example discards encrypted packets, it is highly possible that the mobility action can occur. After the moving action has occurred, the communication between the peers in the private network will be most probably "over protected", and the renegotiation

of this Security Association might occur. Mobility action is of no use if the communication between the peers within the private network does not require any authentication nor protection at any layer. Even though private networks are considered as trusted networks, most communication should still be protected by security mechanisms such as TLS [[RFC5246](#)], WESP [[I-D.grewal-ipsec-traffic-visibility](#)]... Case 3 considers the other way around, that is to say a peer going from the private network to the public network. In that case local policy on the public new IP address might requires first to upgrade security policies before moving, otherwise the connection will be broken since security policies do not match.

Note that in this example we considered the private and public network as two networks that for a peer might have different Security Policies. We can extend this scenario to any networks A and network B with different security policies.

The mobility action is independent of the IP protocol. Connectivity can use IPv6 or IPv4. With one or the other IP version we should use the proper added mechanism ( MIPv6 [[RFC3775](#)] ).

## 12. Scope and Restrictions

Scenarios described in this paper requires extensions of MOBIKE facilities, and more specifically simultaneous use of IP addresses, and the use of transport mode.

Nevertheless, one should be aware that this does not represent a complete solution for mobility, and identified restrictions are presented below. Some of them are related to the use of transport mode and upper layer protocols.

- Application : Applications that can take advantage of the use of mobility and the transport mode must have been designed for in some ways. First the application should be designed with multiple short and fast query/responses. In that sense heavy download application based on one connection should not be considered. Then the application should be able to easily deal with non working connections. A typical application that



matches those two requirements is a web browsing application. A web browser sends a GET message and receives a HTTP/1.1 200 OK response [[RFC2616](#)]. Browser are also used to handle with non reachable server by selecting randomly the IP address from the DNS response, and that end users are used to retry when it does not work on the first click. Changing IP address while browsing the Internet has very few impacts, page downloading are not requested so continuously, the response should be quite fast, and if by chance, the mobility occurs while downloading the page, the user will re-send its request. In the next future other IP architectures involving 3.5 layers HIP [[RFC4423](#)] , LISP [[I-D.farinacci-lisp](#)], SHIM6 [[RFC5533](#)] might overcome some of the restrictions, by avoiding breaking a connection while changing the IP address...

- Transport : The draft only considers modification of the IPsec and IP layer. In that sense, changes do not consider interactions with the transport layer. This means that TCP connections will be broken when a change of the IP address is occurring. The application is expected to re-initiate a connection. In other words no mechanisms are provided in this draft so to make the mobility transparent to the transport layer.
- Mobility : This scenario does not consider simultaneous mobility from the MN and the CN peer. Full mobility management is the purpose of the MIPv6 [[RFC3775](#)]. This draft considers a one end mobility. The mobility considered in this draft is similar as the one considered in MOBIKE [[RFC4555](#)], except that we provide mobility for transport mode. The mobility considered in this draft, with those restrictions, might be better called "opportunistic mobility" or "nomadism".

### [13.](#) Acknowledgments

Daniel Migault is partly funded by 3MING, a research project supported by the French 'National Research Agency'(ANR). We also thanks for their comments Pierrick Seite, Daniel Palomares and Jean Michel Combes.

## 14. Security Considerations

The whole draft is about security.

## 15. IANA Considerations

There is no IANA consideration here.

## 16. References

### 16.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", [RFC 2473](#), December 1998.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.
- [RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", [RFC 3775](#), June 2004.
- [RFC3776] Arkko, J., Devarapalli, V., and F. Dupont, "Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents", [RFC 3776](#), June 2004.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", [RFC 4301](#), December 2005.
- [RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", [RFC 4306](#), December 2005.
- [RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", [RFC 4555](#), June 2006.
- [RFC4877] Devarapalli, V. and F. Dupont, "Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture", [RFC 4877](#), April 2007.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.
- [RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming

---

Internet-Draft      IPsec with mobility and multihoming

Sep 2009

Shim Protocol for IPv6", [RFC 5533](#), June 2009.

## [16.2](#). Informative References

[I-D.farinacci-lisp]

Farinacci, D., Fuller, V., Meyer, D., and D. Lewis,  
"Locator/ID Separation Protocol (LISP)",  
[draft-farinacci-lisp-12](#) (work in progress), March 2009.

[I-D.grewal-ipsec-traffic-visibility]

Grewal, K., "XESP for Traffic Visibility",  
[draft-grewal-ipsec-traffic-visibility-02](#) (work in  
progress), June 2008.

[I-D.ietf-mext-rfc3775bis]

Johnson, D., Perkins, C., and J. Arkko, "Mobility Support  
in IPv6", [draft-ietf-mext-rfc3775bis-04](#) (work in  
progress), July 2009.

[I-D.nikander-esp-beet-mode]

Nikander, P. and J. Melen, "A Bound End-to-End Tunnel  
(BEET) mode for ESP", [draft-nikander-esp-beet-mode-09](#)  
(work in progress), August 2008.

[RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol  
(HIP) Architecture", [RFC 4423](#), May 2006.

[RFC4478] Nir, Y., "Repeated Authentication in Internet Key Exchange  
(IKEv2) Protocol", [RFC 4478](#), April 2006.

[RFC4718] Eronen, P. and P. Hoffman, "IKEv2 Clarifications and  
Implementation Guidelines", [RFC 4718](#), October 2006.

[RFC5201] Moskowitz, R., Nikander, P., Jokela, P., and T. Henderson,  
"Host Identity Protocol", [RFC 5201](#), April 2008.

[RFC5202] Jokela, P., Moskowitz, R., and P. Nikander, "Using the  
Encapsulating Security Payload (ESP) Transport Format with  
the Host Identity Protocol (HIP)", [RFC 5202](#), April 2008.

[RFC5206] Nikander, P., Henderson, T., Vogt, C., and J. Arkko, "End-  
Host Mobility and Multihoming with the Host Identity  
Protocol", [RFC 5206](#), April 2008.

Internet-Draft

IPsec with mobility and multihoming

Sep 2009

Author's Address

Daniel Migault  
Orange Labs R&D  
38 rue du General Leclerc  
92794 Issy-les-Moulineaux Cedex 9  
France

Phone: +33 1 45 29 60 52  
Email: [mgl.t.ietf@gmail.com](mailto:mgl.t.ietf@gmail.com)

Migault

Expires March 5, 2010

[Page 32]