

November, 2007

**Mobile IPv6 support for dual stack
Hosts and Routers (DSMIPv6)
draft-ietf-mip6-nemo-v4traversal-06.txt**

Status of this memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of 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 cite them other than as "work in progress".

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

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

This document is a submission of the IETF MIP6 WG. Comments should be directed to the IPv6 WG mailing list, mip6@ietf.org.

Abstract

The current Mobile IPv6 and NEMO specifications support IPv6 only. This specification extends those standards to allow the registration of IPv4 addresses and prefixes, respectively, and the transport of both IPv4 and IPv6 packets over the tunnel to the HA. This specification also allows the Mobile Node to roam over both IPv6 and IPv4, including the case where Network Address Translation is present on the path between the mobile node and its home agent.

Table of Contents

1. Introduction.....	2
1.1 Motivation for using Mobile IPv6 only.....	3
1.2 Scenarios considered by this specification.....	4
2. Solution overview.....	5
2.1. Home Agent Address Discovery.....	5
2.2. Mobile Prefix Solicitations and Advertisements.....	6
2.3. Binding management.....	7
2.3.1 Foreign network supports IPv6.....	7
2.3.2 Foreign network supports IPv4 only.....	8
2.3.2.2 Visited network supports IPv4 only (private addresses).....	9
2.4. Route optimization.....	10
2.5. Dynamic IPv4 home address allocation.....	10
3. Extensions and modifications to Mobile IPv6.....	10
3.1. Binding update extensions.....	10
3.1.1 IPv4 home address option.....	10
3.1.2 The IPv4 Care-of Address option.....	11
3.1.3 The Binding Update Message extensions.....	12
3.2. Binding Acknowledgement extensions.....	12
3.2.1 IPv4 address acknowledgement option.....	12
3.2.2 The NAT detection option.....	14
3.2.3 Extensions to the Binding Acknowledgement message.....	14
4. Protocol operation.....	15
4.1. Tunnelling formats.....	15
4.2. NAT detection and traversal.....	16
4.3. NAT Keepalives.....	18
4.4. Mobile node operation.....	19
4.4.1 Sending packets from a visited network.....	20
4.4.2 Movement detection in IPv4-only networks.....	21
4.5. Home agent operations.....	21
4.5.1 Sending packets to the mobile node.....	23
4.6. Correspondent node operations.....	23
5. Security considerations.....	23
5.1 Handover interactions for IPsec and IKE.....	24
6. Protocol constants.....	25
7. Acknowledgements.....	25
8. IANA considerations.....	26
9. References.....	26
10. Contributors.....	27
Author's Address.....	27

[1. Introduction](#)

Mobile IPv6 [[MIPv6](#)] and [[NEMO](#)] allow mobile nodes to move within the Internet while maintaining reachability and ongoing sessions, using an IPv6 home address or prefix. However, since IPv6 is not widely

deployed, it is unlikely that mobile nodes will use IPv6 addresses only for their connections. It is reasonable to assume that mobile nodes will, for a long time, need an IPv4 home address that can be

used by upper layers. It is also reasonable to assume that mobile nodes will move to networks that might not support IPv6 and would therefore need the capability to support an IPv4 Care-of Address. Hence, this specification extends Mobile IPv6 capabilities to allow dual stack mobile nodes to request that their home agent (also dual stacked) tunnel IPv4/IPv6 packets addressed to their home addresses, as well as IPv4/IPv6 care-of address(es).

Using this specification, mobile nodes would only need Mobile IPv6 and [\[NEMO\]](#) to manage mobility while moving within the Internet; hence eliminating the need to run two mobility management protocols simultaneously. This specification provides the extensions needed in order to allow IPv6 mobility only to be used by dual stack mobile nodes.

This specification will also consider cases where a mobile node moves into a private IPv4 network and gets configured with a private IPv4 Care-of Address. In those scenarios, the mobile node needs to be able to traverse the IPv4 NAT in order to communicate with the Home Agent. IPv4 NAT traversal for Mobile IPv6 is presented in this specification.

In this specification, the term mobile node refers to both a mobile host and mobile router unless the discussion is specific to either hosts or routers. Similarly, we use the term home address to reflect an address/prefix format.

In this specification, extensions are defined for the binding update and binding acknowledgement. It should be noted that all these extensions apply to cases where the mobile node communicates with a Mobility Anchor Point (MAP) as defined in [\[HMIPv6\]](#). The requirements on the MAP are identical to those stated for the home agent, although it is unlikely that NAT traversal would be needed with a MAP as it is expected to be in the same address domain.

1.1 Motivation for using Mobile IPv6 only

IPv6 offers a number of improvements over today's IPv4, primarily due to its large address space. Mobile IPv6 offers a number of improvements over Mobile IPv4, mainly due to capabilities inherited from IPv6. For instance, route optimization and Dynamic home agent discovery can only be achieved with Mobile IPv6.

One of the advantages of the large address space provided by IPv6 is that it allows mobile nodes to obtain a globally unique care-of address wherever they are. Hence, there is no need for Network Address Translator (NAT) traversal techniques designed for Mobile IPv4. This allows Mobile IPv6 to be a significantly simpler and more

bandwidth efficient mobility management protocol. At the same time,
during the transition towards IPv6, NAT traversal for existing

private IPv4 networks needs to be considered. This specification introduces NAT traversal for this purpose.

The above benefits make the case for using Mobile IPv6 only for dual stack mobile nodes in order to allow for a long lasting mobility solution and minimize the need to changing the mobility stack due to the introduction of IPv6 within a deployed network.

1.2 Scenarios considered by this specification

In [[SNRIO](#)] several scenarios that illustrate potential incompatibilities for mobile nodes using Mobile IPv6 were discussed. Some of the problems associated with mobility and transition issues were presented in [[MIP-PB](#)]. This specification considers a subset of the scenarios in [[SNRIO](#)], which address all the problems discussed in [[MIP-PB](#)]. The scenarios considered in this specification are listed below.

All of the following scenarios assume that both the mobile node and the Home Agent are IPv4 and IPv6-enabled and that only Mobile IPv6 is used between the mobile node and the Home Agent. We also assume that the Home Agent is always reachable through a globally unique IPv4 address. Finally, it's important to note that the following scenarios are not mutually exclusive.

Scenario 1: IPv4-only foreign network

In this scenario, a mobile node is connected to an IPv4-only foreign network. The mobile node can only configure an IPv4 Care-of Address.

Scenario 2: Mobile node behind a NAT

In this scenario, the mobile node is in a private IPv4 foreign network that has a NAT device connecting it to the Internet. If the Home Agent is located outside the NAT device, the mobile node will need a NAT traversal mechanism to communicate with the Home Agent.

Scenario 3: Home Agent behind a NAT

In this scenario, the communication between the mobile node and the Home Agent is further complicated by the fact that the Home Agent is located within a private IPv4 network. However, in this scenario, we assume that the Home Agent is allocated a globally unique IPv4 address. Such address might not be physically configured on the Home Agent interface. Instead, it is associated with the Home Agent on the NAT device, which allows the Home Agent to be reachable through address or port mapping.

Scenario 4: Use Of IPv4-only applications

In this scenario, the mobile node may be located in an IPv4, IPv6 or a dual network. However, the mobile node might be communicating with an IPv4-only node. In this case, the mobile node would need a stable IPv4 address for its application. The alternative to using an IPv4 address is the use of protocol translators; however, end-to-end communication with IPv4 is preferred to the use of protocol translators.

The mobile node may also be communicating with an IPv4-only application that requires an IPv4 address.

The cases above illustrate the need for a stable IPv4 home address to be allocated to the mobile node. This is done using an IPv4 home address. Since running Mobile IPv4 and Mobile IPv6 simultaneously is problematic (as illustrated in [\[MIP-PB\]](#)), this scenario adds a requirement on Mobile IPv6 to support IPv4 home addresses.

2. Solution overview

In order to allow Mobile IPv6 to be used by dual stack mobile nodes, the following needs to be done:

- Mobile nodes should be able to use an IPv4 and IPv6 home or care-of address simultaneously and update their home agents accordingly.
- Mobile nodes need to be able to know the IPv4 address of the home agent as well as its IPv6 address. There is no need for IPv4 prefix discovery however.
- Mobile nodes need to be able to detect the presence of a NAT device and traverse such device in order to communicate with the Home Agent in a secure manner.

This section presents an overview of the extensions required in order to allow mobile nodes to use Mobile IPv6 only for IP mobility management.

2.1. Home Agent Address Discovery

Dynamic Home Agent Address Discovery (DHAAD) was defined in [\[MIPv6\]](#) to allow mobile nodes to discover their home agents by appending a well-known anycast interface identifier to their home link's prefix. However, this mechanism is based on IPv6-anycast routing. If a mobile node is located in an IPv4-only foreign network, it cannot rely on native IPv6 routing. In this scenario, the solution for discovering the home agent's IPv4 address is through the Domain Name System (DNS). If the MN is attached to an IPv6-only or dual stack network,

it may also use procedures defined in [[INTBOOT](#)] to discover Home Agent information. Note that the use of [[INTBOOT](#)] cannot give the

mobile node information that allows it to continue to communicate with the Home Agent if, for example, the mobile node moved from an IPv6-enabled network to an IPv4-only network. In such scenario, the mobile node would need to discover the IPv4 address of its home agent through the DNS.

For DNS lookup by name, the mobile node should be configured with the name of the home agent. When the mobile node needs to discover a home agent, it sends a DNS request with QNAME set to the configured name. An example is "ha1.example.com". If a home agent has an IPv4 and IPv6 address, the corresponding DNS record should be configured with both 'AAAA' and 'A' records. Accordingly the DNS reply will contain 'AAAA' and 'A' records.

For DNS lookup by service, the SRV record defined in [BOOT] is reused. For instance, if the service name is "mip6ha" and the protocol name is "ipv6" for the SRV record, the mobile node SHOULD send a DNS request with the QNAME set to "mip6ha.ipv6.example.com". The response should contain the home agent's FQDN(s) and may have the corresponding 'AAAA' and 'A' records enclosed as well.

If multiple home agents reside on the home link, each configured with a public IPv4 addresses, then the operation above applies. In case the home agents are behind a NAT box, there are two options, 1) configure a public IPv4 address for each home agent on the NAT box, 2) configure one public address and make the home agents share the public address. In either case, the correct DNS entries can be configured. Another possible solution is to designate one home agent on the home link for v4 traversal. The NAT device should associate that home agent with the public IPv4 address configured on it for v4 traversal. In all cases above, both the 'AAAA' and 'A' records returned for a particular name MUST correspond to the same physical home agent; otherwise the mobile node will not be able to bind its addresses correctly.

2.2. Mobile Prefix Solicitations and Advertisements

According to [MIPv6], the mobile node can send a Mobile Prefix Solicitation and receive a Mobile Prefix Advertisement containing all prefixes advertised on the home link.

A dual stack mobile node MAY send a Mobile Prefix Solicitation message encapsulated in IPv4 (i.e. IPv6 in IPv4) in the case where the mobile node has no access to IPv6 within the local network. Securing such messages would require the mobile node to have security association with the home agent, using IPsec (AH or ESP) and based on the mobile node's IPv4 care-of address as described in [MIPv6]. Since the mobile node needs to encapsulate all IPv6 traffic sent to the

home agent into IPv4 while located in an IPv4-only visited network, such SA would affect all packets if the selectors were based on the information in the outer header. That is, the SA selectors being the

protocol number (protocol is always IP in IP), as well as, source and destination addresses are all common to all packets. If this effect is not desired, the mobile node can base the SA on the information in the inner header (i.e. using the home agent's IPv6 address, the mobile node's home address and the ICMP protocol number). Such security association would use transport mode ESP protection.

2.3. Binding management

A dual stack mobile node will need to update its home agent with its care-of address. If a mobile node has an IPv4 and an IPv6 home address it will need to create a binding cache entry for each address. The format of the IP packet carrying the binding update and acknowledgement messages will vary depending on whether the mobile node has access to IPv6 in the visited network. There are three different scenarios to consider with respect to the visited network:

- A. The visited network has IPv6 connectivity and provides the mobile node with a care-of address (in a stateful or stateless manner).
- B. The mobile node can only configure a globally unique IPv4 address in the visited network.
- C. The mobile node can only configure a private IPv4 address in the visited network.

2.3.1 Foreign network supports IPv6

In this case, the mobile node is able to configure a globally unique IPv6 address. The mobile node will send a binding update to the IPv6 address of its home agent, as defined in [\[MIPv6\]](#). The binding update MAY include the IPv4 home address option introduced in this document. After receiving the binding update, the home agent creates two binding cache entries, one for the mobile node's IPv4 home address, and another for the mobile node's IPv6 home address. Both entries will point to the mobile node's IPv6 care-of address. Hence, whenever a packet is addressed to the mobile node's IPv4 or IPv6 home addresses, it will be tunneled in IPv6 to the mobile node's IPv6 care-of address included in the binding update. Effectively, the mobile node establishes two different tunnels, one for its IPv4 traffic (IPv4 in IPv6) and one for its IPv6 traffic (IPv6 in IPv6) with a single binding update. The security implications of this mechanism are discussed in the security considerations section.

In this scenario, the only addition to [\[MIPv6\]](#) is the inclusion of the IPv4 home address option in the binding update message.

After accepting the binding update and creating the corresponding binding cache entries, the home agent MUST send a binding acknowledgement to the mobile node as defined in [\[MIPv6\]](#). In

addition, if the binding update included an IPv4 home address option,
the binding acknowledgement MUST include the IPv4 address

acknowledgment option as described later in this specification. This option informs the mobile node whether the binding was accepted for the IPv4 home address. If this option is not included in the binding acknowledgement and the IPv4 home address option was included in the binding update, the mobile node **MUST** assume that the home agent does not support the IPv4 home address option and therefore **SHOULD NOT** include the option in future binding updates to that home agent address.

The routing header in the binding update **MUST** include the mobile node's IPv6 home address as specified in [[MIPv6](#)].

When a mobile node acquires both IPv4 and IPv6 care-of addresses at foreign network, it **SHOULD** prioritize IPv6 care-of address for MIPv6 binding registration. The mobile node **MUST NOT** register both IPv4 and IPv6 care-of addresses to its home agent.

2.3.2 Foreign network supports IPv4 only

If the mobile node is in a foreign network that only supports IPv4, it needs to detect whether a NAT is in its communication path to the home agent. This is done while exchanging the binding update and acknowledgement messages as shown later in this document. If no NAT is detected between the mobile node and the home agent, the mobile node assumes that it is in a foreign network that supports IPv4 public addresses. Otherwise, the mobile node assumes that private addresses are used in the foreign network. Note that this assumption is only valid for the purposes of the signaling presented in this specification. A mobile node **SHOULD NOT** assume that its IPv4 address is globally unique if a NAT device was not detected. The operations of both cases are discussed below.

2.3.2.1 Foreign network supports IPv4 only (public addresses)

In this scenario the mobile node will need to tunnel IPv6 packets containing the binding update to the home agent's IPv4 address. The mobile node uses the IPv4 address it gets from the foreign network as a source address in the outer header. The binding update will contain the mobile node's IPv6 home address. However, since the care-of address in this scenario is the mobile node's IPv4 address, the mobile node **MUST** include its IPv4 care-of address in the IPv6 packet. The IPv4 address is represented in the IPv4 Care-of address option defined in this specification. If the mobile node had an IPv4 home address, it **MUST** also include the IPv4 home address option described in this specification.

After accepting the binding update, the home agent **MUST** create a new binding cache entry for the mobile node's IPv6 home address. If an

IPv4 home address option were included, the home agent MUST create another entry for that address. All entries MUST point to the mobile node's IPv4 care-of address. Hence, all packets addressed to the

mobile node's home address(es) (IPv4 or IPv6) will be encapsulated in an IPv4 header that includes the home agent's IPv4 address in the source address field and the mobile node's IPv4 care-of address in the destination address field.

After accepting the binding updates and creating the corresponding entries, the home agent MUST send a binding acknowledgement as specified in [MIPv6]. In addition, if the binding update included an IPv4 home address option, the binding acknowledgement MUST include the IPv4 address acknowledgment option as described later in this specification. The binding acknowledgement is encapsulated to the IPv4 care-of address.

2.3.2.2 Visited network supports IPv4 only (private addresses)

In this scenario the mobile node will need to tunnel IPv6 packets containing the binding update to the home agent's IPv4 address. In order to traverse the NAT device, IPv6 packets are tunneled UDP and IPv4. The UDP port allocated for the home agent is TBD.

The mobile node uses the IPv4 address it gets from the visited network as a source address in the IPv4 header. The binding update will contain the mobile node's IPv6 home address.

After accepting the binding update, the home agent MUST create a new binding cache entry for the mobile node's IPv6 home address. If an IPv4 home address option were included, the home agent MUST create another entry for that address. All entries MUST point to the mobile node's IPv4 care-of address included in the binding update message. In addition, the tunnel used MUST indicate UDP encapsulation for NAT traversal. Hence, all packets addressed to the mobile node's home address(es) (IPv4 or IPv6) will be encapsulated in UDP then encapsulated in an IPv4 header that includes the home agent's IPv4 address in the source address field and the mobile node's IPv4 care-of address in the destination address field.

After accepting the binding updates and creating the corresponding entries, the home agent MUST send a binding acknowledgement as specified in [MIPv6]. In addition, if the binding update included an IPv4 home address option, the binding acknowledgement MUST include the IPv4 address acknowledgment option as described later in this specification. The binding acknowledgement is encapsulated in UDP then IPv4 with the home agent's IPv4 address in the source address field and the mobile node's IPv4 care-of address in the destination field. The IPv4 address in the destination field of the IP v4 packet is the source address received in the IPv4 header containing the binding update message. The inner IPv6 packet will contain the home agent's IPv6 address as a source address and the mobile node's IPv6

home address in the destination address field.

0		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
+		+		+		+		+		+		+		+		+		+		+	
		Type				Length				Pref				P				Reserved			
+		+		+		+		+		+		+		+		+		+		+	
		IPv4 home address																			

Type	TBD
Length	6
Pref	The length of the prefix allocated to the mobile node. If only a single address is allocated, this field MUST be set to 32. In the first binding update requesting a prefix, the field contains the prefix length requested. However, in the following binding updates, this field must contain the length of the prefix allocated. A value of zero is invalid and MUST be considered an error.
P	A flag indicating, when set, that the mobile node requests a mobile network prefix. This flag is only relevant for new requests, and must be ignored for binding refreshes.
Reserved	This field is reserved for future use. It MUST be set to zero by the sender and ignored by the receiver.
IPv4 home address	The mobile node's IPv4 home address that should be defended by the home agent. This field could contain any unicast IPv4 address (public or private) that was assigned to the mobile node. The value 0.0.0.0 is used to request an IPv4 home address from the home agent. A mobile node may choose to use this option to request a prefix by setting the address to the All Zeroes and setting the P flag. The mobile node could then form an IPv4 home address based on the allocated prefix. Alternatively, the mobile node may use two different options, one for requesting an address (Static or Dynamic) and another for requesting a prefix.

3.1.2 The IPv4 Care-of Address option

This option is included in the Mobility Header including the binding update message sent from the mobile node to a home agent or Mobility Anchor Point. The alignment requirement for this option is 4n.

This option is included in the Mobility Header including the binding

acknowledgement message sent from the home agent or Mobility Anchor Point to the mobile node. This option indicates whether a binding

Type	TBD
Length	6
Status	Indicates success or failure for the IPv4 home address binding. Values from 0 to 127 indicate success. Higher values indicate failure. The following values are reserved: 0 Success 128 Failure, reason unspecified 129 Administratively prohibited 130 Incorrect IPv4 home address 131 Invalid IPv4 address 132 Dynamic IPv4 home address assignment not available 133 Prefix allocation unauthorized
Pref	The prefix length of the address allocated. This field is only valid in case of success and MUST be set to zero and ignored in case of failure. This field overrides what the mobile node requested (if not equal to the requested length).
Res	This field is reserved for future use. It MUST be set to zero by the sender and ignored by the receiver.
IPv4 home address	The IPv4 home address that the home agent will use in the binding cache entry. This could be a public or private address. This field MUST contain the mobile node's IPv4 home address. If the address were dynamically allocated the home agent will add the address to inform the

mobile node. Otherwise, if the address were statically allocated to the mobile node, the

home agent will copy it from the binding update message.

3.2.2 The NAT detection option

This option is sent from the home agent to the mobile node to indicate whether a NAT was in the path. This option MAY also include a suggested NAT binding refresh time for the mobile node. The alignment requirement for this option is 4n. If a NAT is detected, this option MUST be sent by the home agent.

```

      0               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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type   |   Length   |F|           Reserved           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     Refresh time          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type TBD

Length 6

F This flag indicates to the mobile node that UDP encapsulation is required. When set, this flag indicates that the mobile node MUST use UDP encapsulation even if a NAT is not located between the mobile node and home agent.

Reserved This field is reserved for future use. It MUST be set to zero by the sender and ignored by the receiver.

Refresh time A suggested time (in seconds) for the mobile node to refresh the NAT binding. If set to zero, it is ignored. If this field is set to the all 1s it means that keepalives are not needed, i.e. no NAT was detected.

3.2.3 Extensions to the Binding Acknowledgement message

This specification extends the binding acknowledgement message with a new flag. The new flag is shown and described below.

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Status   |K|R|T| Reserved|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Sequence #   |           Lifetime           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

T

This flag indicates, when set, that the sender

Soliman

[Page 14]

of the binding acknowledgement supports the TLV-tunnel format.

4. Protocol operation

This section presents the protocol operation and processing for the messages presented above. In addition, this section introduces the NAT detection and traversal mechanism used by this specification.

4.1. Tunnelling formats

This specification allows two different types of UDP-based tunnelling formats to be used between the mobile node and its home agent or MAP. The two formats are vanilla UDP encapsulation and TLV-header UDP encapsulation. A vanilla UDP encapsulation format means the following order of headers:

IP (v4 or v6)
UDP
IP (v4 or v6)
Other headers

When using this format the receiver would parse the version field following the UDP header in order to determine whether the following header is IPv4 or IPv6. The rest of the headers are processed normally. The above order of headers does not take IPsec headers into account as they may be placed in different parts of the packet. The above format **MUST** be supported by all implementations of this specification and **MUST** always be used to send the binding update message.

A TLV-header UDP encapsulation is represented by the following order of headers:

IP (v4 or v6)
UDP
TLV-header
Other headers

The receiver of the TLV-header UDP encapsulated packet expects the TLV-header to follow UDP. The TLV header contains the type of the following message and its length. Hence, the TLV header can carry traffic other than IP.

The mobile node negotiates the format for tunnelling payload traffic during the binding exchange. If a mobile node prefers to use the TLV-header UDP encapsulation, it sets the T flag in the binding update sent to the home agent or MAP. If the receiver of the binding update supports the TLV-header format, it **SHOULD** set the T flag in the binding acknowledgement message. Otherwise, the T flag is cleared. The setting of the T flag in the binding acknowledgement indicates to

the mobile node that it must use the TLV-header UDP encapsulation format for all packets sent for the duration of the binding or until

NAT detection is done when the initial binding update message is sent from the mobile node to the home agent. When located in an IPv4-only foreign link, the mobile node sends the binding update message

encapsulated in UDP and IPv4. The source address of the IPv6 packet is the mobile node's IPv6 home address. The destination address is

the IPv6 address of the home agent. The IPv4 header contains the IPv4 care-of address in the source address field and the IPv4 address of the home agent in the destination address field.

When the home agent receives the encapsulated binding update it compares the IPv4 address of the source address field in the IPv4 header with the IPv4 address in the source address of the IPv6 header. If the two addresses match, no NAT device was in the path. Otherwise, a NAT device was in the path and the NAT detection option is included in the binding acknowledgement. The binding acknowledgement, and all future packets, are then encapsulated in UDP and IPv4. The source address in the IPv4 header is the IPv4 address of the home agent. The destination address is the IPv4 address received in the IPv4 header encapsulating the binding update (this address will be different from the IPv4 care-of address when a NAT is in the path). The source port in the packet is the home agent's source port. The destination port is the source port received in the binding update message. Note that the home agent stores the port numbers and associates them with the mobile node's tunnel in order to forward future packets.

Upon receiving the binding acknowledgement with the NAT detection option, the mobile node sets the tunnel to the home agent to UDP encapsulation. Hence, all future packets to the home agent are tunneled in UDP and IPv4. For all tunneled IPv6 packets, the source address in the IPv6 header is the mobile node's IPv6 home address and the destination address is the correspondent node's IPv6 address. All tunneled IPv4 packets will contain the mobile node's IPv4 home address in the source address field of the inner IPv4 packet and the correspondent node's IPv4 address in the destination address field. The outer IPv4 header is the same whether the inner packet is IPv4 or IPv6.

If no NAT device was detected in the path between the mobile node and the home agent then IPv6 packets are tunneled in an IPv4 header, unless the home agent forces UDP encapsulation using the F flag. The content of the inner and outer headers are identical to the UDP encapsulation case.

A mobile node **MUST** always tunnel binding updates in UDP when located in an IPv4-only network. Essentially, this process allows for perpetual NAT detection. Similarly, the home agent **MUST** encapsulate binding acknowledgements in a UDP header whenever the binding update is encapsulated in UDP.

In conclusion, the packet formats for the binding update and acknowledgement messages are shown below:

A. Binding update received by the home agent:

IPv4 header (src=V4ADDR, dst=HA_V4ADDR)

```
UDP header
  IPv6 header (src=V6H0A, dst=HAADDR)
    ESP-header
    Mobility header
      BU [IPv4 HA0]
      IPv4 CoA option
```

Where V4ADDR is either the IPv4 care-of address or the address provided by the NAT device. V6H0A is the IPv6 home address of the mobile node. The binding update MAY also contain the IPv4 home address option IPv4 HA0.

B. Binding acknowledgement sent by the home agent:

```
IPv4 header (src= HA_V4ADDR, dst=V4ADDR)
UDP header
  IPv6 header (src=HAADDR, dst=V6H0A)
    V6H0A (IPv6 home address)
  Mobility header
    BA ([IPv4 ACK], NAT DET)
```

Where V6H0A is IPv6 home address of the mobile node. The IPv4 ACK is the IPv4 address acknowledgement option, which is only included if the IPv4 home address option were present in the BU. The NAT DET is the NAT detection option, which MUST be present in the binding acknowledgement message if the binding update was encapsulated in UDP.

4.3. NAT Keepalives

If a NAT is detected, the mobile node will need to refresh the NAT bindings in order to be reachable from the home agent. NAT bindings can be refreshed through sending and receiving traffic encapsulated in UDP. However, if the mobile node is not active, it will need to periodically send a message to the home agent in order to refresh the NAT binding. This can be done using the binding update message. The binding update/acknowledgement pair will ensure that the NAT bindings are refreshed in a reliable manner. There is no way for the mobile node to know the exact time of the NAT binding. The default time suggested in this specification is NATKATIMEOUT. If the home agent suggests a different refresh period in the binding acknowledgement, the mobile node SHOULD use the value suggested by the home agent.

If the refresh time in the NAT detection option in the binding acknowledgement is set to the all 1s, the mobile node need not send messages to refresh the NAT binding. However, the mobile node may still be required to encapsulate traffic in UDP. This scenario may take place when a NAT is not detected, but the home agent still requires the mobile node to use UDP encapsulation.

It should be noted that a mobile node that does not need to be reachable (i.e. only cares about the session continuity aspect of

Mobile IP) does not need to refresh NAT binding. In this case, the mobile node would only be able to initiate communication with other nodes.

4.4. Mobile node operation

In addition to the operations specified in [[MIPv6](#)] and [[NEMO](#)], this specification requires mobile nodes to be able to support an IPv4 home address.

When sending an IPv6 packet containing a binding update while connected to an IPv4-only access network, mobile nodes MUST ensure the following:

- The IPv6 packet is encapsulated in the vanilla UDP encapsulation format.
- The source address in the IPv4 header is the mobile node's IPv4 care-of address
- The destination address in the IPv4 header is the home agent's IPv4 address.
- The source address in the IPv6 header is the mobile node's IPv6 home address.
- The IPv4 home address option MAY be included in the mobility header. This option contains the IPv4 home address. If the mobile node did not have a static home address it MAY include the unspecified IPv4 address, which acts as a request for a dynamic IPv4 home address. Alternatively, one or more IPv4 home address options may be included with requests for IPv4 prefixes (i.e. with the P flag set.).
- If the mobile node wishes to use UDP encapsulation only, it should set the F flag in the binding update message.
- If the mobile node prefers to use the TLV-header format, it should set the T flag in the binding update.
- The IPv6 packet MUST be authenticated as per [[MIPv6](#)], based on the mobile node's IPv6 home address.

When sending a binding update from a visited network that supports IPv6, the mobile node MUST follow the rules specified in [[MIPv6](#)]. In addition, if the mobile node has an IPv4 home address or needs one, it MUST include the IPv4 home address option in the mobility header. If the mobile node already has a static IPv4 home address, such address MUST be included in the IPv4 home address option. Otherwise, if the mobile node needs a dynamic IPv4 address, it MUST include the IPv4 unspecified address in the IPv4 home address option.

When the mobile node receives a binding acknowledgement from the home agent, it follows the rules in [[MIPv6](#)] and [[NEMO](#)]. In addition, the following actions MUST be made:

- If the status field indicated failure with error code 144, the mobile node MAY resend the binding update without setting the F

flag.

- If the mobility header includes an IPv4 address acknowledgement option indicating success, the mobile node should create two entries in its binding update list, one for the IPv6 home address and another for the IPv4 home address.
- If the NAT detection option were present, the mobile node MUST tunnel future packets in UDP and IPv4. This MUST be indicated in the binding update list.
- If no IPv4 address acknowledgement option were present, and an IPv4 home address option was present in the binding update, the mobile node MUST only create one binding update list entry for its IPv6 home address. The mobile node MAY include the IPv4 home address option in future binding updates.
- If an IPv4 address acknowledgement option were present and it indicates failure for the IPv4 home address binding, the mobile node MUST NOT create an entry for that address in its binding update list. The mobile node MAY include the IPv4 home address option in future binding updates.
- If the T flag was set in the binding update and the binding acknowledgement included a T flag set, the mobile node MUST use the TLV-header UDP encapsulation format.

4.4.1 Sending packets from a visited network.

When the mobile node is located in an IPv6-enabled network it sends and receives IPv6 packets as described in [[MIPv6](#)]. IPv4 traffic is encapsulated in IPv6 packets to the home agent.

When the mobile node is located in an IPv4 only network, it will send IPv6 packets to its home agent according to the following format:

```
IPv4 header (src=V4CoA, dst=HA_V4ADDR)
[UDP header]
  IPv6 header (src=V6HoA, dst=CN)
  Upper layer protocols
```

Where the UDP header is only used if a NAT were detected between the mobile node and the home agent, or if the home agent forced UDP encapsulation. V4CoA is the IPv4 care-of address configured by the mobile node in the visited network.

Similarly, IPv4 packets are sent according to the following format:

```
IPv4 header (src=V4CoA, dst=HA_V4ADDR)
[UDP header]
  IPv4 header (src=V4HoA, dst=V4CN)
  Upper layer protocols
```

Where the UDP header is only used if a NAT were detected between the

mobile node and the home agent, or if the home agent forced UDP encapsulation.

If the mobile node requested, and the home agent agreed, to use the TLV-header UDP encapsulation format, then the TLV-header would be used after the UDP header.

4.4.2 Movement detection in IPv4-only networks

[MIPv6] describes movement detection mostly based on IPv6-specific triggers. Such triggers would not be available in an IPv4-only network. Hence, a mobile node located in an IPv4-only network SHOULD use [DNAV4] for guidance on movement detection mechanisms in IPv4-only networks.

4.5. Home agent operations

In addition to the home agent specification in [MIPv6] and [NEMO], the home agent needs to be able to process the IPv4 home address option and generate the IPv4 address acknowledgement option. Both options are included in the mobility header. Furthermore, the home agent MUST be able to detect the presence of a NAT device and indicate that in the NAT detection option included in the binding acknowledgement.

A home agent must also act as a proxy for address resolution in IPv4 for the registered IPv4 home addresses of mobile nodes it is serving. Moreover, the administrative domain of the home agent is responsible for advertising the routing information of registered IPv4 mobile network prefixes of the mobile nodes.

In order to comply with this specification, the home agent MUST be able to find the IPv4 home address of a mobile node when given the IPv6 home address. That is, given an IPv6 home address, the home agent MUST store the corresponding IPv4 home address if a static one is present. If a dynamic address were requested by the mobile node, the home agent MUST store that address (associated with the IPv6 home address) after it's allocated to the mobile node.

When the home agent receives a binding update encapsulated in UDP and containing the IPv4 home address option, it needs to follow all the steps in [MIPv6] and [NEMO]. In addition, the following checks MUST be done:

- If the IPv4 care-of address in the IPv4 CoA option is not the same as the IPv4 address in the source address in the IPv4 header then a NAT was in the path. This information should be flagged for the binding acknowledgement.
- If the F flag in the binding update were set, the home agent needs to determine whether it accepts forcing UDP encapsulation. If it

does not, the binding acknowledgement is sent with error code 144.

UDP encapsulation MUST NOT be used when the mobile node is located in an IPv6-enabled link.

- If the T flag was set in the binding update, the home agent needs to determine whether it can accept the TLV-header encapsulation format. If it does, it should set the T flag in the binding acknowledgement. Otherwise, the home agent MUST NOT set the T flag in the binding acknowledgement.
- If the IPv4 home address option contains a valid unicast IPv4 address, the home agent MUST check that this address is allocated to the mobile node that has the IPv6 home address included in the home address option. The same MUST be done for an IPv4 prefix.
- If the IPv4 home address option contained the unspecified IPv4 address, the home agent SHOULD dynamically allocate an IPv4 home address to the mobile node. If none is available, the home agent MUST return error code 132 in the status field of the IPv4 address acknowledgement option. If a prefix were requested, the home agent MUST allocate a prefix with the requested length; otherwise the home agent MUST indicate failure of the operation with the appropriate error code.
- If the binding update is accepted for the IPv4 home address, the home agent creates a binding cache entry for the IPv4 home address/prefix. The home agent MUST include an IPv4 acknowledgement option in the mobility header containing the binding acknowledgement.

If the binding update is accepted for both IPv4 and IPv6 home addresses, the home agent creates separate binding cache entries, one for each home address. The care-of address is the one included in the binding update. If the care-of address is an IPv4 address, the home agent MUST setup a tunnel to the IPv4 care-of address of the mobile node.

When sending a binding acknowledgement to the mobile node, the home agent constructs the message according to [MIPv6] and [NEMO]. Note that the routing header MUST always contain the IPv6 home address as specified in [MIPv6].

If the care-of address of the mobile node were an IPv4 address, the home agent includes the mobile node's IPv6 home address in the destination address field in the IPv6 header. If a NAT were detected, the home agent MUST then encapsulate the packet in UDP and an IPv4 header. The source address is set to the home agent's IPv4 address and the destination address is set to the address received in the source address of the IPv4 header encapsulating the binding update.

After creating a binding cache entry for the mobile node's home addresses, all packets sent to the mobile node's home addresses are

tunneled by the home agent to the mobile node's care-of address. If a NAT were detected, packets are encapsulated in UDP and IPv4. Otherwise, if the care-of address is an IPv4 address, and no NAT were detected, packets are encapsulated in an IPv4 header.

4.5.1 Sending packets to the mobile node

The home agent follows the rules specified in [MIPv6] for sending IPv6 packets to mobile nodes located in IPv6 networks. When sending IPv4 packets to mobile nodes in an IPv6 network, the home agent must encapsulate the IPv4 packets in IPv6.

When sending IPv6 packets to a mobile node located in an IPv4 network, the home agent must follow the format negotiated in the binding update/acknowledgement exchange. In the absence of a negotiated format, the default format that MUST be supported by all implementations is:

```
IPv4 header (src= HA_V4ADDR, dst= V4ADDR)
[UDP header]
  IPv6 header (src=CN, dst= V6HoA)
  Upper layer protocols
```

Where the UDP header is only included if a NAT were detected between the mobile node and the home agent, or if the home agent forced UDP encapsulation. V4ADDR is the IPv4 address received in the source address field of the IPv4 packet containing the binding update.

When sending IPv4 packets to a mobile node located in an IPv4 network, the home agent must follow the format negotiated in the binding update/acknowledgement exchange. In the absence of a negotiated format, the default format that MUST be supported by all implementations is:

```
IPv4 header (src= HA_V4ADDR, dst= V4ADDR)
[UDP header]
  IPv4 header (src=V4CN, dst= V4HoA)
  Upper layer protocols
```

Where the UDP header is only included if a NAT were detected between the mobile node and home agent, or if the home agent forced UDP encapsulation.

4.6. Correspondent node operations

This specification has no impact on IPv4 or IPv6 correspondent nodes.

5. Security considerations

This specification allows a mobile node to send one binding update for its IPv6 and IPv4 home addresses. This is a slight deviation from

[MIPv6] which requires one binding update per home address. However, like [MIPv6], the IPsec security association needed to authenticate the binding update is still based on the mobile node's IPv6 home address. Therefore, in order to authorize the mobile node's IPv4 home address binding, the home agent MUST store the IPv4 address corresponding to the IPv6 address that is allocated to a mobile node. Therefore, it is sufficient for the home agent to know that the IPsec verification for the packet containing the binding update was valid provided that it knows which IPv4 home address is associated with which IPv6 home address. Hence, the security of the IPv4 home address binding is the same as the IPv6 binding.

In effect, associating the mobile node's IPv4 home address with its IPv6 home address moves the authorization of the binding update for the IPv4 address to the Mobile IPv6 implementation, which infers it from the fact that mobile node has an IPv6 home address and the right credentials for sending an authentic binding update for such address.

In cases where this specification is used for NAT traversal, it is important to note that it has the same UNSAF vulnerabilities associated with [RFC 3519](#). An attacker is able to hijack the mobile node's session with the home agent if it can modify the contents of the outer IPv4 header. The contents of the header are not authenticated and there is no way for the home agent to verify their validity. Hence, a man in the middle attack where a change in the contents of the IPv4 header can cause a legitimate mobile node's traffic to be diverted to an illegitimate receiver independently of the authenticity of the binding update message.

In this specification the binding update message MUST be protected using ESP transport mode. When the mobile node is located in an IPv4-only network, the binding update message is encapsulated in UDP as described earlier. However, UDP MUST NOT be used to encapsulate the binding update message when the mobile node is located in an IPv6-enabled network. If protection of payload traffic is needed when the mobile node is located in an IPv4-only network, encapsulation is done using tunnel mode ESP over port 4500 as described in [[TUNSEC](#)].

Handovers within private IPv4 networks or from IPv6 to IPv4 networks will have impacts on the security association between the mobile node and the home agent. The following section presents the expected behaviour of the mobile node and home agent in those situations.

[5.1](#) Handover interactions for IPsec and IKE

After the mobile node detects movement it updates its tunnel information depending on the network capability. If the new link is IPv4-only then the mobile node updates its tunnel to UDP using the

DSMIPv6 port and the local IPv4 address. Furthermore, the mobile node updates its local Security Association information to include its local IPv4 address and with the remote address being the home agent's

IPv4 address. The mobile node also removes binding update list entries for correspondent nodes since route optimisation cannot be supported while the mobile node is in an IPv4-only network. This may cause inbound packet losses as remote correspondent node are unaware of such movement. Following this, the mobile node sends the binding update message to the home agent.

Upon receiving the binding update message, the home agent updates its security association locally to include the mobile node's remote address as the IP address received in the outer header. When the mobile node is located in a private IPv4 network (which is detected as described above), this address is the public address allocated by the NAT. Based on the setting of the K flag in the binding update, the home agent updates its IKE security association to include the remote address as that received in the outer header of the binding update message. If the mobile node were located in a private IPv4 network this address is likely to be wrong as the NAT would likely allocate a different address to the IKE messages. Hence, the mobile node **MUST** update the IKE SA in one of two ways as follows. The mobile node **SHOULD** send an empty informational message, which results in the IKE module in the home agent to dynamically update the SA information. Alternatively, the IKE SA should be re-negotiated. Note that updating the IKE SA **MUST** take place after the mobile node has sent the binding update and received the acknowledgement from the home agent.

The mobile node **MAY** use [[MOBIKE](#)] to update its IKE SA with the home agent. Using MOBIKE requires negotiating this capability with the home agent when establishing the SA. In this case, the mobile node and the home agent need not update their IPsec SAs locally as this step is performed by MOBIKE. Furthermore, the use of MOBIKE allows the mobile node to update the SA independently of the binding update exchange. Hence, there is no need for the mobile node to wait for a binding acknowledgement before performing MOBIKE. The use of MOBIKE is **OPTIONAL** in this specification.

6. Protocol constants

NATKATIMEOUT	110 seconds
--------------	-------------

7. Acknowledgements

Thanks to the following members (in alphabetical order) of the MIP6 and NEMO Working Groups for their contributions, discussion, and review: Jari Arkko, Sri Gundavelli, Wassim Haddad, Conny Larsson, Ahmad Muhanna, Vidya Narayanan, Karen Neilsen and Keiichi Shima. Thanks to Karen Neilsen, Pasi Eronen and Christian Kaas-Petersen for raising the issue of IKEv2 interactions and proposing the solution

included in this document.

Soliman

[Page 25]

8. IANA considerations

The specification requires the following allocations from IANA:

- A UDP port is needed for the NAT traversal mechanism described in [section 4.1](#).
- The IPv4 home address option described in [section 3.1.1](#) requires an option type. This option is included in the Mobility header described in [\[MIPv6\]](#).
- The IPv4 address acknowledgement option described in [section 3.2.1](#) requires a new option type. This option is included in the Mobility header described in [\[MIPv6\]](#).
- The NAT detection option described in [section 3.2.2](#) requires a new option type. This option is included in the Mobility header described in [\[MIPv6\]](#).
- The IPv4 Care-of address option described in [section 3.1.2](#) requires an option type. This option is included in the Mobility header described in [\[MIPv6\]](#).

9. References

NORMATIVE

- [IPv6] S. Deering and B. Hinden, "Internet Protocol version 6 (IPv6) specification", [RFC 2460](#)
- [MIPv6] D. Johnson, C. Perkins and J. Arkko, "Mobility Support in IPv6", [RFC 3775](#), June 2004.
- [NEMO] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support protocol", [RFC 3963](#), January 2005.
- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [SEC] Arkko, J., Devarapalli, V., and F. Dupont, "Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents", [RFC 3776](#), June 2004.
- [TUNSEC] Huttunen, A., Swander, B., Volpe, V., DiBurro, L. and M. Stenberg, "UPD Encapsulation for IPsec ESP Packets", [RFC 3948](#), January 2005.

INFORMATIVE

- [BOOT] Giaretta, G. (Ed.), Kempf J., and V. Devarapalli, " Mobile IPv6 bootstrapping in split scenario", [draft-ietf-mip6-bootstrapping-split](#), June 2005, work in progress.

[HMIPv6] Soliman, H., Castelluccia, C., ElMalki, K., and L. Bellier,
"Hierarchical Mobile IPv6 Mobility Management (HMIPv6)",

Draftietf-mipshop-4140bis-04 November 2007.

- [INTBOOT] K. Chowdhury and A. Yegin, "MIP6-bootstrapping for the Integrated Scenario", [draft-ietf-mip6-bootstrapping-integrated-dhc-02](#), Work in Progress.
- [MIP-PB] Tsirtsis, G., and H. Soliman, "Mobility management for Dual stack mobile nodes, A Problem Statement", [RFC 4977](#), August 2007.
- [MIPv4] C. Perkins, "Mobility Support for IPv4", [RFC3344](#)
- [MOBIKE] P. Eronen, "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", [RFC 4555](#), June 2006.
- [SNRIO] Larsson, T., Gustafsson, E., and H. Levkowitz, "Use of MIPv6 in IPv4 and MIPv4 in IPv6 networks", [draft-larsson-v6ops-mip-scenarios-01.txt](#), February 2004.

10. Contributors

This document reflects discussions and contributions from several people including (in alphabetical order):

Vijay Devarapalli
E-mail: vijay.devarapalli@azairenet.com

James Kempf
E-mail: kempf@docomolabs-usa.com

Henrik Levkowitz
E-mail: henrik@levkowitz.com

Pascal Thubert
E-mail: pthubert@cisco.com

George Tsirtsis
E-mail1: G.Tsirtsis@Qualcomm.com
E-mail2: tsirtsisg@yahoo.com

Wakikawa Ryuji
ryuji@sfc.wide.ad.jp

Author's Address

Hesham Soliman
Elevate Technologies
E-mail: Hesham@elevatemobile.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the IETF's procedures with respect to rights in IETF Documents can be found in [RFC 3667](#) ([BCP 78](#)) and [RFC 3668](#) ([BCP 79](#)).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Full Copyright Statement

Copyright (C) The IETF Trust (2007). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

This Internet-Draft expires May, 2008.

