Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Telecommunications and Telematics

Design and Implementation of a Mobile Router

Ágoston Szabó

Master's Thesis

Advisors:

Miklós Aurél Rónai
M.Sc., Ericsson Research, Traffic Lab

Róbert Szabó
Ph.D., Budapest University of Technology and Economics

Budapest, 2003.

Nyilatkozat

Alulírott Szabó Ágoston, a Budapesti Műszaki és Gazdaságtudományi Egyetem hallgatója kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, és a diplomatervben csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Budapest, 2003. május

....
Szabó Ágoston

Acknowledgements

I would like to thank to all of my colleagues in Ericcson Traffic Lab Hungary for their support, especially Miklos Aurel Ronai, Octavian Pop, Csaba Keszei, Peter Gillich, Miklos Szeredi and Adam Marquetant.

Contents

1 Introduction 1

1.1 Mobile nodes and IP mobility . . . 1

1.2 Structure of the Thesis . . . 2

1.3 Mobile IP . . . 2

1.4 Mobile networks . . . 5

1.5 Related network elements and terminology . . . 6

2 Possible solutions for Network Mobility 11

2.1 Hierarchical Mobile IPv6 . . . 11

2.2 Prefix Scope Binding Updates . . . 14

2.3 The MRHA tunnel . . . 16

3 Analysis of the MRHA tunnel 25

3.1 The reason of choice . . . 25

3.2 Drawbacks . . . 26

4 BCMP – a micronet mobility protocol 29

4.1 Overview of the BCMP concept . . . 30

4.2 Address assignment . . . 31

4.3 Handover . . . 32

5 The interface between MRHA and BCMP 34

5.1 Testbed introduction 34

5.2 MRHA part of the testbed 36

5.3 Providing routing capabilities for the mobile router 42

5.4 Preserving the mobile network prefix 43

5.5 Making global routing available 48

5.6 Maintaining routes on the home agent 50

5.7 BCMP part of the testbed 52

5.8 Configuration of the nodes 53

5.9 Tunneling interface collision 54

6 Analysis of the system 56

7 Conclusion 60

List of Figures

1.1 A Mobile IP topology, no Route-Optimization (triangle problem) 3

1.2 Packet encapsulation 4

1.3 A typical mobile network topology 6

1.4 The analysed IPv6 scenario 9

2.1 A typical Hierarchical Mobile IPv6 topology with the Mobility Anchor Point 13

2.2 Prefix Scope Binding Update option field 15

2.3 Prefix Scope Binding Update's Network Prefix Sub-Option field 16

2.4 Binding cache content using Prefix Scope Binding Updates 17

2.5 IPv6 Binding Update message extended with the R-bit 19

2.6 Nested mobile network 20

2.7 Three bi-directional nested tunnels 21

2.8 Nested mobile network, pinball routing 22

2.9 Reverse Routing Header method, Step 1 23

2.10 Reverse Routing Header method, Step 2 23

2.11 Reverse Routing Header method, Step 3 23

2.12 Reverse Routing Header method, Mobile Router 3's packet encapsulation . 24

3.1 Crossover tunnel phenomenon. The red tunnel should conceptionnaly contain the entire blue tunnel. In this scenario the desired tunnel setup procedure is impossible to be performed. 27

4.1 BCMP topology in a foreign network 31

5.1 BCMP handles mobility inside the moving network, while MRHA maintains the mobility of the Mobile Router's network 35

5.2 Home network part of the testbed 37

5.3 Initial setting of MRHA part of the testbed with the Mobile Router at home 38

5.4 Topology and address assignments for testing of the mobile router's routing capabilities 48

6.1 Final topology of the testbed with initial node configuration, mobile router at home 57

6.2 Handoff latencies when the mobile router switches access routers 58

6.3 Handoff latencies when the mobile node switches BCMP access routers . . . 58

Chapter 1

Introduction

Mobile networks [MoNetTerm] are entire IP networks, moving as one unit with the ability to change the network's point of attachment to the Internet without losing their current connection sessions.

1.1 Mobile nodes and IP mobility

In our world today the technology of wireless devices is spreading quickly, there is no doubt about the importance of mobile equipments. Using wireless devices give people the experience of freedom since they are not bound to fixed locations. These mobile nodes are mostly organised in networks. In the world of Internet a unique IP address can be assigned to each participating element of a network. This address is an individual ID of each apparatus and thus, is used for destining and source-labeling data packets. If e.g. there is a laptop with a wireless interface and is able to connect to the Internet, it is possible to assign an IP address to it and place it in an existing IP subnet. If the laptop moves away from this network and connects to the Internet somewhere else, a protocol is needed to maintain its continous connection sessions. Nevertheless, the mobility should be transparent to other hosts in the network. To put it an other way, the same addressing and naming conventions must apply for mobile hosts as for fixed ones, since from a stationary end-system's perspective mobile devices should appear as any other stationary hosts connected to the Internet. The protocol handling with this matter is called Mobile IP [MobileIP, MobileIPWeb, MobileIPTerm].

1.2 Structure of the Thesis

In my thesis I will introduce the concept of IP mobility and mobile networks. I will present the main purposes and behaviour of the Mobile IP protocol and will present the problem scope of handling mobility of nodes in moving networks. I will review some of the solutions proposed for this matter and will present the solution called MRHA in more detail. After the discussion of moving networks' mobility solutions I will give a short overview of an intra-network mobility protocol called BCMP. In the second half of the thesis I will present my implementation of a mobile router that has both MRHA and BCMP functionalities.

1.3 Mobile IP

This protocol gives a solution for the problems and requirements regarding mobility of nodes in the Internet. There are various approaches of mobility handling using Mobile IP; three of them will be detailed below.

Plain Mobile IP

Basically, Mobile IP defines several network sections and introduces a special network element. Figure 1.1 shows a Mobile IP topology with communication routes explained below.

The participating network elements are the following:

Mobile Host: The IP device which is able to change its point of attachment to the Internet. The names like mobile "node", "equipment", "device" and "host" are equivalent.

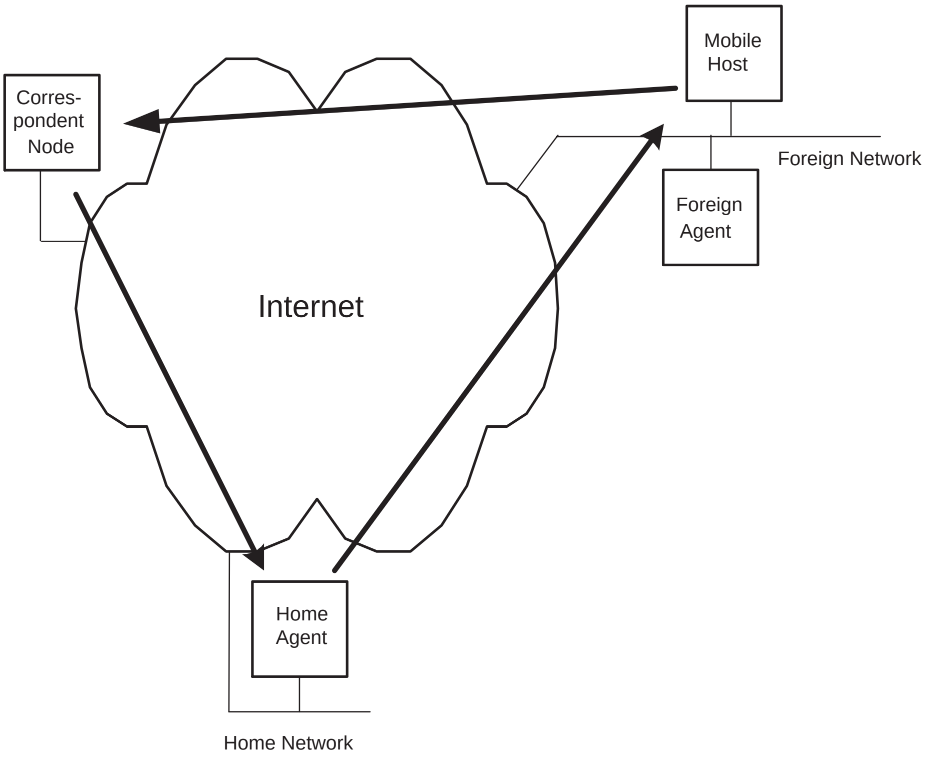

Figure 1.1: A Mobile IP topology, no Route-Optimization (triangle problem)

Home Network: While in the home network, the mobile node does not need any mobility support to send or receive data packets, it acts like any stationary IP device.

Foreign Network: When the mobile host leaves its home network and connects to the Internet elsewhere, its new place of connection is called foreign network.

Care-of-Address: When the device leaves its home network, the IP address that belonged to the host so far becomes unrelevant, since the host cannot be found on that IP address any longer. Hence, when joining a foreign network, a new IP address must be obtained, which is called the device's Care-of-Address.

Foreign Agent: The device in the foreign network that offers a care-of-address for the mobile host when it enters the foreign network.

Correspondent Node: A node the mobile host is communicating with. Any host on the Internet can act as a correspondent node, but we consider it to take place somewhere outside of both the home and the foreign network.

Home Agent: A special IP device in the home network. After leaving the home network, joining a foreign one and receiving a new Care-of-Address, the mobile host must do something to let the world know about this change. What it actually does is that it informs its home agent in its home network about this new care-of-address. The message sent is called Binding Update and contains among other things the host's home address (the host's IP address in the home network) and the care-of-address. The home agent receives this message and stores the information in its Binding Cache. After this procedure, the home agent starts functioning: it starts to "act" like it was the mobile host: it takes the mobile host's place in the home network. Thus, any packet destined to the old IP address to the home network is caught by the home agent. Since the home agent has the care-of-address of the mobile host in its binding cache, it encapsulates and forwards the data packet to the real recipient, to the mobile host. The encapsulation is a method when the data packet is given a new header with the care-of-address in the IP destination field (see figure 1.2). This encapsulated packet is then forwarded to the mobile host in the foreign network.

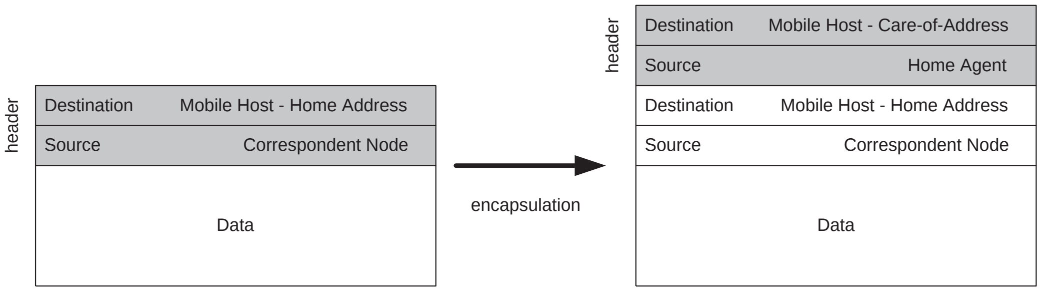

Figure 1.2: Packet encapsulation

Of course the home agent must be informed whenever the mobile host changes its point of attachment to the Internet, thus obtaining a new care-of-address. Every time such change occurs, the mobile host sends another binding update for the home agent with the proper information within.

Mobile IP with Route-Optimization

There is a major problem with the solution described above: the communication between a mobile host and a correspondent node is not effective enough because every packet sent by the correspondent node has to visit the home agent first instead of going straight to the mobile host. This is the so called "triangle problem", since the route leading from the correspondent node towards the mobile host is „triangle-shaped" (see figure 1.1). To enhance effectiveness it is possible to allow the correspondent node to cache the care-of-address of the mobile host and thus to be able to destine the packets directly to this cached address. This way this detour across the home agent can be avoided.

Mobile IPv6

The new version of Internet Protocol, called IPv6 [IPv6] was designed to improve the current version of IP, IPv4. The new features include expanded addressing capabilities, header format simplification, improved support for extensions and options, enhanced authentication and privacy management and better QoS (Quality of Service) performance by means of flow labeling capabilities. Mobile IPv6 [MobileIPv6] is the solution of host mobility support similar to the one designed for IPv4 mobile networks, but is extended with the enhanced capabilities of IPv6.

1.4 Mobile networks

The Mobile Internet Protocol handles continuous connection sessions only for single mobile IP equipments. However, there are cases when set of mobile devices form a mobile network and it is desired to maintain the connection of the whole mobile network. A possible scenario for a mobile network could be a set of mobile devices (e.g. Personal Digital Assistant (PDA), cellular phone, laptop) attached on and used by a person, which equipments thus formulate a Personal Area Network (PAN). This PAN may be in wireless connection with some kind of sensors or transmitters deployed in offices or transport vehicles with connection to the Internet or other networks. In a scenario like this people are for example able to use their laptops with other wireless devices (e.g. mouse, keyboard) on an aeroplane and surf the web while flying. The aeroplane is connected to the Internet through a router with radio interface that communicates with base stations and/or satellites.

1.5 Related network elements and terminology

Figure 1.3 shows a typical mobile network topology.

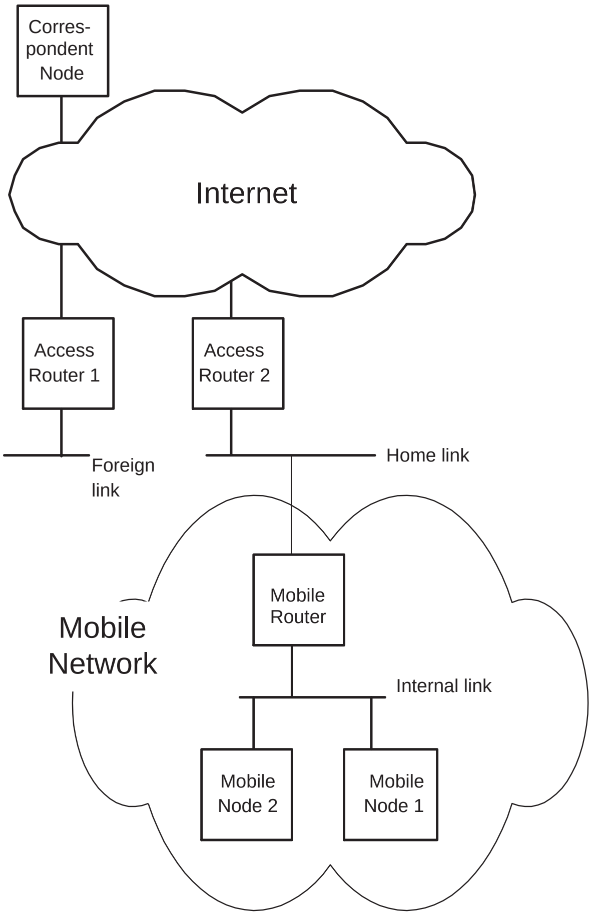

Figure 1.3: A typical mobile network topology

The participating entities of a mobile network are the following:

Access Router: The connection point for mobile networks to the Internet. The role of the access router is similar to the one of Access Points in Wireless Local Area Networks (WLANs).

Mobile Router: The router that connects the mobile network to the access router, thus to the Internet. A mobile router is able to transmit data packages by means of its interfaces. The mobile router forwards data towards the Internet through its egress interface and forwards data towards the mobile network via its ingress interface.

Node behind the mobile router: A device which is part of the mobile network and does not have direct connection to the access router, thus to the Internet. Every packet, either originating from this node and destined to the Internet or originating from the Internet and destinded to this node must pass across the mobile router.

Mobile Network: A network that consists of one or more mobile routers and one or more nodes behind the mobile router.

Mobile Node: A node, either a host or a router, which is able to change its point of attachment to other nodes or network without losing its active connection session.

Fixed Node: A node, either a host or a router, which is not able to change its point of attachment to other nodes or network unless it closes its current active connection sessions and retreives an another IP address.

As we will see, mobility support in a scenario with a mobile network behind a mobile router using unmodified Mobile IP or IPv6 is not thoroughly solved. The mobile router follows the steps of the plain mobile protocol – when leaving its home network and joining a foreign one it gains a new care-of-address and sends the binding update to its home agent located in the home network. From this time on every packet destined to the mobile router will be encapsulated and forwarded to the care-of-address of the mobile router. The problem arises when the communication between a correspondent node and a node behind the mobile router starts. When the mobile router (and hence the whole mobile network) changes point of attachment to the Internet, only the mobile router informs its home agent about this action, since from the nodes behind the mobile router's point of view nothing changed at all. Thus, the outside world is not informed about the movement of the nodes behind the mobile router. To illustrate the problem scope let us see examples [PrefixScope] in a typical mobile network scenario. We will examine what happens when a correspondent node tries to communicate first with the mobile router, then with a node behind the mobile router. Figure 1.4 shows an IPv6 scenario where the communication can be analysed according to our needs. We can see that the mobile network (which here consists of a mobile router and two mobile nodes) previously moved away from the home network, joined a foreign one via an access router router and the mobile router gained a new care-of-address.

First experiment: Communication between the correspondent node and the mobile router

Let us assume that the correspondent node pings the mobile router. The data packet's recipient is the mobile router's home address. When the packet reaches the home network via access router 2, access router 2 sends an ARP (Adress Resolution Protocol) request message [ARP] broadcasted on the home network to gain the mobile router's Ethernet (or MAC - Media Access Control) address. The home agent receives this request and since it pretends to be the mobile router, replies to the access router with the home agent's Ethernet address. The data packet is then forwarded to the home agent. The home agent looks up its binding cache and finds the entry with the mobile router's care-of-address. The home agent then encapsulates and forwards the data packet to the mobile router's care-of-address, where it is then correctly received. For this case the Mobile IPv6 protocol serves as an adequate mobility solution.

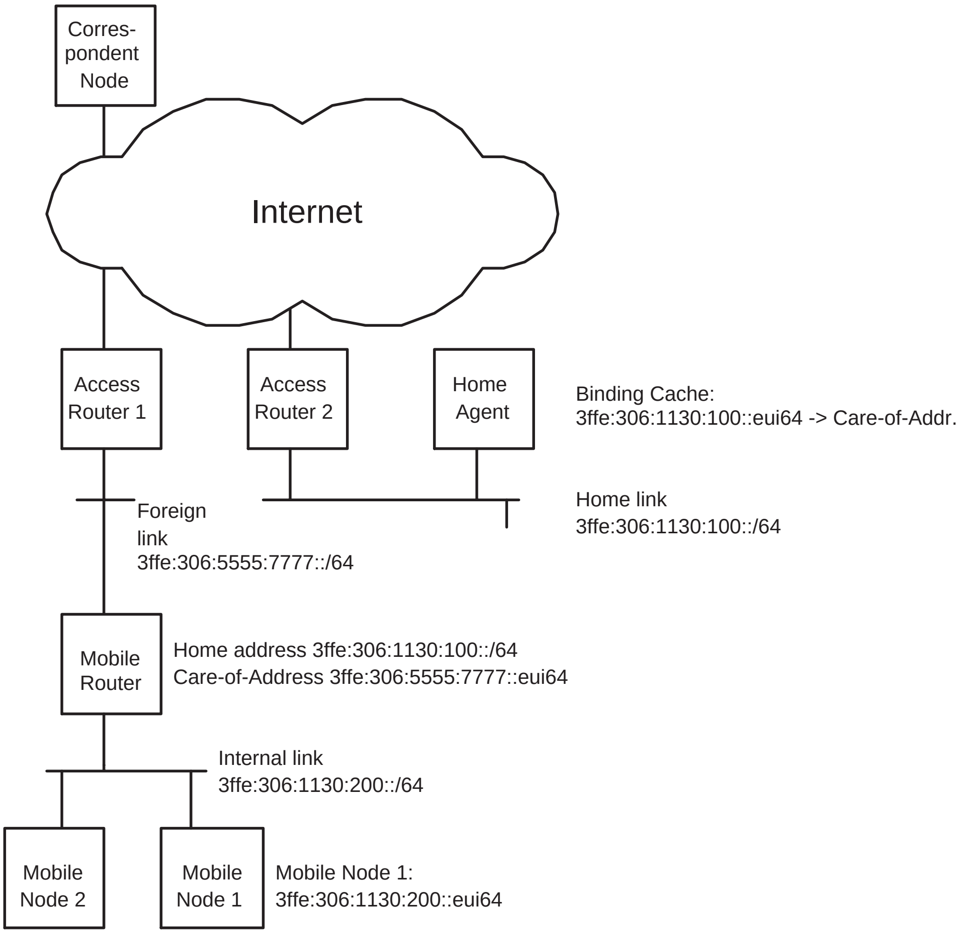

Figure 1.4: The analysed IPv6 scenario

Second experiment: Communication between the correspondent node and mobile node 1

In this second experiment the correspondent node pings mobile node 1's home address, which is 3ffe:306:1130:200::eui64. When the data packet reaches the home network, the access router 2 checks its routing table to find the next hop towards mobile node 1. The result of the search is the mobile router, so the access router, like in he previous case, sends an ARP request broadcasted on the home network. The home agent answers this request on behalf of the mobile router and thus the data packet is then forwarded to it. The home agent has now a packet with a recipient address that the home agent does not have a routing entry to. Thus, the home agent looks up the default routing entry, which is the access router's address. The ping packet is so returned to the access router. The access router repeats the actions once performed before, and sends the packet back to the home agent. Thus, the ping packet enters a routing loop and keeps circling between the access router and the home agent until the packet's TTL (Time-to-Live) expires. We can see that the plain Mobile IPv6 protocol designed for handling mobility for single nodes does not serve an adequate mobility solution for nodes behind mobile routers. Thus, extension of the protocol is needed in order to maintain the mobility of entire mobile networks.

Chapter 2

Possible solutions for Network Mobility

As the experiments showed, the Mobile IP and IPv6 methods do not give a solution for maintaining mobile networks' continous connection sessions. Thus, there has been a need for new techiques to be implemented. There are several solutions proposed by various working groups and task forces for this problem scope; in this section some of them will be presented.

2.1 Hierarchical Mobile IPv6

This technology [HMIPv6] introduces some extensions to Mobile IPv6 to enhance its abilities and thus to support network mobility. The solution aims to make minimal changes in the existing Mobile IPv6 technology. The technology does minimal changes in mobile node and home agent operations and does not make changes in the correspondent node operation. However, the extensions reqiure the introduction of an additional network element.

When the mobile node enters a foreign network, it is needed to obtain a local care-of-address from the mobile router maintaining the foreign network. This information then needs to be submitted to the home agent of the mobile node and if this process succeeds, the mobile node is ready to receive data. After a while, the mobile node may decide to change its connection point to the Internet by leaving the current foreign network and joining another one. According to the Mobile IPv6 solution it is needed to obtain a new care-of-address from the mobile router in the new foreign network. When the new care-of-address is assigned to the mobile node's interface, another binding update message has to be sent to the mobile node's home agent to maintain the connection session. Lots of mobile nodes and lots of movements between foreign network imply lots of binding update messages to be sent. This causes signalling overhead in the networks (especially for the mobile routers) and thus, reduces network throughput efficiency. This phenomenon is called Binding Update Storm (BU Storm) and is of critical importance regarding Quality of Service (QoS) applications. The Hierarchical Mobile IPv6 technology gives a solution for this problem by introducing a new network element, the mobility anchor point (MAP). A Hierarchical Mobile IPv6 topology can be seen on figure 2.1.

Considering a tree-type topology on the Internet, mobility anchor points are installed on a higher level than Mobile Routers. So when a mobile node enters a mobile network, it also enters an area of a mobility anchor point. Installation of the mobility anchor point aims to minimize latency caused by the handoff between foreign networks. Handoff delay typically occurs when the home agent is updated when new care-of-address is obtained. The procedure of informing the home agent is different from the one using traditional Mobile IPv6. Hierarchical Mobile IPv6 distinguishes two types of care-of-addresses, On-link and Regional ones. On-link care-of-address of a mobile node always changes when the mobile node leaves a network and joins another one – it is basically the traditional Mobile IPv6 care-of-address of a mobile node. The regional care-of-address of a mobile node is stored by the mobility anchor point and is permanent as long as the mobile node is in the domain of the mobility anchor point. This domain consists of the foreign networks attached to the access routers within the mobility anchor point's reach. Thus, in such a domain a regional care-of-address must be assigned only once, when the mobile node first enters a foreign network belonging to the domain. The home agent of the mobile node is

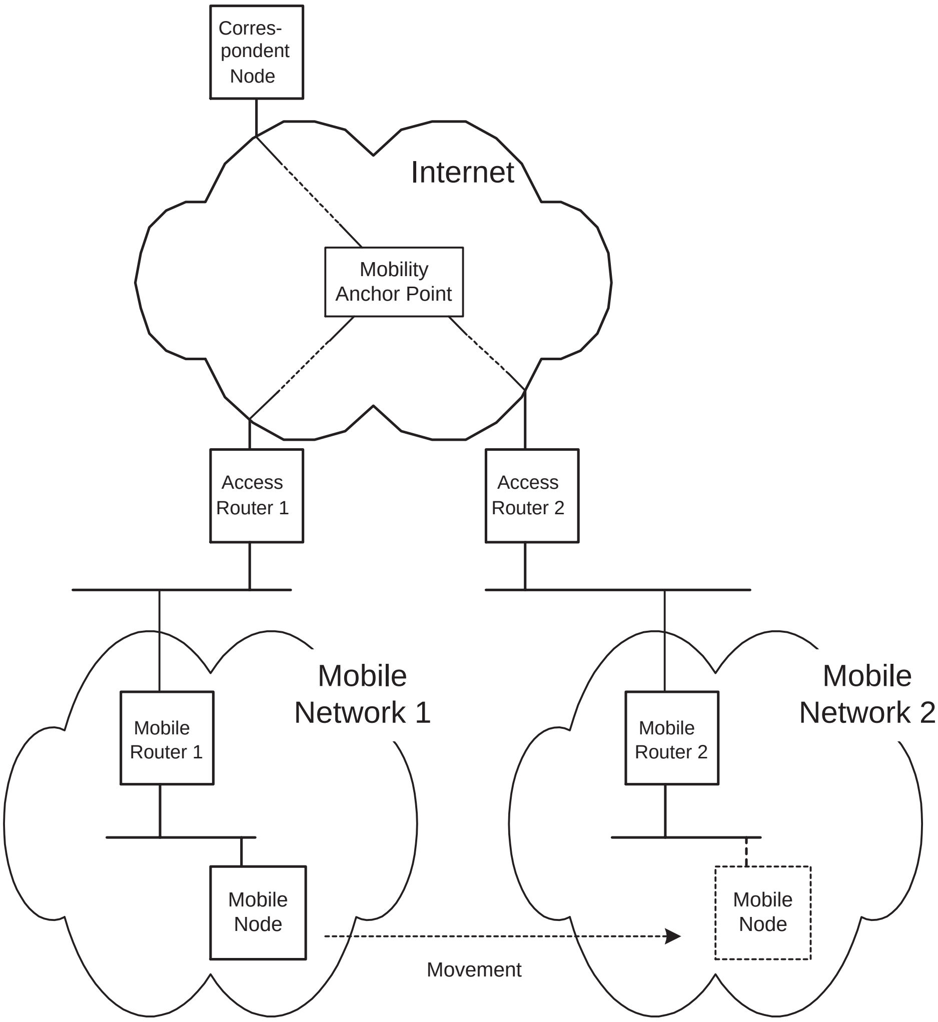

Figure 2.1: A typical Hierarchical Mobile IPv6 topology with the Mobility Anchor Point

informed about this address in a binding update message. From this time on the mobility anchor point acts like some local home agent of the mobile node since it receives data packets on behalf of the mobile node and tunnels these packets towards its on-link care-of-address. Hence, the mobility anchor point maintains a table with regional care-of-address –> on-link care-of-address associations. When the mobile node moves from a network to another and obtains a new on-link care-of-address it sends a binding update message to the mobility anchor point which thus updates its associations table and redirects data packets towards the new on-link address. The binding update is not forwarded towards the home agent of the mobile node; in its aspect the mobile node is still in the same foreign network as so far. The time needed to redirect traffic towards the mobile node's new position takes less time than using the plain Mobile IP protocol since the round-trip-time of the binding update message and its acknowledgement is reduced by not sending these messages through the whole Internet. An update in the home agent is needed only when the mobile node changes mobility anchor point domains, which obviously occurs less frequent than changing foreign networks.

Although the Hierarchical Mobile IPv6 solution is everywhere credited among the technologies for supporting mobile networks, it is seen that it is actually an extension to Mobile IPv6 and yet supports the mobility only of individual mobile nodes, where every node has to handle its own mobility. However, the application of Hierarchical Mobile IPv6 with other, real mobile network supporting technologies can be a good solution for avoiding the phenomenon that causes major problems while considering mobile networks – the binding update storm. Besides, the handoff speed of mobile nodes between foreign networks can also be improved.

2.2 Prefix Scope Binding Updates

This method [PrefixScope] is an extension of the Mobile IPv6 technology proposed to support network mobility. This proposal aims to reduce the signaling overhead caused by the phenomenon called binding update storm by making changes in the binding update mechanism used in the traditional Mobile IPv6 technology. This solution implies changes in the operation of the mobile router, the home agents and the correspondent nodes as well.

Either being in a moving or a non-moving network, the nodes own an address with a prefix that is specific on that network. This proposal utilizes the advantage of the ability of referring to a whole set of nodes with one single network prefix. By sending a modified binding update message, it is possible to associate various nodes with one single care-of-address. The binding update contains the care-of-address of the mobile router and a network prefix referring to the network it supervises. Thus, sending binding update messages becomes the task of the mobile router, which sends this information to its home agent and, if enabling route optimization, to all of the correspondent nodes of itself and the correspondent nodes of the nodes behind it. This implies that with a single binding update message a whole network can be registered in one step independently from the number of nodes. This is especially useful when a correspondent node communicates with several nodes in the mobile router's network.

The solution proposes to extend the Mobile IPv6's binding update message [MobileIPv6] with a new type of Mobility Option. Figure 2.2 shows the modified option field encoded in type-lenght-value (TLV) format. The additional ,,P" bit taken from the ,,Reserved" field stands for ,,Prefix Scope Registration". When it is set, it indicates that the sender of this binding update message attempts to register a care-of-address for an entire network. It also requests the receiver to process the Network Prefix Sub-Option and to re-route packets to the destination address that corresponds to the network prefix. The ,,Reserved" field is reduced to 3 bits and must be ignored by the receiver. Figure 2.3 shows the Network Prefix Sub-Options Field.

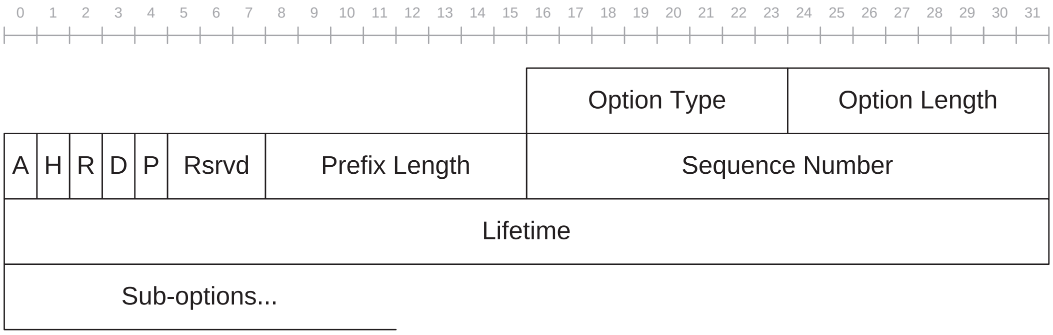

Figure 2.2: Prefix Scope Binding Update option field

Not only the mobile router's operation, but also the home agent's and the correspondent nodes' operation has been extended to handle Prefix Scope Binding Update messages. As seen before, it is the mobile router's task to set the ,,P" bit to 1, to fill the Network Prefix Sub-Option field and to send this message to the home agent and to the correspondent

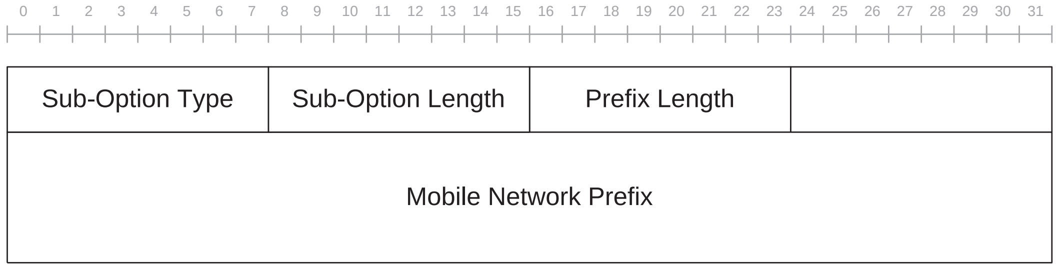

Figure 2.3: Prefix Scope Binding Update's Network Prefix Sub-Option field

nodes.

Upon receiving a binding update, the home agent performs validity checks according to [MobileIPv6]. In addition, it checks whether the ,,P" bit is set. If so, it further checks if the Network Prefix Sub-Option exists. If everything is as it is expected, the home agent first creates a ,,normal" binding entry in its binding cache accordingly to the Mobile IPv6 protocol and then creates a second one with the mobile router's care-of-address associated with the mobile network's prefix. This cache-updating operation of a correspondent node upon receiving a Prefix Scope binding update is the same. The figure 2.4 shows the content of the binding caches in a certain mobile network scenario.

2.3 The MRHA tunnel

As an example in the previous chapter (see figure 1.4) showed, when the mobile router is in a foreign network, and the home agent at the home link intercepts every packet on behalf of the mobile router, there are cases when this method is not effective. Failure occurs in packet transmission if the packet's destination is a node behind the mobile router – the home agent pretends to be the next hop on the home link on behalf of the mobile router, but since the home agent itself does not maintain a routing table with information about the destination node, the packet keeps looping in the home network until its Time-to-Live entry expires. The method called MRHA tunneling [MRTunnel] has been designed to give a solution for this kind of problem.

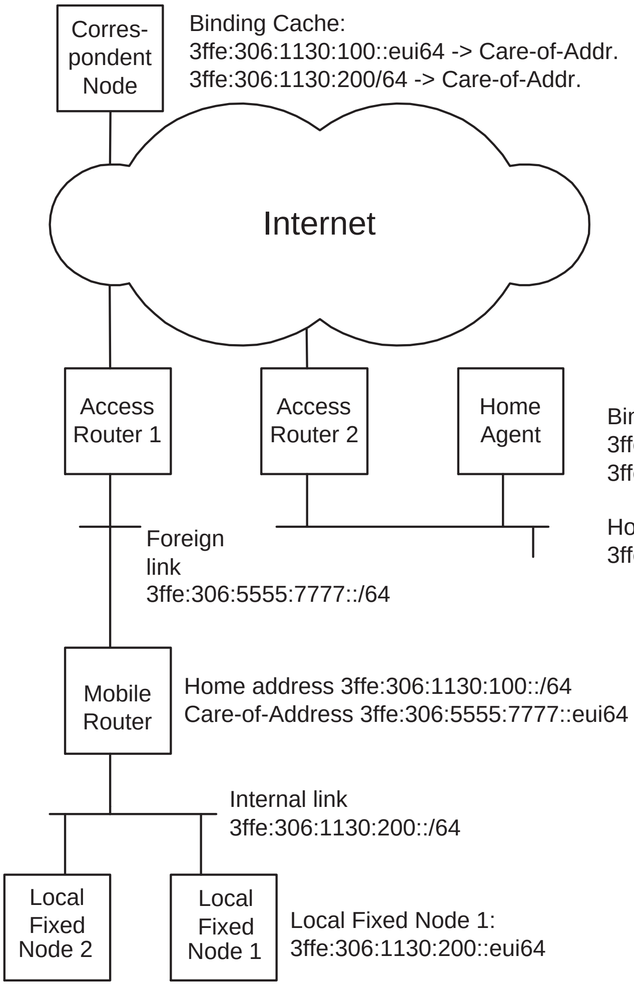

Figure 2.4: Binding cache content using Prefix Scope Binding Updates

The abbreviation MRHA stands for the expression ,,Mobile Router - Home Agent". These entities are the two endpoints of an imaginery tunnel. The tunnel is not a physical wire, it is a route across the Internet consisting of routers and the connections between them. The route is not even fixed – it can change at any time – only the endpoints are given. Eventually, ,,tunnel" expresses the fact that if data is sent from one end it will arrive to the other end, anything happens on the way between is irrelevant. In the tunnel data can be sent in both directions, thus, it is a bidirectional channel.

According to the MRHA concept communication between the mobile router and its home agent is always performed via the MRHA tunnel. Thus, any data that is about to leave in the mobile router's egress interface must be destined, rerouted towards the home agent. Similarly, if the home agent intercepts any data on behalf of the mobile router, the data must be automatically destined towards the mobile router regardless of the content of the packets. The method used for tunneling data is called encapsulation (see figure 1.2). The sender endpoint of the tunnel adds an extra header to every data packet that must be tunneled. This header consists of two addresses. The source address field contains the sender endpoint's address – if the sender is the home agent, then this address is the home agent's address, but if the sender is the mobile router, then the care-of-address must be revealed. Filling the destination field is quite similar: if the endpoint of the tunnel is the home agent, then the destination address is the address of the home agent, but if the endpoint of the tunnel is the mobile router, then the care-of-address of the mobile router must be revealed.

This solution requires support in home agent and mobile router functionality, since both entities must understand that a tunnel is to be established, and packets for the mobile network must be routed through that tunnel. The establishment of the MRHA tunnel is initiated by the mobile router when it sends a binding update message to its home agent. To increase the amount of information submitted by signaling there is a modified binding update message suggested to be sent while establishing an MRHA tunnel. This is an extension of the Mobile IPv6 by extending the binding update message with a mobile router bit, the „R-bit". The modified binding update message can be seen on figure 2.5. R-bit is taken from the reserved field and when set, it indicates that the home agent should create a route to tunnel the mobile network packets to the care-of-address included in the binding update. If the R-bit's value is 0 and a previous tunnel existed, the home agent should delete it and cease forwarding data packets towards the mobile network. Thus, by setting the R-bit to 0, the mobile router is able to cease routing functionality and, in effect, become a mobile node.

Similarly to the method presented in the previous section, the Mobility Options field can be used to submit mobile network prefix information to the home agent.

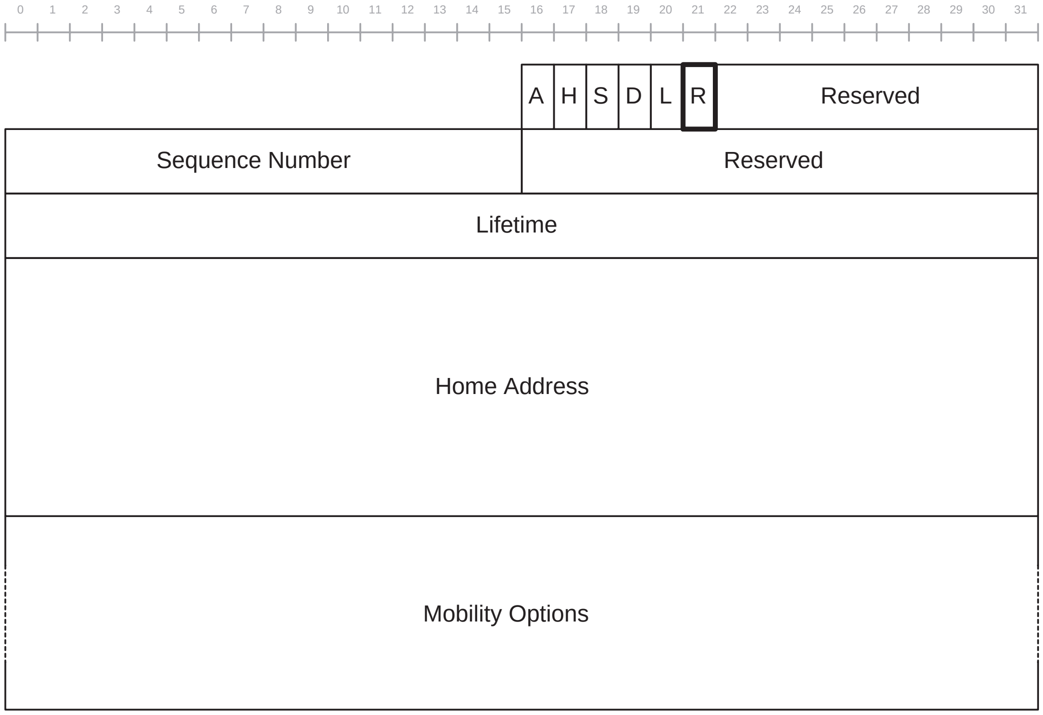

Figure 2.5: IPv6 Binding Update message extended with the R-bit

Regarding mobile network topologies and the related problems there is an important issue to consider – the nesting of mobile networks [RevRouteHeader]. An example of a nested mobile network is when a mobile router joins a foreign network not directly connecting to the access router but to a link in a subnetwork already maintained by another mobile router. A possible scenario of nested networks can be seen on figure 2.6. Following the principles of the MRHA method, all mobile routers in the nested mobile network maintain tunnels towards their respective home agents. Taking the topology on figure 2.6 as an example if Mobile Router 3 forwards data to a correspondent node through the Internet, the packet must be encapsulated and routed towards its home agent, which is henceforth referred to as Home Agent 3. As seen, the route towards Home Agent 3 leads through Mobile Router 2. When Mobile Router 2 intercepts this packet, it only perceives that a packet must be forwarded somewhere across the Internet. Thus, since Mobile Router follows the principles of the MRHA method, it encapsulates the packet again and redirects it towards its own home agent, Home Agent 2. The same steps are

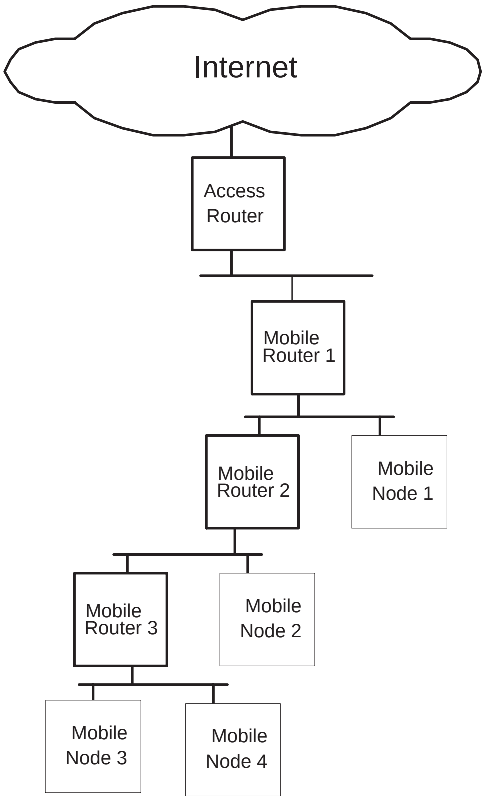

Figure 2.6: Nested mobile network

applied at the next level of the topology, at Mobile Router 1, but the packet is routed towards Home Agent 1. At this point we have three bi-directional nested tunnels, as illustrated on figure 2.7. This method implies two major problems. The first one is the overhead caused by the increase of packet size when an extra IPv6 is added to all the packets at every level of nesting. There has been a method designed to reduce the size of encapsulated IPv6 packet headers [RedAddrDel] but it cannot be implied in this case because in nested networks both the destination and the source addresses are different hop to hop. The second problem this method arises is the so called ”pinball routing” phenomenon. Considering the same communication example as above, when a mobile node behind mobile router sends data to a correspondent node the packets are encapsulated three times and are sent via three nested tunnels. This implies that when the packets enter the internet they are not forwarded straight to the correspondent node, they must first

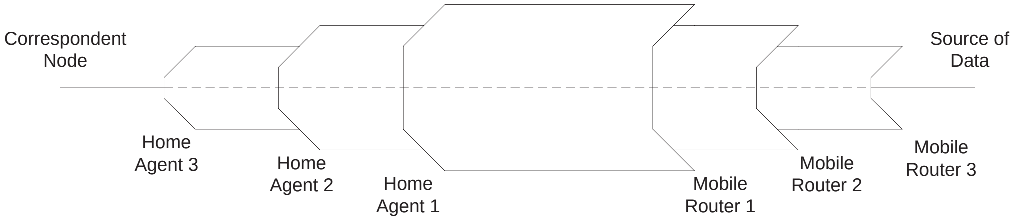

Figure 2.7: Three bi-directional nested tunnels

visit each home agents instead. At each home agent the packets are decapsulated and then forwarded towards the destination found in the modified packets' headers. An illustration of this phenomenon can be seen on figure 2.8. Depending on the relative location of the home agents, this way of routing can be very inefficient because of the delay caused by the detours.

All these problems can be handled with a solution suggested in [RevRouteHeader]. According to this extension the encapsulation operation of mobile routers must be changed and as a result only one bidirectional tunnel will be created. During the introduction of this implementation the topology presented above will be used as an example.

Step 1: The first mobile router on the path, Mobile Router 3, in addition to tunneling the packet to its home agent, adds a reverse routing header with three preallocated slots. The bottom slot is equivalent to the Mobile IPv6's Home Address option. The outer packet has Mobile Router 3's care-of-address as source address and Home Agent 3's address as destination (see figure 2.9).

Step 2: The second router on the path, Mobile Router 2 notices that the packet already contains a Reverse Routing Header and so it overwrites the source address of the packet with its own care-of-address, putting the old source address, Mobile Router 3's care-of-address in the first free slot of the Reverse Routing Header (see figure 2.10).

Step 3: In general, the process followed by the second router is repeated by all the routers in the path. In this example the last router performing this method is Mobile Router 1. When leaving Mobile Router 1, the packet looks like figure 2.11.

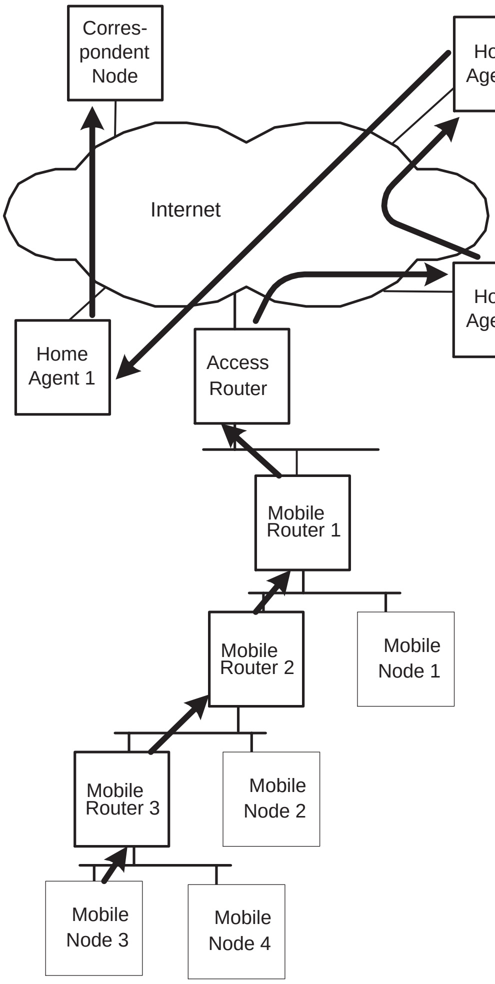

Figure 2.8: Nested mobile network, pinball routing

As seen, the destination address of the packet does not change, it remains Home Agent 3's address. Thus, when the packet leaves the mobile network, it is forwarded towards Home Agent 3. When it intercepts the packet it notices that the packet contains a Reverse Routing Header and it looks at the bottom entry and sees Mobile Router 3's home address. This entry is used as if it was a Mobile IPv6 Home Address destination option, i.e. as an index into the binding cache. After performing authenticity and binding cache checks, Home Agent 3 forwards the inner packet towards the correspondent node. Home Agent 3 stores two items in the binding cache entry associated with Mobile Router 3: the address entries from the Reverse Routing Header, to be used for building reverse headers, and the

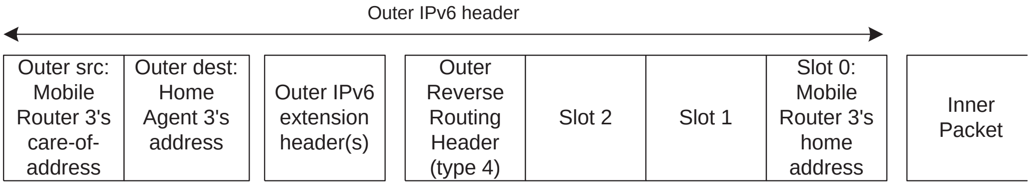
Outer IPv6 header
Outer src: Mobile Router 3's care-of-address
Outer dest: Home Agent 3's address
Outer IPv6 extension header(s)
Outer Reverse Routing Header (type 4)
Slot 2
Slot 1
Slot 0: Mobile Router 3's home address
Inner Packet

Figure 2.9: Reverse Routing Header method, Step 1

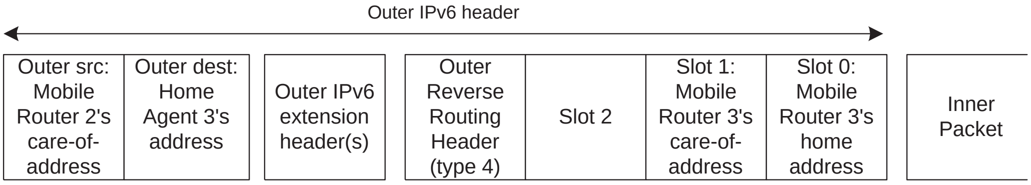
Outer IPv6 header
Outer src: Mobile Router 2's care-of-address
Outer dest: Home Agent 3's address
Outer IPv6 extension header(s)
Outer Reverse Routing Header (type 4)
Slot 2
Slot 1: Mobile Router 3's care-of-address
Slot 0: Mobile Router 3's home address
Inner Packet

Figure 2.10: Reverse Routing Header method, Step 2

packet source address (Mobile Router's care-of-address), to be used as the first hop. From this time on, when Mobile Router 3 intercepts a plain IPv6 packet destined to a node behind Mobile Router 1 it does a binding cache lookup and finds information on building a Reverse Routing Header and Mobile Router 1's care-of-address to be used as the first hop. Then it encapsulates the packet as seen on figure 2.12.

As seen, applying the extension of reverse routing headers for nested networks can be used to avoid the side-effects of tunneling like the increase of packet size and the ,,pinball-routing". The method requires the addition of only one extra header to each packet independent of the number of levels in the mobile network. As the number of levels increase, only additional address slots must be appended to the Reverse Routing

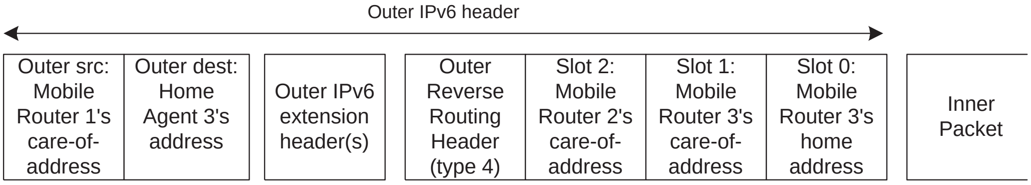
Outer IPv6 header
Outer src: Mobile Router 1's care-of-address
Outer dest: Home Agent 3's address
Outer IPv6 extension header(s)
Outer Reverse Routing Header (type 4)
Slot 2: Mobile Router 2's care-of-address
Slot 1: Mobile Router 3's care-of-address
Slot 0: Mobile Router 3's home address
Inner Packet

Figure 2.11: Reverse Routing Header method, Step 3

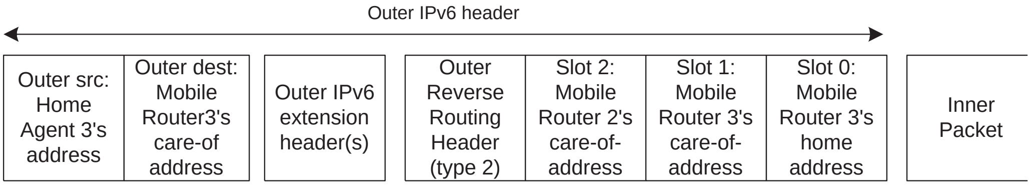
Outer IPv6 header
Outer src: Home Agent 3's address
Outer dest: Mobile Router3's care-of address
Outer IPv6 extension header(s)
Outer Reverse Routing Header (type 2)
Slot 2: Mobile Router 2's care-of-address
Slot 1: Mobile Router 3's care-of-address
Slot 0: Mobile Router 3's home address
Inner Packet

Figure 2.12: Reverse Routing Header method, Mobile Router 3's packet encapsulation

Header, which causes negligible packet enlargement. This proposal also shortens the path the packets would take with the traditional MRHA tunneling method. Instead of going on a ,,pinball" kind of route, the packets are straightly forwarded from the Home Agent 3 towards the correspondent node.

Chapter 3

Analysis of the MRHA tunnel

While examining the possible solutions for network mobility I chose the concept of the MRHA tunneling for further analysis. In this section the features, the advantages and drawbacks of this method are presented.

3.1 The reason of choice

As seen in the previous chapter, there have been various methods designed to support network mobility. These technologies all attempt to give solutions to the problem scope from different perspectives. Thus, every technology has certain advantages in the scenario it was designed for and may have weaknesses in an other network environment.

The Prefix Scope Binding Update approach, for instance, is explicitly designed for a limited scenario [D07]. The mobile network is not multi-homed, i.e. it attaches to the Internet through only one mobile router, and the mobile router has only one egress interface. Another problem is that only local fixed nodes in the mobile network are considered. The Internet Draft does not address problems related to mobile nodes. Nesting of mobile networks is also prohibited. These restrictions result in a very simple mobile network, consisting only of one mobile router, and having only one direct connection to the Internet. It is commonly agreed that there are application fields for such a mobile network, but a more general approach would be very desirable. In principle, the approach could be extended to multi-homed networks, and support for visiting mobile nodes could be added, but for this further research is necessary.

Another technology presented before, the Hierarchical Mobile IPv6 is not yet a solution for network mobility itself, however, it is a good tool for avoiding the binding update storm in bigger network scenarios and to improve mobile nodes' handoff speed between foreign networks.

Among all these possibilities, the MRHA proposal for network mobility seems to serve as the most adequate solution for the needs while managing mobility of entire networks. This technology can handle local fixed nodes, visiting mobile nodes, and also nested network scenarios. Because of these advantages the IETF's[1] NEMO Working Group[2] has chosen the MRHA concept as the solution for managing network mobility in their related projects.

3.2 Drawbacks

In spite of the advantages of the capabilities of the MRHA concept in handling of movements of mobile networks, there are also serious drawbacks that must be considered. These arising disadvantages are:

- *Thick tunnels*: This problem scope was previously presented. Thick tunneling occurs when a packet must be encapsulated and tunneled multiple times. Several levels of mobile networks induce excessive tunneling that can lead to serious packet loss and worsen stack behavior due to frequent packet fragmentation and reassembly. This is especially true in case of wireless environments.
- *Crossover tunneling* occurs when the path between a tunnel's endpoints contains only

[1] Internet Engineering Task Force

[2] Network Mobility Working Group

one endpoint of a tunnel that is conceptionally inside the other one. This problem typically arises when a home agent entity is deployed inside a mobile network. Figure 3.1 shows an example for this scenario.

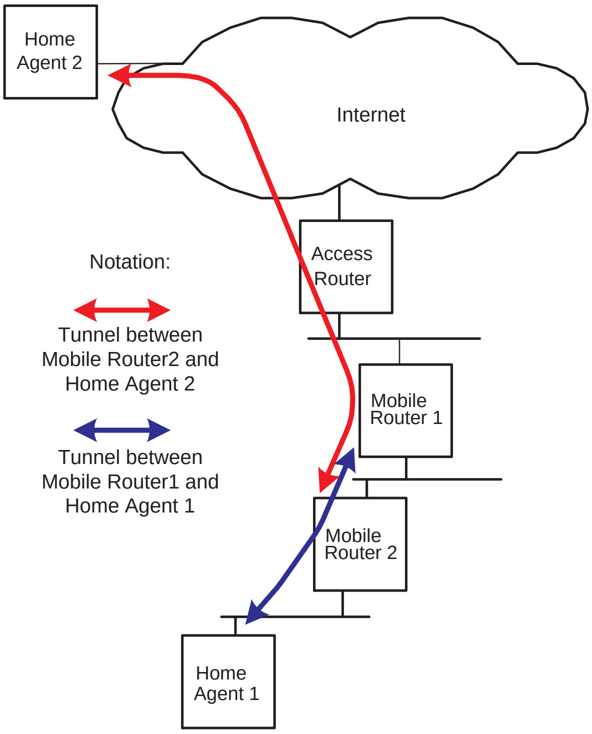
Home Agent 2
Internet
Notation:
Access Router
Tunnel between Mobile Router2 and Home Agent 2
Mobile Router 1
Tunnel between Mobile Router1 and Home Agent 1
Mobile Router 2
Home Agent 1

Figure 3.1: Crossover tunnel phenomenon. The red tunnel should conceptionnaly contain the entire blue tunnel. In this scenario the desired tunnel setup procedure is impossible to be performed.

- *Externally influenced internal communication:* if a mobile node and a local fixed node are in the same foreign network, their internal communication is also affected by the link state between their common mobile router and access router since – because of the tunneling method – the packets are routed towards the home agent of the mobile node, thus, the packets leave the foreign network. If the communication between the mobile router and the access router is lost, no matter how phisically close the mobile node and the local fixed node might be, they cannot communicate with each other.
- *Asymmetric and under-optimal communication paths:* outgoing communication paths

have different lengths than incoming communication paths, between the same two entities. This phenomenon (the so called ,,triangle routing problem") has already been presented and illustrated (see figure 1.1). An other typical behavior of packets is that due to multiple packet encapsulation they are forwarded via a ,,pinball-shaped" route as depicted on figure 2.8. However, these problems can be (and also desired to be) handled by means of routing optimisation techniques.

Chapter 4

BCMP – a micronet mobility protocol

When a mobile node in a moving network connects to a foreign network it is under the supervision of a mobile router, or in case of nested topology, several mobile routers. Since it is a mobile node, it may decide to change its point of attachment to this foreign network regularly, without even leaving the domain of the same mobile router. Even in this case it is desired to maintain the mobile node's continuous connection sessions to its correspondent nodes. This field is often referred to as micro-mobility or intra-domain mobility and there are various solutions suggested in this topic. These solutions mostly differ in the methods of address assingment and the advertisement of binding update messages. In the following section I will give a short overview of some of the existing proposals regarding mobile nodes' micro-mobility.

According to the **Mobile IPv6** solution, each time the mobile node changes its point of attachment to the network, a new care-of-address must be obtained and the node's home agent must be informed by means of a binding update message. In case of many mobile nodes and correspondent nodes this could lead to binding update storm which causes significant load on the air interfaces.

The **Hierarchical Mobile IPv6** introduces a new network element, the mobility anchor point, and separates on-link and regional care-of-addresses. With this extension of the Mobile IPv6 protocol the binding update messages are only needed to be sent to the mobility anchor point, thus to an entity lower in the hierarchy. This method reduces the round-trip-time of the binding update and the acknowledgement. Still, each time the mobile node performs a handoff, a new care-of-address must be obtained, and again, under certain circumstances, this can lead to binding update storm.

4.1 Overview of the BCMP concept

The **BRAIN Candidate Micronet Protocol (BCMP)** [MIND] is a micro-mobility protocol and was designed by the members of the IST's (Information Society Technologies) BRAIN project (Broadband Radio Access for IP based Networks). The BCMP technology can be applied in a network scenario where the mobile nodes are in connection with access routers, which are under the supervision of an anchor point. Figure 4.1 shows a typical BCMP topology in a foreign network. As seen, in the BCMP terminology access routers are not the routers that connect the mobile routers to the Internet but the ones that provide wireless connections for the mobile nodes inside the BCMP network. This is only an inconsistency in the terminologies of different technologies, thus, the entities referred to as access routers so far and in this section are separate devices. The mobile router's task is unchanged compared to previous scenarios since it is not needed for a mobile router to have BCMP capabilities. The anchor points' task is to tunnel data packets towards the mobile nodes and to participate in the address assignment procedure inside the foreign network. Since the access routers are the direct connection points for the mobile nodes, it is their responsibility to supervise handover when a mobile node decides to disconnect from its current access router and establishes connection to an other one. The user registry's task is to handle the mobile node's login into the mobile network.

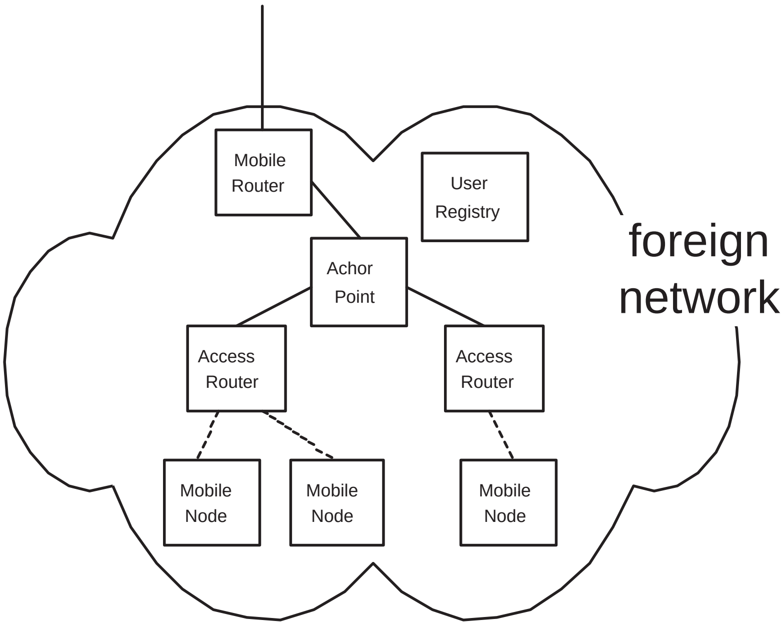

Figure 4.1: BCMP topology in a foreign network

4.2 Address assignment

When the mobile node enters the foreign network, it first starts a login procedure with the user registry. After AAA checks and context creation a new care-of-address is allocated and forwarded to the mobile node. The mobile node informs its home agent about this new address by means of a binding update message. This step has to be done only once, when the mobile node first enters the foreign network. It is not needed to change care-of-address even when the mobile node moves from an access router to an other one, if the two access routers are in the same anchor point's domain. Thus, exact registration of current positions of mobile nodes inside the foreign network is desired. Hence, the anchor point maintains a table of care-of-addresses currently in use by mobile nodes together with the addresses of access routers so that the anchor point can decide towards which access router to send a packet destined to a mobile node's care-of-address. Thus, when for example a packet arrives from the Internet, it is forwarded to the anchor point. The anchor point looks up its table and checks the care-of-address –> access router associations. Then the anchor point encapsulates the packet and tunnels it towards the chosen access router.

When a handover occurs between access routers it is the access router's task to inform the anchor point to change its entry in the table.

The big advantage of BCMP according to other existing mobility proposals is that a new care-of-address must be obtained only when the mobile node changes anchor points. This event evidently does not occur at each and every handover, thus, signaling overhead caused by frequent binding update messages could be reduced.

4.3 Handover

In BCMP there are two ways to perform the handover when a mobile node changes access routers: a regular and a prepared way.

If the handover is not planned, for example the mobile node lost connection with the old access router, the mobile node first sends a check-in message to the new access router. Since the new access router cannot authenticate this message, it sends a query to the old access router. If the old access router recognizes the mobile node identified in the query message, it replies to the new access router with the mobile node's parameters and context. Knowing the context of mobile node, the new access router is able to acknowledge the mobile node's handoff. During this authentication process there may packets arrive to the old access router, which is unable to transmit them to the mobile node, simply because the connection between them does not exist any longer. Thus, these packets are stored in the old access router's buffer. If the handoff happens without errors, the new access router indicates to build a temporary tunnel between the old access router and the new access router in order to transmit the buffered packets to their recipient, the mobile node. In addition, the new access router sends a redirect message to the anchor point, causing the anchor point to change its ,,care-of-address –> old access router" entry to ,,care-of-address –> new access router".

If there is a situation when the mobile node is within reach with two access routers and the mobile node would like to handoff from the current one to the other one, a prepared handoff can be performed. In this scenario the current access router will be referred to as old access router and the new candidate access router as new access router. First the mobile node sends a handoff preparation signal to the old access router, causing the old access router to send the mobile node's attributes and context to the new access router. Upon receipt, the new access router replies to the old access router. This triggers an acknowledgement message from the old access router to the mobile node and a request for building a temporary tunnel between the old access router and the new access router. From this time on, packets destined to the mobile node via the old access router will not be posted to mobile node but to new access router through the temporary tunnel. At the time the mobile node checks in at new access router it delivers packets stored in its buffer to mobile node, builds the tunnel between the old access router and the new access router down and sends redirection message to the anchor point, causing the anchor point to change its „care-of-address –> old access router" entry to „care-of-address –> new access router".

Chapter 5

The interface between MRHA and BCMP

As described before, BCMP is a micronet mobility protocol, thus, it is responsible for handling the mobility of mobile nodes inside a smaller network, in our case, the mobile network. Having MRHA for mobility management on one side of the network and BCMP on the other requires a node that shares its functionalities to be able to connect these two sides. The goal is to create a node entity that will show MRHA capabilities towards the MRHA part of the network and will also have BCMP functionalities towards the mobile network. Thus, the interface between these two technologies can be implemented by creating a node that has capabilities of both technologies. In order to work out the cooperation between BCMP and MRHA it was needed to design an appropriate network scenario and to build a testbed to analyse the interactions in progress.

5.1 Testbed introduction

In order to make connection between the MRHA and the BCMP the scenario was designed to consist of two main network parts – one using only MRHA and the other one purely BCMP for handling node and network mobility. While preparing the scenario, I took into consideration the main purposes of the respective technologies. Namely, according to my observations, the MRHA concept is mainly functional for handling the mobility for entire networks, moving between wider areas, while BCMP was rather designed to handle the mobility in smaller scenarios, like buildings, vehicles, etc. Thus, the testbed was built in a hierarchical way, using BCMP for handling node mobility in a smaller network, while the movement of this network is maintained by using MRHA (see figure 5.1). The task of the mobile router is to provide an interface between the MRHA and the BCMP part of the network, thus, to fulfil MRHA and BCMP capabilities at the same time.

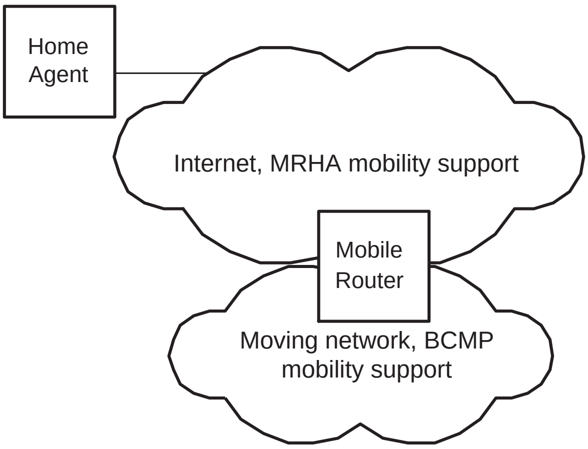
Home Agent
Internet, MRHA mobility support
Mobile Router
Moving network, BCMP mobility support

Figure 5.1: BCMP handles mobility inside the moving network, while MRHA maintains the mobility of the Mobile Router's network

Testbed elements

Both the MRHA and the BCMP part of the network consist of three separate entities. The MRHA part has a home agent and two access routers, while the BCMP part has two other access routers with BCMP capabilities and one equipment with both regular mobile router and BCMP anchor point functionality. Besides these elements there is also a mobile node and a correspondent node present.

Except of the mobile nodes, all the testbed entities are desktop PCs with AMD Athlon XP 1800+ processors, 512 MB RAM and 40 GB HDD. The network connection is provided via Intel Ethernet cards and, where wireless connection is used, via Avaya 802.11 PCM-

CIA wireless network adapters. The mobile node and the correspondent node are ASUS laptops with Intel processors. The mobile node also has an Avaya 802.11 PCMCIA wireless adapter. All the entities run Debian Linux Sarge (testing) operating systems compiled with 2.4.20 kernel and patched with the latest (0.9.5.1) MIPL patch in order to support IPv6 network mobility. The basic kernel configurations for all the nodes are quite similar, except for the home agent. Before compiling the kernel with the MIPL patch, there is an option to be selected according to the role of the given entity in a mobile network. MIPL supports both home agent and mobile node functionality, but at a time only one of them can be selected. Thus, all the entities in the testbed were compiled with mobile node functionality, except for the home agent, which was configured with home agent role support. The 802.11 adapters were not installed by means of the kernel's PCMCIA basic package but with the package *pcmcia-cs (Card Services)* to gain Demo Ad-Hoc functionality for the wireless cards.

Beside the desktop PCs and the laptops there was also a 3COM OfficeConnect dual speed hub used in order to create the home network (Ethernet LAN) environment.

Hence the MRHA technology is based on the IPv6 mobility draft [MobileIPv6], througout the design and implementation IPv6 addresses, signalling and tools were used. IPv4 was only used for early network configuration testing.

5.2 MRHA part of the testbed

The first step to build the scenario was to create and test the MRHA part of the testbed. The initial setting of such an environment is that the mobile router (MR) is at its home network, thus, located in the same network together with its home agent (HA). The home agent and the mobile router have been connected by means of straight UTP (Unscreened Twisted Pair) cables via the 3COM hub in order to represent an Ethernet LAN. The home network has been given IPv6 subnet mask 2002:0:0:a::/64, the home agent has been provided the address 2002:0:0:a::1, while mobile router the address 2002:0:0:a::2 (see figure 5.2). As seen, the "2002:0:0:a" parts of the addresses describe the home network.

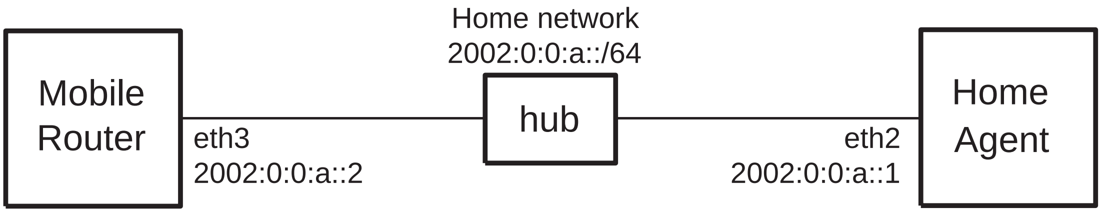

Figure 5.2: Home network part of the testbed

In order to demonstrate the movement of the nodes, there was a need to introduce the scenario of foreign networks to the testbed. Thus, an access router (AR1) has been installed to the network, representing the access point for mobile nodes to a foreign network. The access router has been directly connected with the home agent by means of a cross UTP cable via interfaces 2002:0:0:1::2 and 2002:0:0:1::1, respectively. Although the two entities are for simplicity physically connected, this wire can be substituted to (and also represents) any arbitrary path on the Internet. The foreign network has been given the netmask 2002:0:0:5::/64 and the access router's interface towards the foreign network has been provided the address 2002:0:0:5::2. This latter interface is a wireless one, in order to grant connection for mobile entities to the foreign network.

In order to analyse behaviour of mobile networks while changing connections to the Internet, another foreign network has been introduced to the testbed, in a quite similar way as described previously. The point of attachment to this another foreign network is also an access router (AR2). The only difference between AR1 and AR2 are the addresses assigned to the interfaces involved. AR2 is also directly connected to the home agent by means of a cross UTP cable via interfaces 2002:0:0:2::1 and 2002:0:0:2::2, respectively. This second foreign network has been given the netmask 2002:0:0:6::/64 and AR2's wireless interface towards this foreign network has been provided the address 2002:0:0:6::2. The initial setting of the MRHA part of the testbed with the mobile router being at its home network can be seen on figure 5.3.

With the scenario built and the addresses of the interfaces set, the next step was to configure certain parameters of the entities in order to function according to their roles.

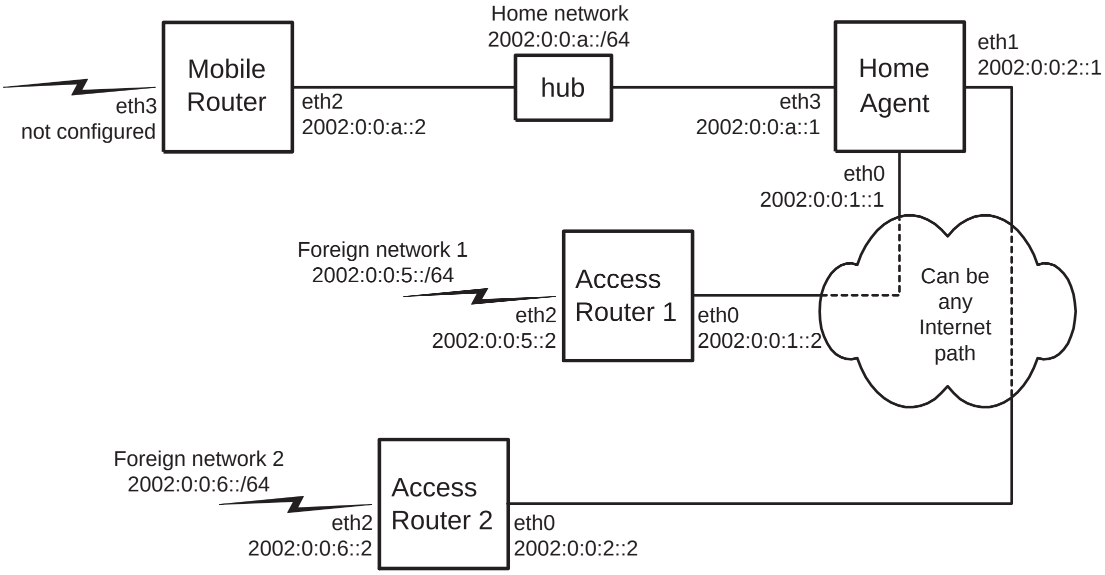

Figure 5.3: Initial setting of MRHA part of the testbed with the Mobile Router at home

The home agent has got to advertise its services on the home network so that the other hosts on the network can make use of its capabilities. The program called *radvd* can be used to advertise certain pieces of information on a network, thus, it was ideal for the purpose. This program was found to be particularly configurable by means of the file */etc/radvd.conf*. The content of this file set on the home agent is the following:

```
interface eth3
{
   AdvSendAdvert on;
   AdvHomeAgentFlag on;
   prefix 2002:0:0:a::1/64
   {
        AdvRouterAddr on;
   };
};
```

The program *radvd* run with this configuration file[1] periodically sends router advertisement messages on the home network telling that on interface eth3 with address

[1] There are several other optional parameters of *radvd* (eg. message sending period). Leaving them out from */etc/radvd.conf* the program runs with their default values.

2002:0:0:a::1 a home agent can be found.

When the mobile router leaves the home network and wants to connect to a foreign one, it finds the point of attachment also by means of router advertisement messages. Thus, the access points in the foreign networks also have to run the program *radvd*. This time however, the parameters must be set differently, since the purpose of the advertisement here is not broadcasting home agent capabilities, but to share the information with the mobile router, that there is a foreign network nearby with a certain network prefix. The */etc/radvd.conf* file of Access Router 1 is the following:

```
interface eth2
{
   AdvSendAdvert on;
   prefix 2002:0:0:5::2/64
   {
        AdvRouterAddr off;
   };
};
```

Finally, there were parameters to be configured also on the mobile router, in order to provide the desired relationship between itself and its home agent. In the file */etc/network-mip6.conf* the following values have been set:

```
FUNCTIONALITY=mn
HOMEADDRESS=2002:0:0:a::2/64
HOMEAGENT=2002:0:0:a::1/64
```

As seen in the configuration file above, ,,mobile node" functionality was chosen. The reason of this choice is that in the plain Mobile IPv6 protocol there is no functionality as ,,mobile router", thus, among the available options (mobile node, correspondent node, home agent), the most similar functionality seemed to be the most reasonable choice. As we will see, the design and implementation of a mobile router is as a matter of fact an extension of a mobile node with routing capabilities and certain functionalities that can provide the mobile router the information needed for maintaining a mobile network. At this point our mobile router has got no routing capabilities yet, but as a mobile node it is ready to test the functionality of the MRHA technology when the router moves from its home network to different foreign networks. The scenario was the following: the mobile router was at home, in the same network with its home agent, which was periodically sending home agent advertisement messages. Initially none of the access routers were advertising foreign network prefix information. I manually disconnected the mobile router from the home network by unplugging the UTP cable from its eth2 interface. I ran *radvd* on the first access router to start the advertisement of the first foreign network prefix, 2002:0:0:5::/64. The mobile router noticed the presence of a nearby foreign network, and thus its wireless interface gained a new care-of-address, namely 2002::5:202:2dff:fe42:d569. This address is automatically generated by combining the received foreign network prefix and the PCMCIA wireless network adapter's hardware address. After the care-of-address is assigned, the mobile router automatically sends a binding update message to its home agent and builds a tunnel between itself and the home agent. The tunnel can be represented as an interface (with name *ip6tnl1*) and it also shows up when using the *ifconfig* command and in the routing table as the default gateway (see *route -A inet6*). The current states of the MRHA tunnel and the mobile router's binding update list are the following:

```
MR# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002:0:0:a::1 \
                   local 2002::5:202:2dff:fe42:d569 \
                   hoplimit 255 flags ELKM

MR# mipdiag -l
Mobile IPv6 Binding update list
Recipient CN: 2002:0:0:a::1
BINDING home address: 2002:0:0:a::2 \
        care-of address: 2002::5:202:2dff:fe42:d569
        expires: 129 sequence: 10 state: 1
        delay: 1 max delay 256 callback time: 82
```

Upon the receipt of the binding update message, the home agent creates an entry in its binding cache assigning the mobile router's home address with its care-of-address and creates a tunnel towards it. This reverse tunnel, similarly to the tunnel created by the mobile router, can be represented by an interface (also with name *ip6tnl1*), and it shows up in the routing table as the interface to be used for packets destined towards the mobile router. The current states of the MRHA reverse tunnel and the home agent's binding cache are the following:

```
HA# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002::5:202:2dff:fe42:d569 \
                   local 2002:0:0:a::1 \
                   hoplimit 255 flags ELKM

HA# mipdiag -c
Mobile IPv6 Binding cache
Home Address 2002:0:0:a::2
Care-of Address 2002::5:202:2dff:fe42:d569
Lifetime 867
Type 2
```

The mobile router's disconnection from a foreign network can be triggered by several reasons. One of them is that the mobile router loses contact with its access router, thus, its point of attachment to the foreign network. If there is another foreign network's access router nearby, handoff to that foreign network can be performed. Otherwise, the mobile router – and thus its moving network – loses contact with the Internet. In my scenario I demostrated the mobile router's loss of contact with the first access router by pulling down AR1's wireless interface. At this point none of the access routers were functioning, thus, the mobile router completely lost the connection with the Internet. Next I pulled up the second access router's wireless interface and started the advertisement of the new network prefix (2002:0:0:6::/64). Upon receipt of the advertising packets, the mobile router gained another care-of-address (2002::6:202:2dff:fe42:d569) with the new network prefix. The steps followed by the mobile router and the home agent are similar to the case when the mobile router joined the first foreign network: MRHA and reverse MRHA tunnels are reconfigured, the binding update list, the binding cache and the routing tables are automatically refreshed according to the new address information. With a graphical user interface I created using the combination of shell scripts, C code and TCL/TK the functionality of the access routers' wireless interfaces can be switched on and off without being at the access routers, thus, the simulation of the handoff can be triggered remotely.

5.3 Providing routing capabilities for the mobile router

At this time we have a mobile node which is – according to its functionality – an endpoint in the network, it does not forward IP packets yet, it only sends and receives them. This behaviour of network elements can be changed by setting the *forwarding*[2] value to 1 or 0 whether allowing IPv6 packets to be forwarded between interfaces or not, respectively. For mobile nodes this value is set to 0 by default. However, in the case of a mobile router it is cruical to set this variable to 1 since setting it to 0 prevents the mobile router to forward packages and thus, to fulfil its functionality. After the appropriate setting it is expected that the mobile router behaves the same way as in the experiment above but it is also able to pass IPv6 packets between its interfaces. However, my experiments showed that the mobile router with the new setting failed to follow the steps of the Mobile IPv6 protocol, namely it refused to gain a new care-of-address when – after leaving its home network – a foreign network's access router was nearby. This problem itself lies in the source code of the Mobile IPv6 patch in the kernel. From the kernel source dictionary, in the file *net/ipv6/ndisc.c* the following sample of code can be found:

```
static void ndisc_router_discovery(struct sk_buff *skb)
{
...
if (in6_dev->cnf.forwarding || !in6_dev->cnf.accept_ra) {
          in6_dev_put(in6_dev);
                  return;
        }
...
}
```

According to the source code sample if he variable *forwarding* is set to 1 the function *ndisc_router_discovery()* returns before the mobile router can be provided with a new care-of-address with the foreign network's prefix. The reason of this solution in the Mobile IPv6 source code might have been inspired by the assumption that routers de not move and thus, they do not need to accept routing advertisement messages. However, in our case it is necessary. To avoid this pitfall, the kernel source code had to be modified by removing

[2] It can be found and modified at */proc/sys/net/ipv6/conf/all/forwarding*

the step of examining the value of the variable *forwarding.* After recompiling the kernel the problem of discarding the router advertisement messages was solved.

5.4 Preserving the mobile network prefix

Since our mobile router's task is to ensure connection to the Internet for its mobile nodes, it is necessary to make a routing entry in the mobile router's routing table pointing towards them in order to ensure the delivery of the packages destined to the nodes in the mobile network. In the testbed the mobile network was given the prefix 2002:0:1::/48. By registering this address to the routing table manually the forwarding of the appropriate packages towards the mobile network is ensured. However, when launching the Mobile IPv6 extension by running the command */etc/init.d/mobile-ip6 start* all the IPv6 routes set previously become lost, and the routing table is rebuilt according to the Mobile IPv6 local settings and information packages (eg. routing advertisements). The same thing happens at the mobile router's each and every handoff and mobility event. Thus, the mobile network prefix must be preserved when the routing table is cleaned up.

First, the desired network address must be introduced to the kernel. When the Mobile IPv6 protocol is started, the program called *mipdiag* is run and it passes variables and settings to the kernel, which can be manually set in the file named */etc/network-mip6.conf.* I added the entry ,,NETWORKADDRESS=2002:0:1::/48” to this file. In the next step I extended the source code *mipdiag.c.* First I added commands to read the network prefix information from the configuration file and extended the function *rtn_set_mn_info().* With this modification the function not only passes the home agent's address and the home address of the mobile router for the kernel but also the given network prefix. The code sample of the required modifications is the following (I declared the flags ,,IFA_NPREFIX” and ,,IFA_NADDRESS” in the appropriate place previously):

```
static struct option long_options[] =
    {
        ...
        {"networkaddress", 2, 0, 'N'},
        ...
    };

c = getopt_long (argc, argv, "i:IP:mslc?Vd::t::H::h:N::",
                 long_options, &option_index);

int rtn_set_mn_info(int ifindex, struct in6_ifreq *home,
                    struct in6_ifreq *ha, struct in6_ifreq *na)
    // this last parameter is the network address
{
    ...
    addattr_l(&req.n, sizeof(req), IFA_NPREFIX, &na->ifr6_prefixlen,
              sizeof(u_int32_t));
    addattr_l(&req.n, sizeof(req), IFA_NADDRESS, &na->ifr6_addr,
              sizeof(struct in6_addr));
    ...
    if (rtnl_talk(&rth, &req.n, 0, 0, NULL, NULL, NULL) < 0)
        return -2;
}
```

Now that the parameters stored in the configuration file are read and sent to the kernel by means of the function *rtnl_talk()*, the information must be sent to its appropriate location. The variables given by *mipdiag* are next read by function *inet6_rtm_newaddr()* in file *addrconf.c*. There were modifications needed also in this function to read our variables as well.

```
inet6_rtm_newaddr(struct sk_buff *skb, struct nlmsghdr *nlh, void *arg)
{
    ...
    if (rta[IFA_NPREFIX-1])
    {
        if (pfx == NULL || !(ifm->ifa_flags & IFA_F_HOMEADDR))
            return -EINVAL;
        if (RTA_PAYLOAD(rta[IFA_NPREFIX-1]) < sizeof(nprefix))
            return -EINVAL;
        nprefix = *(u_int32_t*)RTA_DATA(rta[IFA_NPREFIX-1]);
    }
    if (rta[IFA_NADDRESS-1])
    {
        struct in6_addr *na;
        if (pfx == NULL || !(ifm->ifa_flags & IFA_F_HOMEADDR))
            return -EINVAL;
        if (RTA_PAYLOAD(rta[IFA_NADDRESS-1]) < sizeof(*na))
```

```
            return -EINVAL;
        na = RTA_DATA(rta[IFA_NADDRESS-1]);
        addrconf_set_mipv6_mn_network_address(na, nprefix);
    }
    ...
}
```

The function *addrconf_set_mipv6_mn_network_address(na, nprefix)* is declared in the file *mipglue.h* which is responsible for the mobility integration into the plain IPv6 protocol. The modifications needed in this file are the following:

```
static inline void
addrconf_set_mipv6_mn_network_address(struct in6_addr *networkaddress,
                                      u_int32_t nprefix)
{
    MIPV6_CALLPROC(mipv6_set_network_address)(networkaddress, nprefix);
}
/* pointers to mipv6 callable functions */
struct mipv6_callable_functions
{
    ...
    void (*mipv6_set_network_address)(struct in6_addr *home_addr,
                                      u_int32_t n_prefix);
    ...
};
```

And finally, our network prefix can be read by the file *mn.c*, which configures the initial settings of a mobile node, including the parameters given by us. The extensions to this source code are the following:

```
int __init mipv6_mn_init(void)
{
    ...
    MIPV6_SETCALL(mipv6_set_network_address,
                  mipv6_mn_set_network_address);
    ...
}

void mipv6_mn_set_network_address(struct in6_addr *naddr,
                                  u_int32_t nprefix)
{
    networkaddress=(*naddr);
    networkprefix=nprefix;
}
```

As mentioned before, at every mobility event the Mobile IPv6 protocol cleans up the whole IPv6 routing table and reconfigures it. With a network prefix associated with the mobile router, the protocol can now be informed about the network that lies behind the mobile router and thus, the cleanup of the routes pointing towards this mobile network can be prevented. The mobility detection of the protocol is the responsibility of the file *mdetect.c*. In this source we can find a function that is responsible for the cleanup of the routing table. The function does not delete every route, it first performs a check to each and every route in the routing table, whether they can be deleted or not. In previous cases, this was the point where our manually configured mobile network address passed the test, thus, it became deleted. This time – with some extensions – we can make this test fail for our network address. This addition is based on a prefix check: if the address currently under examination is the same as our network address for the prefix length determined by us (in the configuration file), then the route fails the test, thus, it will not be deleted. The modification of *mdetect.c* can be seen in the following code sample:

```
if ((type & (IPV6_ADDR_MULTICAST | IPV6_ADDR_LINKLOCAL)) ||
     rt->rt6i_dev == &loopback_dev || rtr_is_gw(rtr, rt) ||
     // additional prefix check here
     mipv6_prefix_compare16(&rt->rt6i_dst.addr, na, (int)networkprefix) ||
     is_prefix_route(rtr, rt) || (rt->rt6i_flags & RTF_DEFAULT))
        ret = 0;
```

We can see that if the current address has the same prefix as our network address, then the test returns value 0, thus, the route to this address will not be deleted. In the original code the *mipv6_prefix_compare16()* function was not implemented, it is actually a modification of the kernel's *mipv6_prefix_compare()* function. The reason for writing an additional prefix checker function was that the built-in one checks prefixes for every 32 bits, not for every 16 bits, as needed in our case. The source code of this new prefix checker function is the following:

```
int mipv6_prefix_compare16(struct in6_addr *addr,
                           struct in6_addr *prefix, unsigned int nprefix)
{
    int i;

    if (nprefix > 128)
        return 0;

    for (i = 0; nprefix > 0; nprefix -= 16, i++) {
        if (nprefix >= 16) {
            if (addr->s6_addr16[i] != prefix->s6_addr16[i])
        return 0;
        } else {
            if (((addr->s6_addr16[i] ^ prefix->s6_addr16[i]) &
                ((~0) << (16 - nprefix))) != 0)
                return 0;
            return 1;
        }
    }
return 1;
}
```

With all these modifications described above, it is now possible to set up a permanent route in the mobile router towards an arbitrary network. After recompiling the kernel I tested the system by adding „NETWORKADDRESS=2002:0:1::/48" to */etc/network-mip6.conf* and determining some routes towards this network. After several handoffs between access routers – and hence several automatic routing table cleanups – I checked the routing table and found that the routes I manually set are still in the routing table – on the contrary to previous cases, when I used the plain Mobile IPv6 kernel without modifications.

```
MR# route -A inet6
Destination        Next Hop  Flags Metric Ref Use Iface
::1/128            ::        U     0      5     0 lo
...
2002:0:1:3::/64    ::        U     1      16    0 eth0
2002:0:1:4::/64    ::        U     1      16    0 eth1
```

Thus, the mobile router is now ready to route incoming packets towards the mobile network.

5.5 Making global routing available

Before considering micro-mobility management in the mobile network, I first set up the access routers of the mobile network for routing test purposes. I also introduced a correspondent node to the network. The current topology and the address assignments can be seen on figure 5.4.

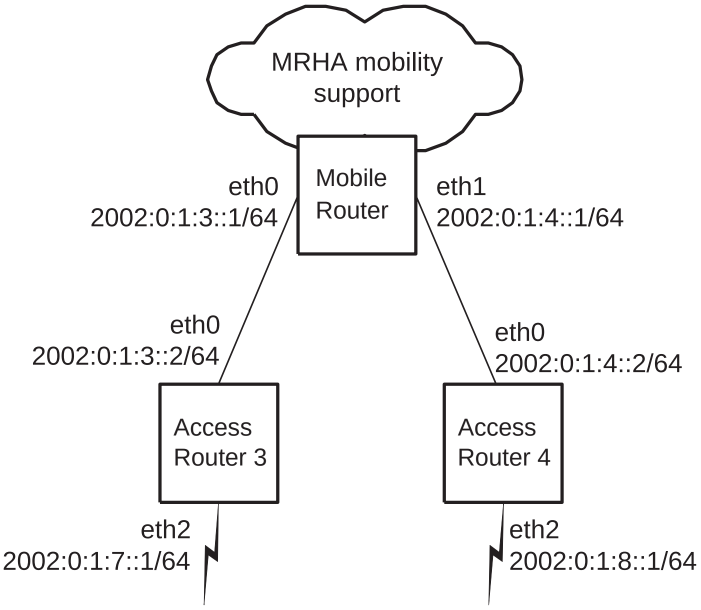

Figure 5.4: Topology and address assignments for testing of the mobile router's routing capabilities

In this case mobility management is not tested, thus, the location of the mobile router – whether being at its home network or in a foreign network – is regardless. While testing, the correspondent node pinged different interfaces in the network. In the first stage the correspondent node pinged the mobile router's eth0 interface (2002:0:1:3::1/64) successfully. However, when the correspondent node pinged the access router's eth0 interface (2002:0:1:3::2/64), there was no response – the ping failed. I traced the route of the ping packets from interface to interface and I found that though the access router 3 gets the request and replies normally, the mobile router drops the reply messages instead of forwarding them towards the correspondent node. Checking the routing table of the mobile router, the result was:

```
MR# route -A inet6
Destination  Next Hop                 Flags Metric Ref    Use Iface
::/0         ::                       U     0      5        0 ip6tnl1
...
::/0         fe80::202:2dff:fe42:d578 UGDA  1024   512939   0 eth3
```

According to the first experiment, when the correspondent node pinged one of the mobile router's interfaces, and also according to the routing table everything seems to be normal, the mobile router should forward the ping reply packets towards the correspondent node. The solution for this problem lies in the route flags. The flags seen in the routing table mean:

- U - Route is up
- G - Route is a gateway
- D - Route was dynamically installed by daemon or redirect
- A - Route was installed by addrconf

The flags that cause the problem are the D and A flags. Removing them from the route entry serves a good solution. However, when a mobility event occurs, the routing table gets cleaned up and the routes are reconfigured, the entry causing the problem is automatically created again with the same (UGDA) flags. To avoid this, the kernel source code was needed to be modified again. The function that is responsible for creating default routers and distibuting them with the flags is *rt6_add_dflt_router()* in file *route.c*. Modifying this function the flags that cause the problem are no more associated with the default gateway:

```
struct rt6_info *rt6_add_dflt_router(struct in6_addr *gwaddr,
                                     struct net_device *dev)
{
    struct in6_rtmsg rtmsg;
    memset(&rtmsg, 0, sizeof(struct in6_rtmsg));
    rtmsg.rtmsg_type = RTMSG_NEWROUTE;
    ipv6_addr_copy(&rtmsg.rtmsg_gateway, gwaddr);
    rtmsg.rtmsg_metric = 1024;
// old flags
//  rtmsg.rtmsg_flags = RTF_GATEWAY | RTF_ADDRCONF | RTF_DEFAULT | RTF_UP;
// new flags
    rtmsg.rtmsg_flags = RTF_GATEWAY | RTF_UP;
    rtmsg.rtmsg_ifindex = dev->ifindex;
    ip6_route_add(&rtmsg);
    return rt6_get_dflt_router(gwaddr, dev);
}
```

Checking the routes again using the modified kernel, but under the same circumstances as previously, the output is the following:

```
MR# route -A inet6
Destination Next Hop                 Flags Metric Ref    Use Iface
::/0        ::                       U     0      5        0 ip6tnl1
...
::/0        fe80::202:2dff:fe42:d578 UG    1024   512939   0 eth3
```

It can be seen that the flags are now set properly and according to the tests the routing functionality of the mobile router is adequate – the ping from the correspondent node to any interface in the network was successful.

5.6 Maintaining routes on the home agent

While designing an MRHA scenario, besides the extensions of the mobile router, a minor modification also in the functionality of the home agent must be made. Additions must be applied in the home agent's routing table maintenance towards the mobile network's prefix. Initially, when the mobile router is located in its home network, a routing entry can be added to the home agent pointing towards the mobile network determining the mobile router's home address as the next hop. However, when the mobile router leaves its home network and joins a foreign one, the routing entry containing the mobile router's home address becomes invalid. The mobility event itself does not trigger the correction of this route automatically – on the contrary to the correction of the route towards the mobile router itself which is triggered by the binding update message sent by the mobile router. This implies that the state of the mobile router must be checked regularly and the routing table of the home agent must be maintained accordingly. The two events when changes must be made are when the mobile router leaves the home network and when it returns to it. Initially the routing table of the home agent shows the following:

```
HA# route -A inet6
Destination    Next Hop                 Flags Metric Ref    Use Iface
...
2002:0:1::/48  2002:0:0:a::2            UG    1      0        0 eth3
```

While the mobile router moving away, the address determined as the next hop (2002:0:0:a::2) becomes invalid and sending the packet via interface eth2 also becomes void. Thus, the routing entry towards the prefix 2002:0:1::/48 must be changed, and since the mobile router is currently reachable via the tunnel interface denoted as *ip6tnl1*, the appropriate commands must be applied:

```
HA# route -A inet6 del 2002:0:1::/48 gw 2002:0:0:a::2
HA# route -A inet6 add 2002:0:1::/48 dev ip6tnl1
HA# route -A inet6
Destination    Next Hop                 Flags Metric Ref    Use Iface
...
2002:0:1::/48  ::                       U     1      15170   -3 ip6tnl1
```

The routing table now has an up-to-date and valid entry towards the mobile network. Similar steps must be applied when the mobile router returns to the home network – the route via the tunnel must be removed and the gateway entry to the home address of the mobile router must be restored. These mobility events – leaving and returning to the home network – can be tracked by regularly checking the state of the tunnel interface. Change in the tunnel preferences means a mobility event. Examples for the states of the tunnel interface are the following:

```
// mobile router being at home
HA# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote :: \
                   local :: \
                   hoplimit 255 flags K
// mobile router connects to a foreign network via an access router
HA# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002:0:0:a::1 \
                   local 2002::5:202:2dff:fe42:d569 \
                   hoplimit 255 flags ELKM
```

With regular checking of the tunnel's state the routing table can be maintained by deleting the invalid entries and adding up-to-date ones.

5.7 BCMP part of the testbed

Having a network part ready for MRHA mobility management the other part of the network can be installed. As mentioned before, BCMP is a micronet mobility protocol and will maintain the mobility management in the mobile network part of the scenario. In this testbed the BCMP network consists of a user registry, an anchor point, two access routers and a mobile node. The mobile router is the node that will have both MRHA and BCMP capabilities in order to connect the two network parts. The implementation of the MRHA part of the mobile router was detailed in the previous sections, in this section the installation of the BCMP part is presented.

The different BCMP nodes are theoretically separate entities with different tasks and responsibilities. However, since the entities are actually programs that can be run in user space, it is possible to integrate some of them in a way that they can run on the same machine at the same time. In this testbed I decided to integrate the mobile router, the user registry and the anchor point functionalities into one computer. The access routers are still separate devices since they are quite identical – both in their roles and configuration – and, even if we consider the system as a real network scenario, they must be located far from each other to provide accessibility for mobile nodes in as wide area as possible.

Thus, the mobile router entity will show MRHA capabilities towards the MRHA part of the network and will seem to be a BCMP anchor point and user registry in the BCMP access routers' point of view.

5.8 Configuration of the nodes

The BCMP part of the testbed was designed according to the topology shown previously on figure 5.4. The user registry and the anchor point modules were installed on the mobile router. Following the addressing convention I used during the implementation of the mobile router's route preserving functionality, I configured the address space 2002:0:1::/48 for advertisement by the user registry. The main parameters that can be manually configured and are used by both the user registry and the anchor point module can be found in the file *net.conf*. The main parameters are the following:

```
user_registry = 2002:0:1:3::1

anp_ip = 2002:0:1:3::1
anp_pool = 2002:0:1::/48

# Access Router 1
ar_ip = 2002:0:1:3::2

# Access Router 2
ar_ip = 2002:0:1:4::2
```

After the similar setting of these parameters on each node, the programs for the user registry and the anchor point could be started on the mobile router, while the access routers started the program written for access routers. In the testbed the mobile node arrives to the mobile network with its air interface unconfigured.

5.9 Tunneling interface collision

When a mobile node is in the BCMP network, it is under supervision of an access router. Naturally, in case of movement, the mobile node can change access routers within the network at any time. The current access router information for each mobile node is stored in the anchor point module, since it is responsible for directing the incoming traffic towards the current access router in order to reach the mobile node. The anchor point forwards the traffic by means of tunnels pointing from the anchor point to the access router. However, since in our case the mobile router is the node that has anchor point capabilities, the tunnel's endpoints are the mobile router and one of the two BCMP anchor points.

In the MRHA technology, if the mobile router is not located in its home network, a bidirectional tunnel is built between the mobile router and its home agent. To be more specific, on the mobile router's side a tunnel is created, for example:

```
MR# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002:0:0:a::1 \
                   local 2002::5:202:2dff:fe42:d569 \
                   hoplimit 255 flags ELKM
```

It is seen that the MRHA protocol automatically configures the tunnel interface *ip6tnl1*. However, the BCMP anchor point also starts to create tunnels towards the mobile network starting from *ip6tnl1*. If we installed the mobile router with BCMP support without MRHA capabilities, we would have a BCMP tunnel configuration like this:

```
MR# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002:0:1:7::1 \
                   local 2002:0:1:3::1\
                   hoplimit 255 flags EL
```

However, with both MRHA and BCMP support turned on, this concludes in an interface reserving collision, and the system becomes inoperatable. The reason of this reserving convention is that originally BCMP was written for a system that has BCMP capable devices separately, that is, user registry, anchor point, access routers are all run on separate computers. In our case however, the anchor point and the mobile router are located in the same device, thus, their independency also in their used resources must be assured. This problem can be solved by modifying the BCMP's tunnel reservation algorythm by specifying a tunnel interface sequence number big enough to avoid reservation collision with high probability. By determining an integer for the first tunnel interface to be used by BCMP, tunnels only with equal to or higher sequence numbers will be reserved towards the mobile network's access routers. Since MRHA only uses interface *ip6tnl1*, this sequence number can be set to 2 – representing *ip6tnl2* – but to avoid any possible future collisions it is suggested to choose a higher integer. In my system I chose sequence number 50 for BCMP tunnel reservations. Thus, the anchor point module will only reserve tunnels *ip6tnl50* and above towards the BCMP acess routers. Applying the modifications and enabling both MRHA and BCMP capabilities of the mobile router will result a similar tunnel setup if both the MRHA tunnel and a tunnel towards an access router in the mobile network is up:

```
// MRHA bidirectional tunnel
MR# ipv6tunnel show ip6tnl1
ip6tnl1: IPv6/IPv6 remote 2002:0:0:a::1 \
                   local 2002::5:202:2dff:fe42:d569 \
                   hoplimit 255 flags ELKM

// BCMP tunnel towards access router 3
MR# ipv6tunnel show ip6tnl50
ip6tnl1: IPv6/IPv6 remote 2002:0:1:7::1 \
                   local 2002:0:1:3::1\
                   hoplimit 255 flags EL
```

Chapter 6

Analysis of the system

By creating a mobile router that connects two different types of mobility handling technologies the testbed can be used to analyse the traffic performance of a mobile network in points of different types of handovers. The final topology of the scenario can be seen on figure 6.1. As seen on the picture, a correspondent node with an USB camera was introduced to the system. The data stream sent by the camera is addressed to the mobile node on the other end of the network, which is to display the received video. In average the data stream generates 360 kbit/sec UDP traffic in the network from the correspondent node towards the mobile node. During the test the mobile node will register to the mobile network and will perform handovers between BCMP anchor points no. 3 and 4. Meanwhile, the mobile router will move away from its home network, it will register to one of the foreign networks and will perform handovers between anchor points no. 1 and 2. The latencies caused by the different types of handoffs will be measured separately and while both the mobile router and the mobile node are changing anchor points.

The operation of the MRHA part of the network has been described in previous sections – the mobile router leaves its home network and connects to a foreign network. At this point a bidirectional tunnel is built between the mobile router and its home agent. If the mobile router changes foreign networks, the tunnel is reconfigured, and the current connection session is recovered. However, according to measurements, the recovery of

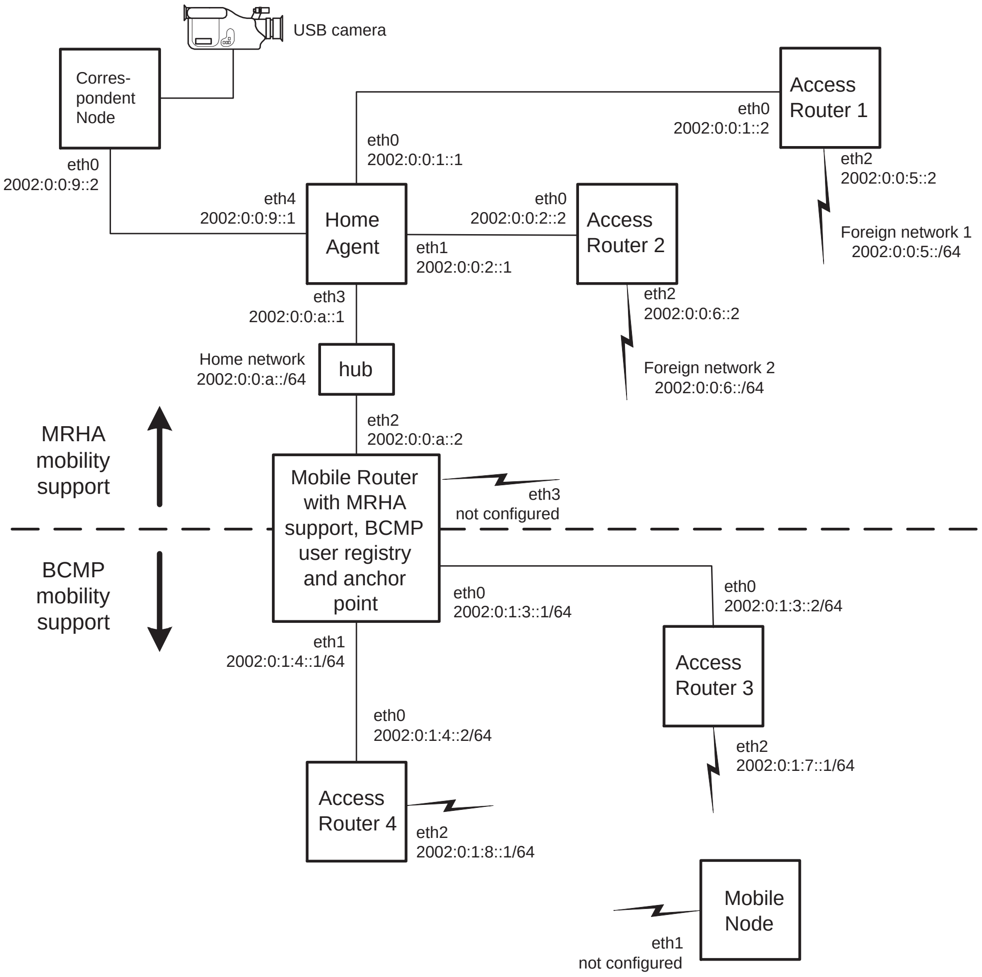

Figure 6.1: Final topology of the testbed with initial node configuration, mobile router at home

the session is rather slow, as seen on figure 6.2. This amount of latency can be very annoying while using real-time applications, like in our case. The video stream stopped for several seconds, and though the session had been recovered, real-time communication was disturbed. In order to improve the MRHA's handoff efficiency, further research is needed.

The packets arriving from the MRHA part of the network to the mobile router are simply rerouted towards the appropriate BCMP tunnel created by the anchor point module. Except for the encapsulation implied by the tunneling, no modifications are made to

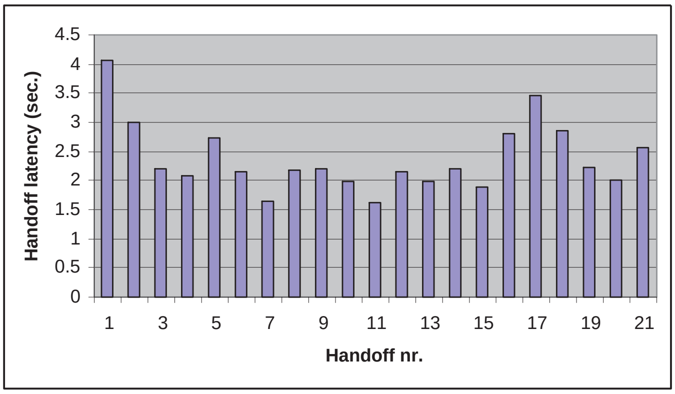

Figure 6.2: Handoff latencies when the mobile router switches access routers

the packets. Thus, no significant latency can be measured between the ingress and egress interfaces of the mobile router.

After the packets enter the BCMP part of the network, they arrive to an anchor point and are forwarded towards the mobile node. If a handoff inside the mobile network is performed, no modifications are made to the mobile node's care-of-address configured at the registration time into the mobile network. However, the appropriate tunnel in the mobile router must be reconfigured, which is done by the anchor point module. Measuring the latencies caused by intra-network movement handled by BCMP we get results seen on figure 6.3. While observing the video stream received on the client side, I found that

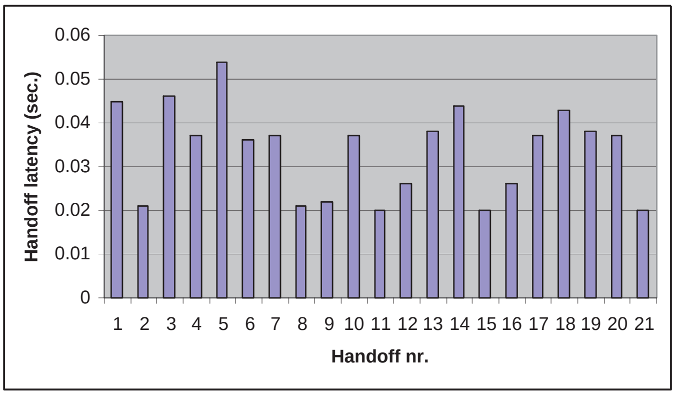

Figure 6.3: Handoff latencies when the mobile node switches BCMP access routers

the handoffs performed by the mobile node were absolutely seamless. The reason for this phenomenon is that when the mobile node does not perform handovers, the time intervals between the packets received are the same size as during hadovers. This implies that no packets were lost, thus, latencies in this experiment are not latencies, they are normal time intervals. The results of this specific experiment showed that the BCMP handovers of the mobile node do not influence the effective operability of such a real-time application.

Chapter 7

Conclusion

The technologies of MRHA tunneling and BCMP offer solutions for IP mobility from different aspects – the latter aims to provide seamless mobility for nodes moving inside a smaller network, while the former is a solution for maintaining the connection sessions for entire mobile networks. In order to interconnect these two technologies a device had to be designed and implemented that shares functionalities in a way that both mobility handling protocols can be supported. The interface providing connection for MRHA and BCMP is the mobile router, which – with certain modifications – can be provided with capabilities that are required for supporting the needs of both technologies.

In my thesis I gave an overview of the Mobile IP and the mobile network concepts, and I presented some possible solutions proposed by various task forces for handling the mobility of moving networks. I gave a detailed description of the MRHA concept and introduced the BCMP protocol that provides mobility management for mobile nodes in smaller networks. In the second part of my thesis I presented the configuration of a testbed that I installed for modeling a mobile network scenario and described the modifications that had to be made in the mobile router in order to support network mobility. These requirements were

- providing routing capabilities for the mobile router

- reserving a network address for the mobile network
- assuring global routing
- maintaining routing table entries in the home agent.

After the presentation of the solution addressing the tasks listed above I configured the BCMP part of the network and made modifications in the BCMP anchor point module in order to avoid the collision at tunnel reservation. Finally I analysed the system by measuring the latencies caused by different types of handoffs performed by both the mobile router and the mobile node. The tests showed that the mobile router is able to interconnect the two different mobility handling technologies, MRHA and BCMP.

Bibliography

[MobileIP] Pravin Bhagwat, Charles Perkins, Satish Tripathi: *,,Network Layer Mobility: an Architecture and Survey"*

[MobileIPWeb] Charles Perkins: *,,Mobile Networking Through Mobile IP"*, http://www.computer.org/internet/v2n1/perkins.html

[MobileIPTerm] Charles Perkins: *,,Mobile Networking Terminology"*, http://www.computer.org/internet/v2n1/terms.html

[IPv6] S. Deering, R. Hinden: *,,Internet Protocol, Version 6 (IPv6) Specification"*, RFC 2460, December 1998.

[MobileIPv6] David B. Johnson, Charles E. Perkins, Jari Arkko: *,,Mobility Support in IPv6"*, Internet Draft, June 2002.

[MoNetTerm] Thierry Ernst, Hong-Yon Lach: *,,Network Mobility Support Terminology"*, Internet Draft, November 2002.

[PrefixScope] Thierry Ernst, Alexis Olivereau, Ludovic Bellier, Claude Castelluccia, Hong-Yon Lach: *,,Mobile Networks Support in Mobile IPv6 (Prefix Scope Binding Updates)"*, Internet Draft, March 2002.

[ARP] Gorry Fairhurst: *,,Address Resolution Protocol (arp)"*, http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html, January 2001.

[HMIPv6] Hesham Soliman, Claude Castelluccia, Karim El-Maki, Ludovic Bellier: *,,Hierarchical MIPv6 mobility management (HMIPv6)"*, Internet Draft, July 2002.

[MRTunnel] T.J. Kniveton, Jari T. Malinen, Vijay Devarapalli, Charles E. Perkins: *,,Mobile Router Tunneling Protocol"*, Internet Draft, November 2002.

[RevRouteHeader] P. Thubert, M. Molteni: *,,IPv6 Reverse Routing Header and its application to Mobile Networks"*, Internet Draft, April 2003.

[RedAddrDel] Deering, Zill: *,,Redundant Address Deletion when Encapsulating IPv6 to IPv6"*, draft-deering-ipv6-encap-addr-deletion-00.txt, Internet Draft, November 2001.

[MIND] ,,*IST-2000-28584*", MIND protocols and mechanisms specification, simulation and validation, http://www.dit.upm.es/ ist-mind/deliverables/MIND_D22_annex.pdf, Deliverable, November 2002.

[D07] ,,*Concept of Mobile Router and IVAN management*", IST-2001-35125(OverDRiVE), http://www.comnets.rwth-aachen.de/˜o_drive/publications/OverDRiVE_WP3D07_v1.1.pdf, Deliverable, March 2003.