MIPL Mobile IPv6 for Linux in HUT Campus Network MediaPoli

Antti J. Tuominen and Henrik Petander
Telecommunications and Software Engineering Institute
Helsinki University of Technology
{ajtuomin,lpetande}@tml.hut.fi

July 16, 2001

Abstract

MIPL Mobile IPv6 for Linux is an implementation, developed as a part of the GO/Core research project at HUT, of the Mobility Support in IPv6[1]. Since IPv6 and Mobile IPv6 are new emerging technologies we provide an introduction to them and a brief comparison to their version 4 counterparts. We illustrate the interoperation with FreeS/WAN IPSec implementation to provide secure and scalable mobility also in untrusted networks.

We also describe architecture and technology behind the MART nodes, custom built embedded Linux boxes acting as wireless access points used in the campus research network, MediaPoli. We will present a firsthand account on implementing Mobile IPv6 in the Linux kernel and using it in a wireless network.

1 Introduction

In the future more and more nodes connected to the Internet will be wireless. As a consequence they may also be mobile. With the expanded address space contributed by IPv6 we will be able to assign global IP addresses to any device wanting to participate in the Internet. Just having more addresses doesn't solve the problem of mobility. Because part of the IP address is used for routing purposes it must be topologically correct. This is where Mobile IP comes in.

Industry funded GO Project aims to create a vision of future mobile communications and services. GO/Core subproject is committed to providing a platform for new mobile services developed in the other GO branches. For this purpose we have developed an implementation of Mobile IPv6 for Linux. Infrastructure for this test bed is provided by HUT campus research network MediaPoli. MediaPoli's goal is to be a test-bed for future technologies both wired and wireless.

2 IPv6 and mobility

Mobility in terms of TCP/IP networking means a change of the network a mobile node is attached to. In practice the problem of user mobility is quite close to the network renumbering problem, where a whole network of nodes changes its point of attachment to the Internet.

2.1 IPv6 vs. IPv4

The phenomenal growth of the Internet was not something the designers of Internet Protocol expected when designing the TCP/IP protocol suite. Although there is not exactly a lack of single IPv4 addresses, network addresses, or continuous blocks of addresses, have been sparse since the late 90's. Also the nonaggregability of IPv4 addresses has led to the rapid increase in the size of the routing tables of back bone routers.

IPv6 was designed to fix these problems and provide a network protocol, which would work for the foreseeable future and not run out of addresses[2]. It has an address space of 2^{128}, which should be enough even if every imaginable device was equipped with an unique address. IPv6 also enables the back bone ISPs and enterprises to organize their addresses hierarchically, thus decreasing the amount of routes needed in back bone routers. Besides the larger address space IPv6 also provides other new features such as address autoconfiguration, enhanced mobility support and IPSec[3] integrated into the standard IPv6 protocol stack.

2.2 Mobility support in IPv6

As a *Mobile Node* (MN) changes its point of attachment from one network to another its IP address needs to change, to allow routers to deliver datagrams to the new network address. However, at the same time other hosts communicating with MNs, called *Correspondent Nodes,* need to be able to send packets to the MNs. Mobile IP aims to solve this problem in a way that scales to large numbers of fast moving MNs.

Mobility support in IPv6, or short Mobile IPv6, solves the routing problem caused by mobile users. Mobile IPv6 introduces *Home Agent* (HA), which keeps track of the current *care-of address* (CoA) of a MN. With this information it can deliver datagrams originally sent to the *home address* of the MN, by tunneling them to the CoA. When MN moves, it informs the HA of its new CoA by sending a *Binding Update* (BU) to the HA. The BU binds the new CoA to the home address of MN for a certain time. HA stores the *bindings* in a special data structure called the *binding cache.*

If a CN communicates with a MN using the home address of the MN, MN sends datagrams directly to the CN, but CN's packets are routed via the home network. If the home network is topologically far from the current location of MN, this is inefficient. To cure this Mobile IPv6 introduces the concept of route optimization. As a MN receives a packet tunneled by its HA it can determine that the original sender of the packet is not aware of the mobility of MN. To inform the CN, MN sends a BU to it. With the binding the CN can send datagrams directly to the MN's CoA, using a routing header. Figure 1 illustrates this procedure.

If the delay between MN and its HA and CNs is large, hand-offs may lead to significant packet loss, especially on high bandwidth links, such as WLANs. To perform smoother hand-offs a MN can also send a BU to its previous router, which will then act as a temporary HA and tunnel datagrams originally sent to the previous CoA to the new CoA of MN.

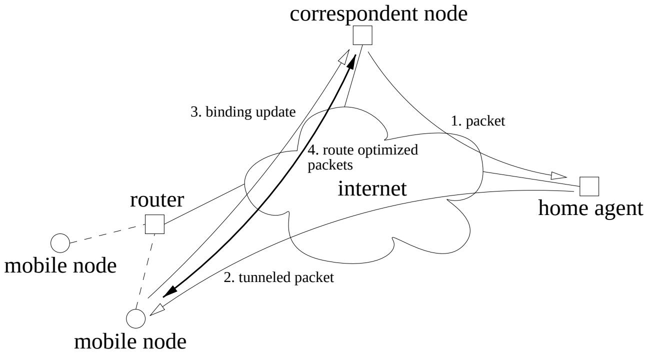

Figure 1: Mobile IPv6 Operation

The signaling in Mobile IPv6 uses destination option headers which are one type of IPv6 extension headers. The use of IPv6 destination option headers allows piggybacking of signaling information in packets carrying application data. Mobile IPv6 uses four new types of destination options:

- *Home Address* option is used to carry the home address of MN, when it is away from its home network. It is necessary for allowing CN to demultiplex the datagrams it receives.
- *Binding Update* (BU) option creates, updates and deletes entries in the binding caches of HA and CN. It is used for creating a binding between the source address of the datagram and the home address in the home address option.
- *Binding Acknowledgment* (BA) option is sent by HA and by CNs in response to a BU to inform MN of the status of the binding update.
- *Binding Request* (BR) option is sent by CN to request a MN to refresh the binding cache entry for it in the CN by sending a BU to the CN.

2.3 Mobility and IPSec

BU and BA change state in the receiving nodes and thus they need to be authenticated. Especially BUs need to be authenticated as they remotely redirect the routing of datagrams to the home address of

MN. Mobile IPv6 uses *IPsec Authentication Header*, AH, for this purpose. IPSec is a set of protocols designed to secure the TCP/IP protocol suite and it should be a part of every IPv6 implementation. IPSec AH is an IPv6 extension header, which protects the integrity of the whole datagram. Thus it also verifies the identity of the sender of the datagram.

Although IPSec provides a means of authenticating the signaling it does not solve the problem of authorization: How can MN prove to a CN that it has the authority to change the routing of its datagrams. This is not a problem between MN and HA as they are likely to already have a *Security Association*. MN and CN may not have any knowledge of each other at the beginning of their communication and thus the setting up of a SA is not trivial. The use of IKE (Internet Key Exchange) together with DNSSec provides a solution, with the assumption that both MN and CN use the same Public Key Infrastructure.

3 MIPL Mobile IPv6 for Linux

MIPL Mobile IPv6 for Linux is an implementation of the up coming standard Mobility Support in IPv6 developed in Helsinki University of Technology.

3.1 Project background

MIPL Mobile IPv6 for Linux was started as a project in the HUT Software Project course in 1999. Original project team consisted of Sami Kivisaari, Niklas Kmpe, Toni Nyknen, Juha Mynttinen, Henrik Petander and Antti Tuominen. The initial goal during the software project course was to design a working prototype of Mobile IPv6 with limited functionality.

Since June 2000 the development of MIPL has continued in the GO/Core project at the Telecommunication Software and Multimedia laboratory (TML) of HUT. The development of MIPL has contributed to the broader objective of GO/Core to design a scalable mobility management architecture for frequently moving users. Besides the efforts put in by the project team we have received lots of contributions from external developers in the form of bug reports and patches.

3.2 Design decisions

Linux was chosen as the development platform for our project for various reasons. The TM laboratory had good experiences of Linux as an open development environment from previous research projects. Also tens of Linux network access points, MART nodes had already been deployed. One of the main reasons behind choosing Linux as the platform was to make it possible to add Mobile IPv6 support to the MART-nodes later. Project team also viewed access to the operating system source code as an important factor. Thus the decision to use Linux instead of some other possible platform.

Our principal guideline was to change as little as possible of the existing IPv6 stack but it was obvious we couldn't do with no changes at all. Due to this MIPL had to have two distinct parts: a kernel patch and the actual Mobile IPv6 software. MIPL is implemented as a kernel module. It is distributed as a patch against the newest kernel and a set of userspace tools. User can configure the module for Correspondent Node (CN), Mobile Node (MN) or Home Agent (HA) functionality. Correspondent node functionality is implicitly present also with MN and HA.

Though the 2.4 kernel series were far from complete when the project began, we decided to use the 2.3 development series as the basis for our work instead of the 2.2 stable series. Keeping up with the changing kernel version did cause some extra work but in retrospect it paid off. Now with 2.4 safely out no action in our part is required.

Since Mobility Support in IPv6 being a draft was subject to change we made the decision to keep up with the development. Reason for this was much the same as with the kernel versions. We did not want to build something that would be already obsolete when complete. Again, this required extra work when compared to sticking with the same draft revision.

3.3 The implementation

First thing to do was packet interception. Home Agent needs to intercept all packets sent to the Mobile Node when MN is not at home. Since at this stage there was no intelligence in the system we used to intercept packets from hard-coded addresses. Interception was done using Netfilter. Packets are intercepted at NF_IP6_PRE_ROUTING. Concurrently we started working on extension header handling.

Next step was to tunnel intercepted packets to the actual MN location. Mobile IPv6 uses normal IPv6-in-IPv6 tunneling. Again we used predefined address for the destination. For now we had been writing only Home Agent code but at this stage we started the Mobile Node code. MN needed to decapsulate tunneled packets in order to process the real packet from the Correspondent Node. Now we had triangular routing in place.

Mobile IPv6 includes MN's home address in a Home Address Option when MN is in a foreign network. This way packets sent by MN will have a topologically correct address in the IP header and wont be dropped by possible ingress filtering. For this to work we started working on the extension header adding code. At the same time extension handling code was beginning to be usable. This required some changes to the IPv6 stack so instead of discarding the new destination options they were passed on to our handlers.

Having the triangular routing working sufficiently our next priority was to add the binding cache and binding update list. When binding cache had enough functionality for basic operation, we were also able to add Binding Updates to outgoing packets with the extension adding code. Route optimization started to take form when MN was able to send BU to CN informing it of the current care-of address. We still needed to make CN use a routing header when replying.

An essential part of Mobile IPv6 is movement detection i.e. how node detects it has moved. For this purpose MIPv6 uses Router Advertisements. The specification makes some alterations to RAs which had to be implemented. We had to change some kernel header files in order to get the IPv6 stack to accept them. MIPL has its own router advertisement handler for the modifications. When a move is detected default route must be changed to the new access router. Also old routes must be cleaned up. This was done using routing table functions.

After we had all basic functionality in place and working we started to work on the less critical ones. While IPSec is mandatory for Mobile IPv6 we had ignored it for the time being. We had been discussing IPv6 IPSec support with FreeS/WAN and were hoping we could use their services. However, IPv6 support meant a complete redesign of FreeS/WAN's kernel part KLIPS and a task this great is not completed over night.

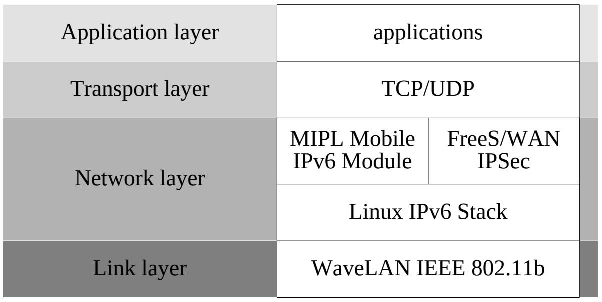

Figure 2: Network layering

MIPL implements limited IPSec functionality to authenticate the binding updates which can remotely change the routing of datagrams to the home address of a mobile node. This functionality includes IPSec AH and a Security Association database. To provide MN and CN with shared secret keys MIPL uses the Pluto key management daemon, which is a part of the FreeS/WAN IPSec package.

Since WLANs are easy to eavesdrop and the link layer encryption in WLANs can not be relied on, the use of end-to-end encryption is necessary for securing the traffic between MN and CN. Further work on the interworking of MIPL and FreeS/WAN will be started, when FreeS/WAN supports the encryption of IPv6 packets. Figure 2 illustrates the protocol layering and how MIPL and FreeS/WAN fit into the overall stack.

One of the latest features in MIPL is home agents list and Dynamic Home Agent Address Discovery (DHAAD). Home agents list keeps track of home agents on a link. When a MN starts up on a foreign network and doesn't know its HA's address it can use DHAAD to discover this information. When HA received DHAAD Request it responds with a list of addresses from the home agents list. DHAAD utilizes anycast which was not supported in Linux and made implementing this feature a little harder. DHAAD remains a bit kludgy on the home agent side but works when configured properly.

3.4 Status

MIPL is still work-in-progress. Development has moved from implementing large operational entities to single feature fixing. We anticipate to release fully draft revision 13 compliant version sometime in the fall 2001. However, Mobility Support in IPv6 is still to become a RFC and we have yet to see how much the final specification differs from draft 13. We started off with revision 8 and already five times have updated the code to comply with the latest draft.

MIPL has been to two interoperability and conformance testing events. In October 2000 we participated in the ETSI IPv6 Bake-off in France and March 2001 in Connectathon, USA. These events have uncovered bugs and discrepancies but also proved the implementation to conform to the specification quite well and to interoperate with other implementations.

4 MediaPoli

MediaPoli is a high speed research network mainly located in the Helsinki University of Technology Campus and neighboring areas.

4.1 Background

MediaPoli, founded in 1998, is a new research network covering HUT campus area. MediaPoli's objective is to provide high performance test bed for new technology, innovation and new types of services. MediaPoli is operated by a separate company partly owned by HUT to allow piloting commercial services while such is not allowed in the university network.

MediaPoli has a Gigabit Ethernet backbone which connects the building LANs. Wireless MediaPoli offers up to 11Mbps speeds with IEEE 802.11 WaveLAN technology. Radio coverage is available in nine MediaPoli connected buildings. Three outdoor antennas also provide coverage around HUT Main Building, CS Building and TML premises. High speed networks allow such services as video-on-demand, live streaming services and multipoint video conferencing.

4.2 Technology: The MART Node

Wireless MediaPoli consists of about 80 wireless access points. Rather than using standard off-the-shelf WaveLAN access points MediaPoli uses so called *MART nodes* developed in the Mobile Ad hoc Routing Testbed (MART) project[4]. MART node is an embedded PC running Linux, acting as a WaveLAN - Ethernet router. MART node has a motherboard with PC/104 (ISA) bus and a built-in 10baseT Ethernet interface. The node is also equipped with a PCMCIA bridge and for 802.11 interface a Lucent WaveLAN IEEE PCMCIA card is used.

A MART node has 32 megabytes of memory and a 20MB solid state flash mass storage. The filesystem is packed on the solid state drive to reduce needed disk space. On boot the filesystem is decompressed to a RAM disk.

Using the afore described setup provides very flexible environment for testing new technologies. MART nodes can be expanded very easily. Originally MART nodes offered choice of DHCP/NAT or (Dynamics) Mobile IPv4[5]. While DHCP with NAT is sufficient for many services like browsing the web, other services require real addresses only available with Mobile IP. We are adding (MIPL) Mobile IPv6 support to MART node's repertoire allowing users to choose MIPv6, MIPv4 or just DHCP with NAT.

5 Putting it together

GO/Core aims to equip the entire Wireless MediaPoli with Mobile IPv6 support. In addition to implementing Mobile IPv6 we had another challenging task of fitting MIPL in the MART node and wireless environment.

All MIPL development work has been done on wired networks and we used to simulate moving by unplugging a computer from a switch and plugging it in another. First time we actually tried it on a wireless network we hit the wall immediately. The problem was that the Lucent WaveLAN IEEE drivers did not seem to support IPv6 which was rather bizarre. With a later pcmcia-cs package versions this problem suddenly disappeared. Later on it turned out that it may have actually been caused by change of the C compiler.

Next we stumbled on the non-standard ad hoc mode. While MART nodes serve as access points and there are no real access points we cannot use the infra-mode. We must use WLAN cards in ad hoc mode to allow peer to peer communication instead of peer to access point. Many of the earlier WaveLAN cards have the manufacturer's own ad hoc mode which unfortunately is not compatible with others. Fortunately however, this can be cured with a simple firmware update. After getting all the WaveLAN cards talking the same language (IBSS Ad hoc mode) we just needed a new driver version for Linux supporting IBSS which appeared quite soon after the firmware.

After resolving the problems we had a working prototype for the new MART node with MIPv6 support. Since we wanted both MIPv4 and MIPv6 support at the same time we had to make sure there was no conflict in having Dynamics and MIPL in the same computer. Any fears we might have had were proven unfounded and everything worked perfect on the first go.

6 Conclusion

While we have not yet reached our goal of supporting Mobile IPv6 in entire MediaPoli wireless LAN, we have developed building blocks that will in the near future take us there. Also we have demonstrated interoperation of these components in laboratory scale.

Combination of Linux, the MART node and MIPL Mobile IPv6 offers excellent testbed for new mobile technologies. Using Linux allows development on desktop machines for easier testing. MART node is a full PC and runs same software as any PC which alleviates the need for emulated environment when developing and testing on desktop.

The MART node with Linux provides an extremely flexible and customizable alternative to factory made access points with slightly less cost. It also allows integrating far more intelligence in the access point if needed. Mobile IPv6 allows testing and developing future applications utilizing IPv6.

In the future we will continue using products of this project for research, designing and implementing technologies such as mobile ad hoc networks and mobile security.

References

[1] D. B. Johnson and C. E. Perkins, "Mobility support in IPv6." Internet-Draft, Nov. 2000. Work in progress.

[2] S. King, R. Fax, D. Haskin, W. Ling, T. Meehan, R. Fink, and C. E. Perkins, "The case for IPv6." Internet-Draft, June 2000. Work in progress.

[3] S. Kent and R. Atkinson, "RFC 2401: Security architecture for the Internet Protocol," Nov. 1998.

[4] H. Arppe, D. Forsberg, J. Malinen, P. Massetti, and J. Salmi, "Mobile ad hoc routing testbed (MART)." Software/Hardware, 1999. http://www.cs.hut.fi/ mart/.

[5] D. Forsberg, J. K. Malinen, J. T. Malinen, and T. Weckstrm, "Dynamics - HUT Mobile IP technical document." In the software release 0.5, Aug. 1999.