

Version  
2.0beta

MOSQUITONET MOBILE IPv4

---

Mobile Computing Group  
Stanford University

# User's Manual

MOSQUITONET MOBILE IPv4

# User's Manual

---

© Stanford University  
Mobile Computing Group  
Stanford University  
Email: [mip@MosquitoNet.Stanford.EDU](mailto:mip@MosquitoNet.Stanford.EDU)  
Web: <http://MosquitoNet.Stanford.EDU/mip>

# Table of Contents

<b>A BRIEF INTRODUCTION TO MOBILE IP .....</b>	<b>1</b>
WHY MOBILE IP .....	1
PROTOCOL OVERVIEW .....	2
<i>Running with the Support of Foreign Agent</i> .....	2
<i>Running with a Co-located Care-of Address</i> .....	3
FURTHER INFORMATION .....	3
<b>MOSQUITONET MOBILE IPV4 IMPLEMENTATION.....</b>	<b>4</b>
OUR DESIGN OVERVIEW .....	4
NEW FEATURES IN THIS RELEASE.....	5
RELATED WORK WITHIN OUR GROUP .....	5
<i>STRIP</i> .....	5
COMPATIBILITY .....	5
COMPONENTS IN THIS DISTRIBUTION .....	6
<i>Kernel modifications</i> .....	6
<i>Daemon code</i> .....	6
<i>Scripts</i> .....	6
<i>Documentation</i> .....	6
MAILING LIST .....	6
<b>COMPILATION AND INSTALLATION OF THE DISTRIBUTION.....</b>	<b>7</b>
BUILDING KERNEL WITH MOBILITY SUPPORT .....	7
<i>Obtaining Linux Kernel Source Tree</i> .....	7
<i>Applying Our Patch to the Kernel</i> .....	7
<i>Kernel Configuration</i> .....	8
<i>Kernel Compilation</i> .....	8
COMPILING USER-LEVEL DAEMONS .....	8
INSTALLATION.....	9
<i>On the Home Agent</i> .....	9
<i>On the Mobile Host</i> .....	9
<b>CONFIGURATION AND USE OF THE SYSTEM.....</b>	<b>10</b>
CONFIGURATION .....	10
<i>On the Home Agent</i> .....	10
<i>On the Mobile Host</i> .....	10
USING THE SYSTEM .....	11
<i>Starting Up Mobility Support on Home Agent</i> .....	11
<i>Using Mobile IP Support on Mobile Hosts</i> .....	12
SYSTEM MESSAGES AND DEBUGGING .....	12
<b>LIMITATIONS AND THINGS TO BE DONE.....</b>	<b>14</b>
LIMITATIONS OF THE CURRENT RELEASE .....	14
FEATURES TO BE RELEASED IN LATER DISTRIBUTIONS .....	14
<b>REFERENCES.....</b>	<b>15</b>

---



# A Brief Introduction to Mobile IP

This chapter provides the background information for MosquitoNet Mobile IPv4 implementation. It is only intended to highlight the most important parts of the protocol: the motivation, and the basic operation of the protocol, with pointers to further detailed information.

## Why Mobile IP

IPv4 uses hierarchical routing, and therefore a host's IP address uniquely identifies the host at a particular point of attachment to the Internet. The hierarchical routing is necessary to deal with the scalability issue of the Internet routing. This, unfortunately, means that a host has to be on the network indicated by its IP address in order to receive packets destined to it. For a mobile host to be able to communicate as it moves from one location to another, one of the following two things has to be in place:

- The mobile host must change its IP address whenever it moves to a new place;
- Host-specific routes must be propagated throughout the Internet routing fabric.

The first breaks higher layer (such as transport) connections and makes it impossible for other hosts to initiate communication to a mobile host when the mobile host changes location. The second has serious scalability problem, and is simply unrealistic.

Mobile IP is a scalable mechanism designed to accommodate host mobility within the Internet. It enables a mobile host to change its point of attachment to the Internet while keeping an unchanging IP address called its home IP address. It does not require changes in the existing routing infrastructure. It sits below the transport layer of the TCP/IP stack, and thus provides transparent mobility support to higher layer protocols.

Unlike CDPD that also provides mobility support, Mobile IP protocol sits above the link layer and is technology independent. That is, it works equally well for mobility across homogeneous media as across heterogeneous media. To Mobile IP, it makes no difference whether a mobile

host is moving from one Ethernet segment to another or it is actually from Ethernet connection to a wireless LAN.

Mobile IP is especially suited to address the "macro" mobility management problem. Though it can also be used for "micro" mobility management such as handoffs between small cells of wireless network, link-layer mechanism will probably be more appropriate than Mobile IP.

## Protocol Overview

The basic idea behind Mobile IP is for a mobile host to always keep its static home IP address no matter what its current point of attachment to the Internet is. Packets addressed to it will always go via its home network and forwarded on from there when necessary.

When the mobile host is in its home network, it acts just like a normal stationary host. When it is away visiting a foreign network (i.e. a network other than its home network), the mobile host needs to obtain a temporary IP address (called care-of address) in this subnet. It then registers the care-of address with its home agent, a stationary host in the mobile host's home network that provides mobility support for mobile hosts. The home agent will therefore intercept packets destined to the mobile host in its home network, and forward them to the mobile host's current point of attachment. The forwarding is done by "tunneling" the packets to the mobile host's care-of address by encapsulating the original packet in another IP packet destined to the mobile host's care-of address. At the end of the tunnel, the packets are de-capsulated and thus revealing the original packets before delivering to the mobile host. Packets originated from a mobile host is sent the same way as on other stationary hosts, except that the source IP address of the packets is set to be the mobile host's home IP address to preserve its home identity.

This is how the basic protocol works. Depending on how a mobile host obtains its care-of address, there are two operation modes of Mobile IP on a mobile host.

Running with the Support of Foreign Agent

A foreign agent is a stationary host in the network a mobile host is visiting that provides mobility support to mobile hosts. The mobile host uses the care-of address provided by the foreign agent to register with its home agent.

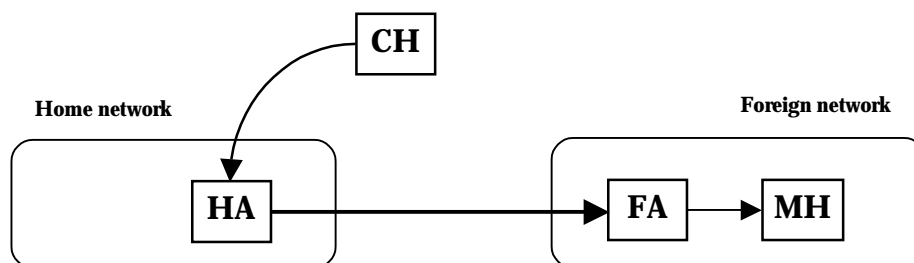


Figure 1: Running with foreign agent support.

Running with a Co-located Care-of Address

The mobile host obtains its own temporary care-of address either through DHCP or any other means. Therefore, the other end of the tunnel is the mobile host itself and it needs to de-capsulate packets itself.

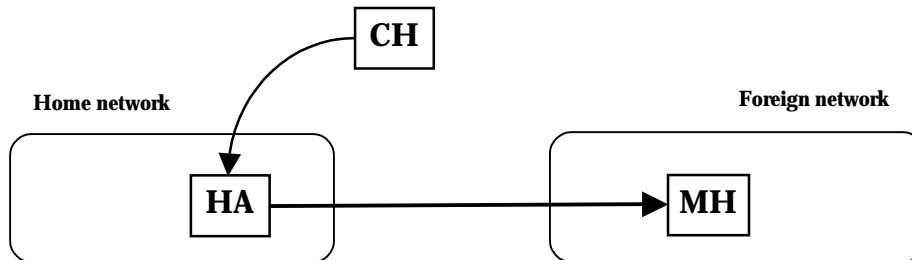


Figure 2: Running with co-located care-of address.

## Further Information

Documents included in this distribution:

- RFC 2002: IP Mobility Support;
- RFC 2003: IP Encapsulation within IP;
- RFC 2004: Minimal Encapsulation within IP;
- RFC 2005: Applicability Statement for IP Mobility Support;
- RFC2344: Reverse Tunneling for Mobile IP.

IETF Mobile IP Working Group is in charge of the standardization process of the protocol. A good source of information on work related to Mobile IP is:

- Mobile IP Working Group's Home Page:  
<http://www.ietf.org/html.charters/mobileip-charter.html>
- Mailing List: [mobile-ip@standards.nortelnetworks](mailto:mobile-ip@standards.nortelnetworks)

Most of the discussions are carried out through the mailing list. Please do not send join request directly to this list. Refer to the web site for instructions on how to join the mailing list.

Information about IETF can be found at <http://www.ietf.org/>.

# MosquitoNet Mobile IPv4 Implementation

This chapter describes the design focus and implementation status of MosquitoNet Mobile IPv4.

## Our Design Overview

MosquitoNet Mobile IP implementation is based on IETF Mobile IP specification (RFC 2002). We have implemented all required capabilities on both mobile host and home agent. Our implementation, however, does not provide yet another foreign agent implementation. Instead, we put more emphasis on running with co-located care-of address and exploit the extra flexibility made possible by this running mode. The motivations are:

- The advantage of avoiding assuming foreign agent support in the network a mobile host is to visit. Refer to our USENIX'96 paper "Supporting Mobility in MosquitoNet". It is also available at:

<http://mosquitonet.Stanford.EDU/publications/usenix96.mobile.ps>.

- The extra flexibility made possible when running with co-located care-of address. Refer to our MobiCom'98 paper "Flexible Network Support for Mobility". It is available at URL:

<http://mosquitonet.stanford.edu/publications/mobicom98-flexible.ps>

- Running with co-located care-of address seems to be the direction to go, if not already happening before IPv6 is here. A foreign agent providing a care-of address and encapsulate/de-capsulate support for mobile hosts and then requiring mobile hosts to register through it should really be replaced by a dynamic address allocation mechanism. Let the mobile host both be a real host on the visited network and preserve its home identity. That is most desirable.

Our mobile host, however, can interoperate with foreign agent if there is one available in the network it is visiting. This is not just for the conformance purpose. We do not think foreign agent



support is a bad thing, just that we do not want to always rely on it and favor running in co-located care-of address mode.

## New Features in this Release

This release is mainly a port of our previous release to 2.2.x Linux kernels. It is integrated with the policy routing mechanism provided by the Linux kernel, and makes use of netlink mechanism for communication with the kernel.

## Related Work within Our Group

The following projects are related with MosquitoNet Mobile IP implementation. We use them together with our Mobile IP implementation in daily operation. They help enhance the experience with mobility support provided by Mobile IP.

### STRIP

STRIP is a protocol for carrying IP packets over the "Star mode" interface of 900MHz packet radios from Metricom. STRIP gives you a "wireless IP subnet", similar to a conventional Ethernet network, except slower (and without the wires).

We wrote a device driver to use Metricom radio to send and receive IP packets, so that we can use Mobile IP to maintain ongoing network conversations while switching between Ethernet interface and Metricom radio (when we leave our office).

Check out the following web page at:

<http://mosquitonet.Stanford.EDU/software/strip.html> for details.

## Compatibility

MosquitoNet Mobile IP implementation conforms to IETF Mobile IP specification (RFC 2002). Although we currently do not provide a foreign agent implementation of our own for the reasons mentioned earlier, our mobile host and home agent implementation is not a subset of the IETF Mobile IP base protocol.

Our mobile host implementation can use foreign agent support if a foreign agent is available in the network the mobile host is visiting. We have successfully conducted the interoperability test at the Connectathon hosted by FTP Software.

## Components in this Distribution

### Kernel modifications

Kernel part of our Mobile IPv4 implementation is in directory `./kernel`. It includes a patch file for 2.2.5 kernel (default of RedHat 6.0).

### Daemon code

User level daemons of our Mobile IPv4 implementation are in directory `./daemons`. It includes source code for a mobile host daemon (mhd) and a home agent daemon (had).

### Scripts

Supporting scripts and sample configuration files are in directory `./scripts`. The `./etc` directory contains configuration files, the `./sbin` directory contains a script (mip) used to control the mobile host daemon, and the `./usr` directory contains man pages for programs included in the distribution.

### Documentation

User's manual (this document) and RFCs relevant to Mobile IP protocol specification are in directory `./doc`.

## Mailing List

For bug report or comments on our Mobile IP implementation, please send email to [mip@mosquitonet.stanford.edu](mailto:mip@mosquitonet.stanford.edu).

## Compilation and Installation of the Distribution

This chapter describes how to install MosquitoNet Mobile IP software. The implementation contains two parts: kernel modifications for core mobility support, and user-level daemons that mainly handle protocol messages. Therefore first of all, you need to build a kernel with Mobile IP support. Then you need to compile the daemons and install them into proper directories. Please follow the steps in this chapter for installation, and then we will provide instructions on configuration in the next chapter.

### Building Kernel with Mobility Support

Obtaining Linux Kernel Source Tree

Our kernel patch is based on most commonly used stable Linux kernel. We have prepared patches for kernel version 2.2.5 (default of RedHat 6.0), 2.2.12 (RedHat 6.1), and 2.2.14 (RedHat 6.2). You need to have the whole kernel source tree in order to apply our patch.

The Linux kernel source tree can be obtained from:

<http://www.kernel.org/pub/linux/kernel/v2.2>

or its mirror sites. Do:

```
cd /usr/src  
  
gzip -cd linux-2.2.5.tar.gz | tar xvf -
```

to get the kernel source tree in place.

Applying Our Patch to the Kernel

“cd” into the top-level directory of the Linux kernel source tree. Copy our patch file for the kernel over, and apply it using Larry Wall’s “patch” program. Type:

```
patch -p1 < mip-kernel-2.2.5.patch
```

This patch also includes new files added in the kernel to provide Mobile IP support.

Kernel Configuration

Do a “make xconfig”, and make sure that the following options are enabled in the networking options section:

- Mobile IP support
- IP: advanced router (be sure to enable the related policy routing)
- IP: tunneling
- Kernel/User netlink socket
- Routing messages

Kernel Compilation

First, make sure that your `/usr/include/asm` and `/usr/include/linux` are set correctly as symbolic links to corresponding directories in Linux kernel source tree:

```
cd /usr/include
rm -rf asm linux
ln -s /usr/src/linux/include/asm asm
ln -s /usr/src/linux/include/linux linux
```

Then, do “make dep; make clean; make bzImage”. You will also need to do “make modules; make modules\_install” if you are chosen to have some kernel components built a loadable modules. The kernel built will be used on both the mobile host and the home agent.

## Compiling User-level Daemons

Before compiling the daemon code, make sure that `/usr/include/asm` and `/usr/include/linux` both are pointing to the patched kernel source tree. This is important! Otherwise, the compiler will complain about some undefined symbols.

Then, change directory to the top-level directory that contains the daemon code, and do “make clean; make all”. The mobile host daemon (mhd) and the home agent daemon (had) will be built in their respective directories `./mhd` and `./had`.

## Installation

Install the new kernel image enhanced with Mobile IP support on both the mobile host and the home agent. Make them the kernel to boot from next time your computer reboots. You may want to save your previously used kernel image just in case anything goes wrong.

Then copy files under `./scripts` directory to the corresponding directories under the root file system (`/`) on your computer.

On the Home Agent

Put home agent daemon (`had`) in `/sbin`.

On the Mobile Host

Put mobile host daemon (`mhd`) in `/sbin`.

## Configuration and Use of the System

This chapter describes how the system should be configured and used. You need to modify the configuration files for both the home agent daemon and the mobile host daemon before invoking the daemons.

### Configuration

On the Home Agent

Copy `scripts/etc/had.conf` file to `/etc`. Modify the configuration to suit your site according to this sample file. Most of the configurable entries are self-explanatory. There are two things that need to be emphasized:

- The number of mobile hosts supported (as specified by keyword `MOBILE_HOSTS`) should match the number of lines following that specify the security association between these mobile hosts and the home agent.
- The security association between a mobile host and its home agent is specified by the mobile host's home IP address, SPI (Security Parameter Index), and shared secret key in this order. The SPI is an integer, and you can pick any valid integer as long as it is the same in the mobile host's configuration file. The shared secret key should be a 16-character string.

On the Mobile Host

Copy `scripts/etc/mhd.conf` to `/etc`. Modify the configuration to suit your site according to the sample file. Most of the configurable entries are self-explanatory. Just make sure that the security association between the mobile host and its home agent should match the corresponding entry on the home agent.

Copy `scripts/etc/mpt.conf` to `/etc`. This is the configuration file for choosing a set of mobile policies to use according to the current care-of address in use. A sample configuration is included

below. If you just want to use the default behavior, i.e. always using Mobile IP with bi-directional tunneling, simply leave this file empty.

```
care-of address=171.64.0.0
netmask=255.255.0.0
entries=2
0.0.0.0 0.0.0.0 80 0 0
0.0.0.0 0.0.0.0 0 1 0

care-of address=0.0.0.0
netmask=0.0.0.0
entries=3
0.0.0.0 0.0.0.0 80 0 0
171.64.0.0 255.255.0.0 0 1 1
0.0.0.0 0.0.0.0 0 1 0
```

This configuration indicates that we have specified two mobile policy entries to use for care-of addresses in network 171.64.0.0 with netmask 255.255.0.0. The first entry indicates that for destination to any network (the first 0.0.0.0 with netmask being the second 0.0.0.0) and using port 80 (usually http traffic) should not use mobile IP (the first 0) and hence no bi-directional tunneling (the second 0). The second entry indicates you can use triangular route with mobile IP for all other traffic when using care-of addresses within the range specified. For all other care-of addresses, there are three mobile policy entries. The only difference is that the traffic not using port number 80 will use Mobile IP with bi-directional tunneling when the destination is to network 171.64.0.0 with netmask 255.255.0.0, while others can still use triangular route with Mobile IP. This is a sample configuration in use when the IP source address filtering routers are at the boundary of our institution (i.e. network 171.64.0.0).

We also provide some scripts to help using the mobile host daemon. Copy sbin/mip to /sbin. This is the script to control the mobile host daemon. The man pages for mip, mhd, and had are included in usr/local/man/man1.

## Using the System

Now you can reboot your computer and run the new daemon.

Starting Up Mobility Support on Home Agent  
The virtual interface should be set up by:

```
ifconfig vif up <IP_address_of_home_interface>
netmask <netmask_of_home_interface>
broadcast <broadcast_address_of_home_interface>
```

The home agent daemon is invoked by `"/sbin/had -d -v"`, `-v` for verbose, `-d` for debugging.

You may want to add them to `/etc/rc.d/rc.local` so that it automatically gets started when your home agent is rebooted.

Using Mobile IP Support on Mobile Hosts

The mobile host daemon is invoked by `"mip start"`. Do not try to invoke `mhd` directly. Always control the mobile host daemon through the script.

The mobile host daemon will not try to register with the home agent when it is started. Instead, it will run in background, waiting for signals. You may want to add it to the init script after `syslogd` is started so that it is automatically invoked when the mobile host is rebooted. The reason that it needs to be put after `syslogd` is that it uses `syslog` facility for error and debug messages.

`"mip restart"` and `"mip update"` send `HUP` and `USR1` signal to the mobile host daemon respectively, which causes the mobile host daemon to reevaluate its current point-of attachment and register with its home agent accordingly. The difference between `"mip restart"` and `"mip update"` is that `"mip restart"` will need to reread the configuration file for the daemon.

Basically, to use the mobile host daemon to maintain registration with the home agent, you need to do one of the following two things:

- If you are running with a co-located care-of address, you need to set up the interface and routing table just like any other normal host on the same network. Then, give the mobile host daemon a kick by either `"mip update"` or `"mip restart"`.
- If you want to probe for foreign agent support, remove default route setup in the routing table (as an indication that the mobile host does not know its current point of attachment) before giving the mobile host daemon a kick. The mobile host daemon will then be in hunting for foreign agent mode. If it picks up one, the mobile host daemon will set up routing table and network interface accordingly.

Therefore, what you need to do is whenever you change your current point of attachment you should give the mobile host daemon a kick. You may want to add `"mip update"` to the `PCMCIA` script so that when the card insertion/remove events will trigger the mobile host to reevaluate its point of attachment to the Internet.

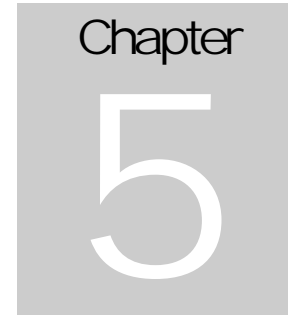
## System Messages and Debugging

Both mobile host daemon and home agent daemon use `syslog` facility. Depending on your `syslog` configuration, the error and debug messages will be in one of the `/var/log` files. There are three types of messages produced by the daemons, `LOG_INFO` for informational, `LOG_ERR` for error conditions, and `LOG_DEBUG` for debugging purpose only.



When something is wrong, you can look at these log files for clues. Invoking the daemon with “-v” flag will produce more verbose log information, and with “-d” flag will include debugging messages.

For home agent daemon, sending it HUP signal will dump to the current status of all mobile hosts served by the home agent to syslog.



## Limitations and Things to be Done

This chapter describes the known limitations of the current distribution. It also describes some new features that will be included in future releases.

### Limitations of the Current Release

We have implemented the required features of the specification. Some of the optional features are therefore currently not supported. Those include:

- Only time-stamp based replay attack detection is implemented. The nonce replay attack protection is not implemented.
- Only IP-within-IP encapsulation is used in tunneling, while optional minimal encapsulation is not implemented yet.

### Features to Be Released in Later Distributions

On our to-do list is the multicast support for mobile hosts, which was present in our previous releases for 2.0.x kernels. Due to the amount of work involved in setting up the test environment (including setting up mouted), it is not included in this first pass of the porting. We will release the multicast support for mobile hosts in the next release.

## References

C. Perkins, Editor, "IP Mobility Support." *RFC 2002*, October 1996.

C. Perkins, Editor, "IP Encapsulation within IP." *RFC 2003*, October 1996.

J. Solomon, "Applicability Statement for IP Mobility Support." *RFC 2005*, October 1996.

G. Montenegro, Editor, "Reverse Tunneling for Mobile IP." *RFC 2344*, May 1998.

Mary G. baker, Xinhua Zhao, Stuart Cheshire, and Jonathan Stone, "Supporting Mobility in MosquitoNet." *Proceedings of the 1996 USENIX Technical Conference*, January 1996.

Xinhua Zhao, Claude Castelluccia, and Mary Baker, "Flexible Network Support for Mobility." *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 1998)*, October 1998.