Evaluation of Mobile IP implementations under Linux

Attila Weyland*

December 22, 2000† ‡

Abstract

The purpose of this document is to give an overview of the current Mobile IP implementations under Linux, to provide results from the basic tests done with them (Part I) and to show in detail how to set up a Mobile IP test environment based on the Dynamics distribution (Part II).

These tasks were described in a project from the Computer and Distributed Systems Group (RVS) at the Institute of Computer Science and Applied Mathematics (IAM), University of Berne and performed as a student project.

*E-Mail: `weyland@iam.unibe.ch`

†Part I last updated on `July 12, 2001`

‡Part II last updated on `July 6, 2001`

Contents

Part I
Synopsis

1 Introduction

This part of the document shows current Mobile IP implementations, explains the test environment and provides results from the basic tests.

1.1 Mobile IP

Mobile Internet Protocol (IP) has been designed to support host mobility. It enables mobile hosts to stay connected to the Internet regardless to any changes to their location.

1.1.1 Features

- No geographical limitations:
  - A user can take a palmtop or laptop computer anywhere without losing the connection to the home network.
- No physical connection required:
  - Mobile IP finds local IP routers and connects automatically. It is phone jack and wire free.
- Modifications to other routers and hosts is not required:
  - Other than mobile nodes/routers, the remaining routers and hosts will still use current IP. Mobile IP leaves transport and higher protocols unaffected.
- No modifications to the current IP address and IP address format:
  - The current IP address and address format remains the same.
- Supports security:
  - Authentication is performed to ensure that rights are being protected.

1.1.2 Entities

Mobile IP is consisting of the following entities:

Mobile Node (MN): A host or router that may change its point of attachment from one network or subnetwork to another through the Internet. This entity is pre-assigned a fixed home address on a home network, which other correspondent hosts will use to address their packets to, regardless of its current location.

Home Agent (HA): A router that maintains a list of registered mobile nodes in a visitor list. It is used to forward mobile node-addressed packets to the appropriate local network when the mobile nodes are away from home. After checking with the current mobility bindings for a particular mobile node, it encapsulates datagrams and sends it to the mobile host's current temporary address.

Foreign Agent (FA): A router that assists a locally reachable mobile node that is away from its home network. It delivers information between the mobile node and the home agent.

Care-of-address (COA): An address, which identifies the mobile node's current location. It can be viewed as the end of a tunnel (see below) directed towards a mobile node. It can be either assigned dynamically or associated with its foreign agent.

Correspondent Node (CN): A node sends the packets, which are addressed to the mobile node.

Home Address: A permanent IP address that is assigned to a mobile node. It remains unchanged regardless of where the mobile node is attached to the Internet.

Mobility Agent: An agent, which supports mobility. It could be either a home agent or a foreign agent.

Tunnel: The path, which is taken by encapsulated packets. It is the path which leads packets from the home agent to the foreign agent.

1.1.3 Supported services

The following services are supported in Mobile IP:

Agent Discovery: Home agents and foreign agents broadcast their availability on each link to where they can provide service. A newly arrived mobile node can send a solicitation on the link to learn if any prospective agents are present.

Registration: When the mobile node is away from home, it registers its care-of-address with its home agent so that the home agent knows where to forward its packets. Depending on the network configuration, the mobile node could either register directly with its home agent, or indirectly via the help of its foreign agent.

Encapsulation: The process of enclosing an IP datagram within another IP header, which contains the care-of-address of the mobile node. The IP datagram itself remains intact and untouched throughout the enclosing process.

Decapsulation: The process of stripping the outermost IP header of the incoming packets so that the enclosed datagram can be accessed and delivered to the proper destination. Decapsulation is the reverse process of encapsulation.

1.1.4 Overall Processes

Four different stages in chronological order:

Agent discovery: When a mobile node is away from home, it wants to find agents so it does not lose access to the Internet. There are two ways of finding agents. The first is by selecting an agent from among those periodically advertised, and the second is by sending out a periodic solicitation until it receives a response from a mobility agent. The mobile node thus gets its care-of-address, which may be dynamically assigned or associated with its foreign agent.

Registration: The mobile node registers its care-of-address with its home agent in order to obtain service. The registration process can be performed directly from the mobile node, or relayed by the foreign agent to the home agent, depending on whether the care-of-address was dynamically assigned or associated with its foreign agent. Note that simultaneous registrations with multiple care-of-addresses is possible.

In service: This is the period after the registration process and before the service time expiration, provided that the mobile node stays in the service area. During service time, the mobile node gets forwarded packets from its foreign agent which were originally sent from the mobile node's home agent. Tunneling is the method used to forward the message from home agent to foreign agent and finally to mobile node.

Deregistration: After the mobile node returns home, it deregisters with its home agent to drop its registered care-of-address. In other words, it sets its care-of-address back to its home address. The mobile node achieves this by sending a registration request directly to its home agent with the lifetime set to zero. There it is no need to deregister with the foreign agent because the service expires automatically when the service time expires.

1.1.5 Conclusion

Mobile IP is a newly defined protocol, which supports mobile users but also is compatible with the current IP. It is still in the process of being standardized, and there are still many items that need to be worked on and enhanced, such as the security issue and the routing issue. The IETF has been continuously working on the problems which had been found on the base Mobile IP protocol.

2 Available solutions

The listing does not claim to be complete, although a deep search has been undertaken. For more information on the topic will you consult the links given under References of Part I on page 11.

Since some of the available solutions are quite out of date, a specific distinction between newer and older software packages has been made in Table 1 and Table 2. Due to the fact that there is still a lot of ongoing research and development work in the Mobile IP area, only recent implementations were tested in the experiments.

2.1 Overview

2.1.1 Recent implementations

These are the most recent Mobile IP implementations for Linux:

Project name	Developing Institution
Dynamics [1]	TSE-Institute (Telecommunication and Software Engineering) Helsinki University of Technology
Linux MobileIP [2]	MosquitoNet Mobile Computing Group Stanford University

Table 1: Recent Mobile IP implementations

2.1.2 Other implementations

The implementations listed in Table 2 are more than one year old or their development has been discontinued:

Project name	Developing Institution
Linux Mobile-IP [3]	Department of Computer Science Binghamton University
Mobile IP [4]	Networking and Security Center Sun Microsystems Laboratories
Mobile IP [5]	HP Labs Bristol, UK
NUS Mobile IP [6]	CCN Lab (Computer Communication Networks) National University of Singapore (NUS)
Secure Mobile Networking [7]	Department of Computer Science Portland State University (PSU)

Table 2: Other implementations

2.2 Description

The following sections describe the most recent implementations.

2.2.1 Dynamics - HUT Mobile IP

- *Version:* `0.7.1` (development branch), 2000/11/05
- *Characteristics:* Dynamics is a hierarchical Mobile IPv4 software. It consists of three independent daemon executables:
  - Home Agent (HA)
  - Foreign Agent (FA)
  - Mobile Node (MN)

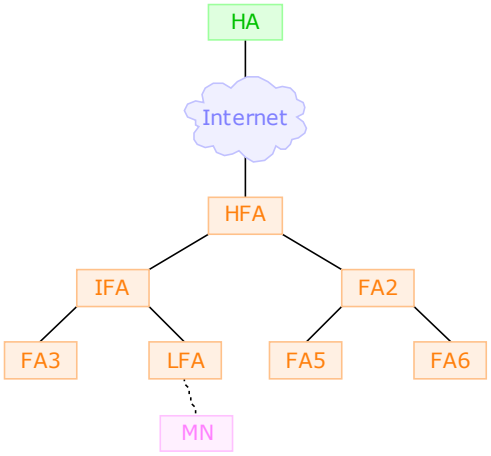

Figure 1: Mobility Agent hierarchy

In order to make Standard Mobile IP better scale to frequent location updates, a concept of FA hierarchies, wherein the number of levels is not restricted, has been implemented as an extension to the RFC 2002.

It is possible to set up a static configured tree of FAs, where the Highest Foreign Agent (HFA) represents the root providing the connection to the HA. In addition, Lowest Foreign Agents (LFAs), which are the closest FAs to the MN (the leaves of the tree), should be defined. Between the HFA and LFAs Intermediate Foreign Agents (IFAs) are responsible for passing the messages along the current path through the tree (see Figure 1).

One can also flatten the hierarchy to one level by configuring a single FA to be HFA, LFA and IFA at the same time.

The decision of the MN daemon to change the current Agent is not directly taken by the Dynamics software. Instead it follows the behavior of the wireless hardware. That means the MN daemon is only listening on the LINK layer for incoming AAs.

For example if the MN moves to far from the current Access Point (AP) in the Home Network the signal quality will drop below a certain threshold and the wireless card itself starts to scan the channels/frequencies for a closer AP with better signal quality. Shortly after it finds one in the Foreign Network and the card will start the process of registration with the new AP.

Now the MN daemon gets the AA from the FA and the process of deregistration in the old subnet starts.

More technical information can be found in the 802.11 standard [9].

2.2.2 MosquitoNet Linux Mobile IPv4

- *Version:* `2.0.2 beta`, 2000/08/22
- *Characteristics:* This implementation focuses on the co-located care-of address mode, therefore it includes only two user-level daemons:
  - Home Agent (HA)

– Mobile Node (MN)

For the MosquitoNet Linux Mobile IPv4 implementation the kernel needs to be patched.

The fact that no FA daemon is being provided emphasizes the preference of MosquitoNet to use co-located care-of address mode when roaming in Foreign Networks (e.g. use DHCP to obtain the temporary care-of address).

However the usage of a (third-party) FA, which provides the care-of address, is supported.

2.3 Comparison

Table 3 gives a brief comparison of the major features supported by each of the current implementations. Currently the only and - in compliance to RFC 2002 - obligatory authentication algorithm implemented, is keyed-MD5.

Name	Version	Features				
		Authentication Algorithm	Replay Protection Method		Encapsulation Types	
			timestamps	nonces	IPIP	GRE
Dynamics	`0.7.1`	keyed-MD5	✓	✓	✓	✓
MosquitoNet	`2.0.2 beta`	keyed-MD5	✓	–	✓	?

Table 3: Comparison of the latest implementations

3 Test environment

At the beginning of the project the idea was to use Cisco routers with Mobile IP support to emulate HA and FA behavior. But a difficult process of configuring and testing a lot of options led to no result. Instead of having pieces from different implementers, a complete, one party solution has been chosen: Dynamics - HUT Mobile IP.

The test scenario has been built along the needs of this software (for more details regarding the setup see the instructions in Part II of this document).

The MosquitoNet implementation has not been tested because of its restriction to co-located care-of address mode, for there is no DHCP server in our lab. In addition no third-party FA would work together with the MosquitoNet daemons.

- Hardware used:
  - 5 PCs (HA, FA, CN, 2 GWs)
  - 1 Laptop (MN)
  - 3 WaveLAN cards (1 for the MN, 2 for the APs)
  - 2 Access Points
  - 1 Switch (VLANs)

Each subnet hosts a WaveLAN Access Point sending on a different frequency. Together they spawn a WaveLAN in which the MN can move. The two Gateways provide the connection to the outside world for the corresponding subnet.

The HA and the FA are both running as daemon on the respective PC. The MN daemon is running on the Laptop.

For compatibility reasons with the Dynamics implementations, where the HA or FA respectively and a router could not be the same machine, two Gateways were added.

In a previous scenario the HA daemon was running on the gateway of the Home Network shown in Figure 2. When the MN moved to the Foreign Network and wanted to register there, the FA daemon forwarded the Registration Request to the HA. But the HA send its Registration Reply directly through the interface `eth0` to the outside world and therefore the packet got the global address of the machine. This source address did of course not match the one defined for the HA in the configuration file of the FA daemon, so the packets were dropped by the FA.

Note: With the release of version `0.7.1` the structure of the HA configuration file has been adapted to the one from the FA. It is now also possible to force IP addresses for interfaces of the HA. This compensates the problem described above, wich appeared in version `0.7.0`. To maintain the integrity of Part I only Part II of this document has been updated to reflect these improvements.

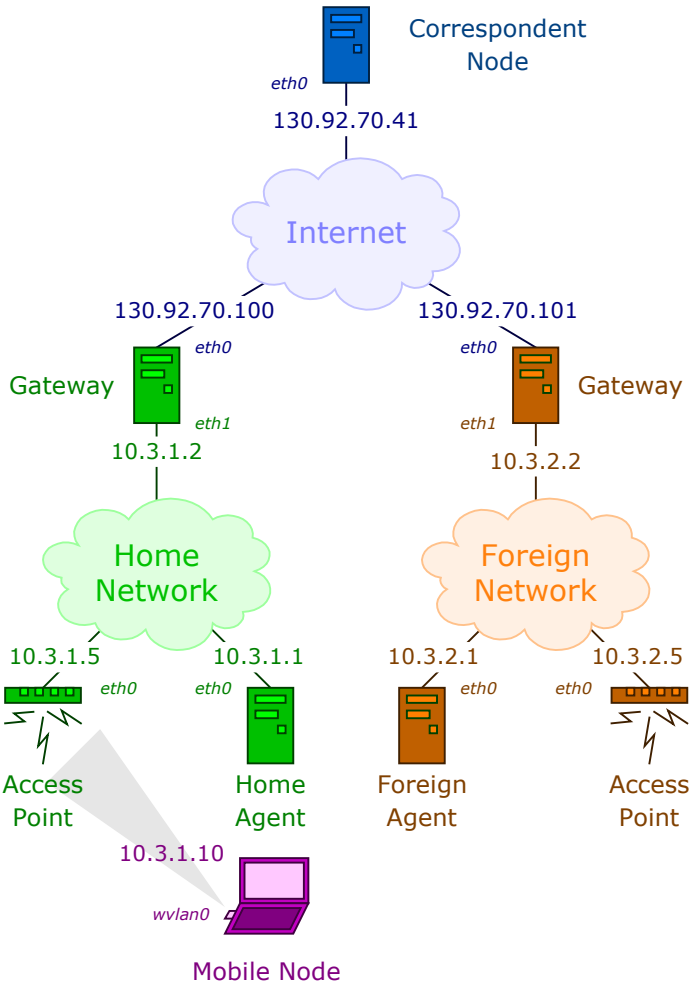

Figure 2: Test environment

4 Test

The Dynamics software has been tested by building up a connection from the MN to the Correspondent Node (CN) and then roaming through the different subnets. Meanwhile the time to update the current location of the MN was taken from logfiles created with `strace`.

For more detailed information about the use of `strace` see section 11.

To get optimal results measuring the switching time (i.e. to minimize interfering factors), the roaming has been done by using ethernet instead of wireless links and pulling the cable on a switch from one VLAN to the other.

When roaming in Real Life (e.g. physically moving from the 3rd [Home Network] to the 1st [Foreign Network] floor with the MN [see Figure 3]), the MN daemon keeps the link to the HA till the MN is really close to the Access Point (10 meters), although he received the AA of the FA earlier (20 meters). This behavior is caused by an internal procedure of comparing and selecting the links based on their quality.

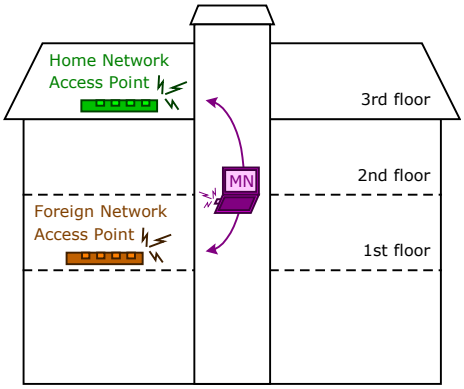

Figure 3: Roaming with the Mobile Node

5 Results

Figure 4 on page 12 illustrates the messages sent during the location update process. Starting in the Home Network the MN roams to the Foreign Network and back to the Home Network again. No Agent Solicitation message is shown because the HA and FA have been configured to send their Advertisements regardless to the receipt of Agent Solicitation messages.

The duration of a complete location update for different time intervals of the AA is shown for Reverse (Figure 5 on Page 13) and Triangle Tunneling (Figure 6 on Page 14).

It can be seen that there is a proportional relation between the time interval of the AAs and the time the daemons need to update the current location of the MN.

What was common too, is that upon first time change from the Home Network to the Foreign Network, the first Registration Reply never got accepted by the MN daemon (Wrong id field in the reply). This may be caused by a wrong starting value when calculating the Identification field (see Section 3.4 and 5.6 in [8] for in depth explanations).

References

[1] Dynamics - HUT Mobile IP. Project information available online from `http://www.cs.hut.fi/Research/Dynamics/`

[2] MosquitoNet Linux Mobile IPv4. Project information available online from `http://mosquitonet.stanford.edu/mip/`

[3] Binghamton Linux Mobile-IP. Project information available online from `http://anchor.cs.binghamton.edu/ mobileip/`

[4] Sun Mobile IP. Project information available online from `http://playground.sun.com/pub/mobile-ip/`

[5] HP Labs Mobile IP. Project information available online from `http://www.hpl.hp.com/personal/Jean_Tourrilhes/MobileIP/`

[6] NUS Mobile IP. Project information available online from `http://mip.ee.nus.edu.sg/`

[7] Secure Mobile Networking. Project information available online from `http://www.cs.pdx.edu/research/SMN/`

[8] C. Perkins. *IP Mobility Support*, RFC 2002, October 1996. Available online from `http://www.faqs.org/rfcs/rfc2002.html`

[9] IEEE. *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Std 802.11, 1999. Available online from `http://standards.ieee.org/getieee802/802.11.html`

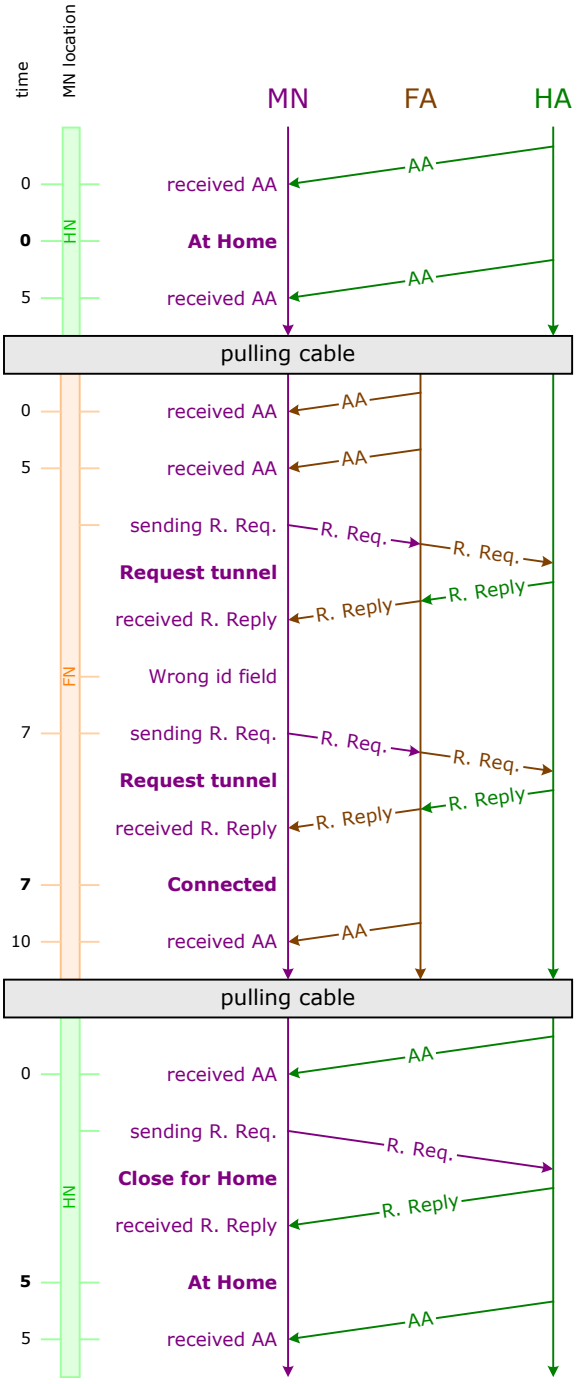

Figure 4: Message-Time Diagram

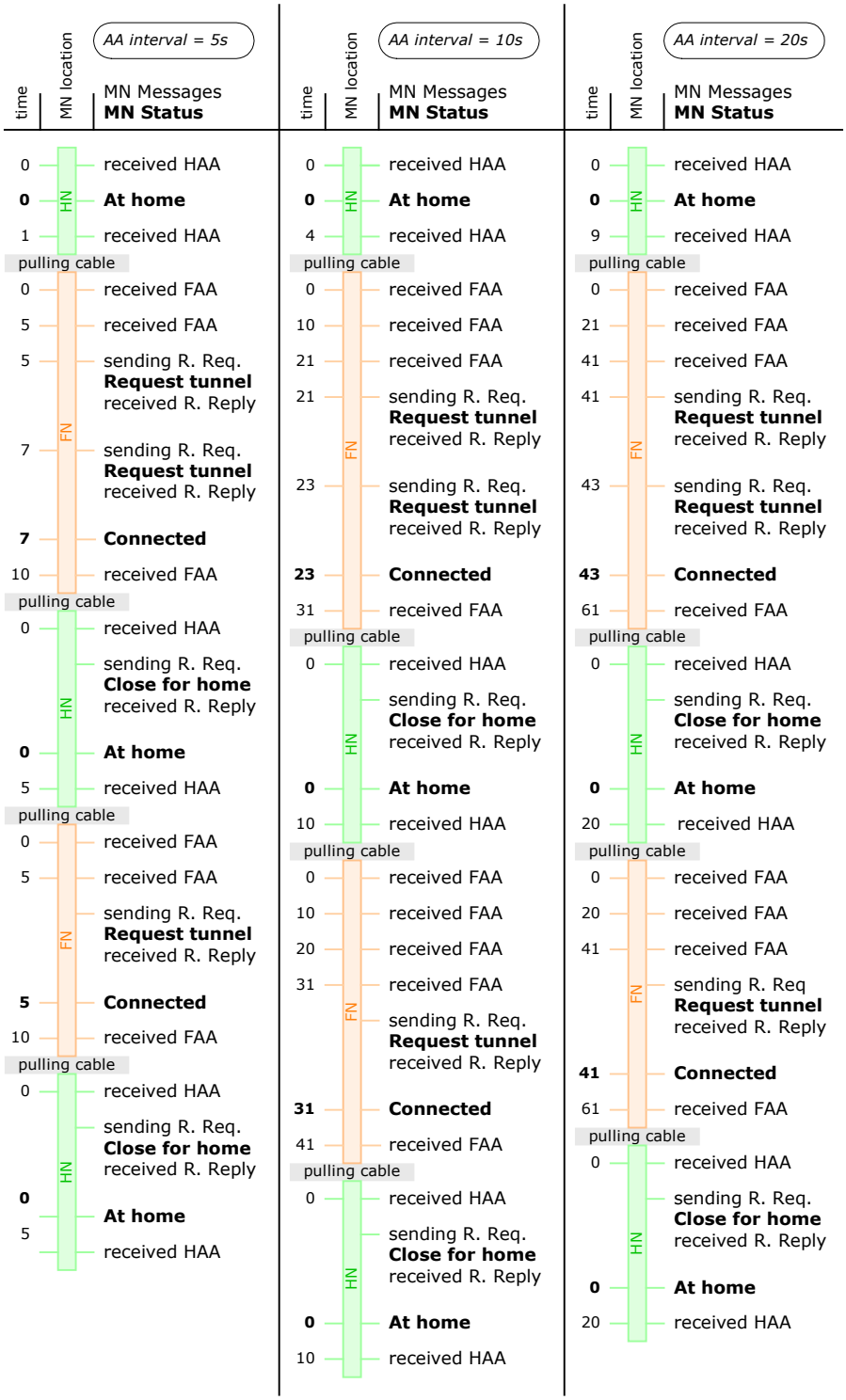

Figure 5: Reverse Tunneling

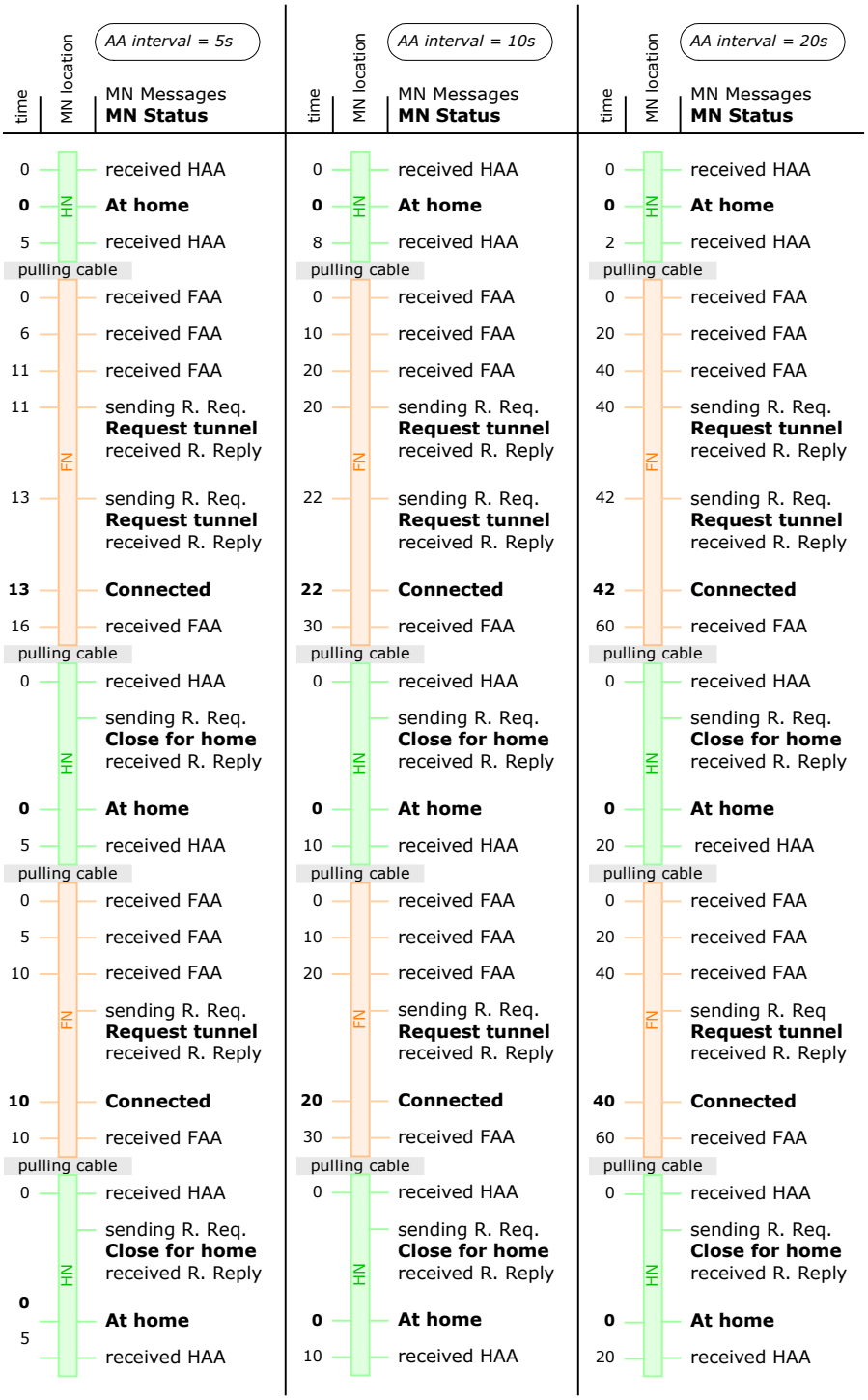

Figure 6: Triangle Tunneling

Part II
Mobile IP setup

6 Introduction

This part of the document guides through the process of establishing a Dynamics HUT software Mobile IP test environment under Linux.

For the sake of clarification important steps are marked with a special sign ✠ on the right side, like here.

6.1 Scenario overview

The scenario used is shown in Figure 7 on page 16.

6.2 Scenario description

The Home and Foreign Network (HN and FN) are both realized as Virtual LANs (VLAN) on a switch. The machines running thea Home and the Foreign Agent (HA and FA) act also as gateways to the corresponding subnet. The Access Points (APs) take care of Mobile Nodes (MNs) roaming through the covered administrative domain respectively.

The HA, FA and MN need to be correctly configured and run the corresponding daemon (dynmnd, dynhad and dynfad).

6.3 Hardware used

- 3 Linux-PCs (running Debian)
- 1 Latitude [DELL]
- 3 WaveLAN IEEE Turbo PC Card (Bronze) [Lucent]
  - Primary Functions Firmware: `4.00`
  - Station Functions Firmware: `7.28`
- 2 WavePoint-II Access Point [Lucent]
  - Firmware: 3.78
- 1 SmartSWITCH 6000 [Cabletron]

6.4 Software used

- Dynamics `0.7.1` [HUT]
- Linux Kernel `2.2.19`
- ntp `4.0.98`
- PCMCIA Card Services `3.1.26`
- strace `3.1.0.1`

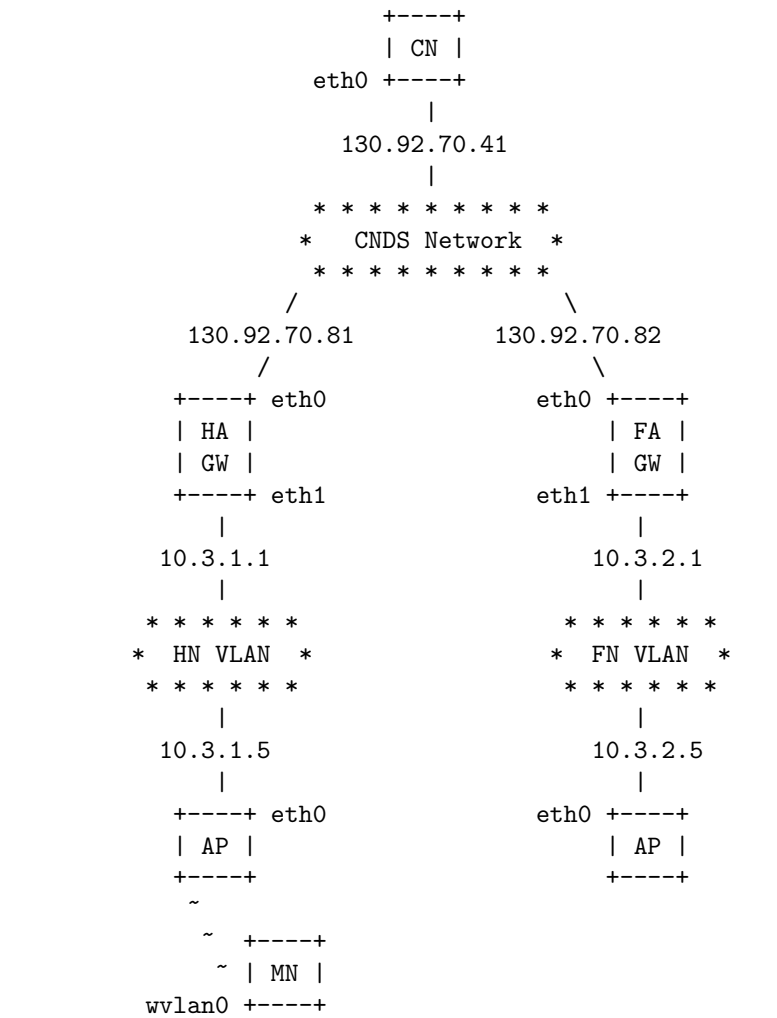

Figure 7: Scenario

- AP Manager `1.64` [Lucent]
- ORiNOCO Client Manager `1.76` [Lucent]

7 Kernel configuration

For each machine the kernel to be used and its additional compiling options is listed below.

- **Home Agent**

  Version: `2.2.19`
  Options:
  ```
  Networking options
    Packet socket (CONFIG_PACKET)
    Kernel/User netlink socket (CONFIG_NETLINK)
    Routing messages (CONFIG_RTNETLINK)
    Socket Filtering (CONFIG_FILTER)
    IP: tunneling (CONFIG_NET_IPIP)
  ```

- **Foreign Agent**

  Version: `2.2.19`
  Options:
  ```
  Networking options
    Packet socket (CONFIG_PACKET)
    Kernel/User netlink socket (CONFIG_NETLINK)
    Routing messages (CONFIG_RTNETLINK)
    Socket Filtering (CONFIG_FILTER)
    IP: advanced router (CONFIG_IP_ADVANCED_ROUTER)
    IP: policy routing (CONFIG_IP_MULTIPLE_TABLES)
    IP: tunneling (CONFIG_NET_IPIP)
  ```

- **Mobile Node**

  Version: `2.2.19`
  Options:
  ```
  Networking options
    Packet socket (CONFIG_PACKET)
    Kernel/User netlink socket (CONFIG_NETLINK)
    Routing messages (CONFIG_RTNETLINK)
    Socket Filtering (CONFIG_FILTER)
    IP: tunneling (CONFIG_NET_IPIP)
  Network device support
    Wireless LAN (non-hamradio) (CONFIG_NET_RADIO)
  ```

- **all machines**

  Version: `2.2.19`
  Options:
  ```
  Character devices
    Enhanced Real Time Clock Support (CONFIG_RTC)
  ```

8 Network configuration

The network devices of each machine need to be configured according to Figure 7 on page 16. For the PCs running Debian this can be done by editing the files `/etc/network/interfaces` and `/etc/network/options`.

On all involved PCs (HA, FA and MN) spoof protection must be disabled. ✠ This can be done in the last file mentioned above.

To gain additional speed during the "change of location" process the following value can be set accordingly.

```
/proc/sys/net/ipv4/route/min_delay      0
```

Furthermore, two routes for the outgoing packets of the two subnets have to be configured on the "main" router of the CNDS test network. Packets with `10.3.1.x` (`10.3.2.x`) as target address are sent to `130.92.70.81` (`130.92.70.82`).

The Access Points need to be configured using the Lucent Access Point Manager (AP Manager), which is running under Windows. To boot Windows on the MN just choose `win95` from `lilo`.

After starting the AP Manager follow these steps:

1. select the Access Point you want to configure from the list
2. click on Edit and enter the password ("public" by default)

   [a dialog (Edit Access Point configuration - *AP IP address*) with several tabs containing all the setting options appears]

All tabs mentioned in this and the next two sections (8.1, 8.2) refer to the last dialog opened in step 2.

Due to security reasons, currently only the following WaveLAN-cards are allowed to access the Access Points:

```
00:60:1D:04:2E:24  MN WaveLAN card
00:60:1D:04:2E:25  WC1 WaveLAN card
00:60:1D:04:2E:26  WC2 WaveLAN card
00:60:1D:04:32:96  WC3 WaveLAN card
00:60:1D:04:32:D6  tmp WaveLAN card
```

This list of MAC addresses can be changed under the Access Control tab.

Additionally these cards must have an IP address in one of the following subnets:

```
10.3.1.0/225.255.255.0
10.3.2.0/255.255.255.0
```

This SNMP IP Access List can be found under the SNMP tab.

An Access Point builds up a wireless cell. It is important to use different ✠ `Channel/Frequency` with a minimal distance of 25 MHz for each Access Point serving a different subnet, so the WaveLAN-card driver can distinguish between different cells and choose the nearest Access Point to communicate with.

Together the two Access Points (in the HN and FN) must cover a common ✠

wireless LAN. To do so their `Network Name`s must be equal. This can be checked under the Wireless Interfaces tab.

To ensure that the APs from the test network do not interfere with other ✠ APs from the public networks, the `Wireless System` needs to be closed.

A good source for more information is the *ORiNOCO Manager Suite User's Guide* [3].

8.1 Home Network

- **Access Point**

```
Name                    :          AP_Cell_1
Description             : WavePOINT-II V3.78
wvlan0 : MAC            :  00:60:1D:04:2E:25
         Firmware       :               7.28
eth0   : MAC            :  00:60:1D:03:DA:23
```

under the Wireless Interfaces tab, in the PC card slot A section, click the Advanced button:

```
Network Name            :        MIP_Network
Channel/Frequency       :        1/2.412 GHz
Distance Between APs    :             Large
```

under the Wireless Interfaces tab, in the PC card slot A section, click the Security button:

```
Close Wireless System   :                 x
```

under the Access Point IP tab, in the Specify an IP address section:

```
Access Point IP Address  :           10.3.1.5
Access Point Subnet Mask :      255.255.255.0
```

- **Home Agent**

under `/etc/network/` edit the following files

```
interfaces     iface eth0 inet static
                 address         130.92.70.81
                 netmask        255.255.255.0
                 network          130.92.70.0
                 broadcast      130.92.70.255
                 gateway          130.92.70.1
               iface eth1 inet static
                 address             10.3.1.1
                 netmask        255.255.255.0
                 network             10.3.1.0
                 broadcast         10.3.1.255

options        ip_forward                yes
               spoofprotect               no
```

8.2 Foreign Network

- **Access Point**

```
Name                      :          AP_Cell_2
Description               : WavePOINT-II V3.78
wvlan0 : MAC              :  00:60:1D:04:2E:26
         Firmware         :                7.28
eth0   : MAC              :  00:60:1D:F4:40:2C
```

under the Wireless Interfaces tab, in the PC card slot A section, click the Advanced button:

```
Network Name              :        MIP_Network
Channel/Frequency         :        7/2.442 GHz
Distance Between APs      :              Large
```

under the Wireless Interfaces tab, in the PC card slot A section, click the Security button:

```
Close Wireless System     :                  x
```

under the Access Point IP tab, in the Specify an IP address section:

```
Access Point IP Address   :           10.3.2.5
Access Point Subnet Mask  :      255.255.255.0
```

- **Foreign Agent**

under `/etc/network/` edit the following files

```
interfaces      iface eth0 inet static
                  address         130.92.70.82
                  netmask        255.255.255.0
                  network          130.92.70.0
                  broadcast      130.92.70.255
                  gateway          130.92.70.1
                iface eth1 inet static
                  address             10.3.2.1
                  netmask        255.255.255.0
                  network             10.3.2.0
                  broadcast         10.3.2.255

options         ip_forward                yes
                spoofprotect               no
```

8.3 Roaming unit

- **Mobile Node**

under `/etc/pcmcia/` edit the following files

```
network.opts    *,*,*,*)
                  IPADDR              10.3.1.10
                  NETMASK          255.255.255.0
                  NETWORK               10.3.1.0
                  BROADCAST           10.3.1.255
                ;;

wireless.opts   *,*,*,*)
                  ESSID             MIP_Network
                  MODE                  managed
                  RATE                     auto
                ;;
```

under `/etc/network/` edit the following file

```
options         spoofprotect                no
```

9 Dynamics installation and configuration

This section will cover the installation and setup of the Dynamics software.

9.1 Installation

The Dynamics software is available in two versions. The distribution from the stable branch (`0.6.2`) provides a well tested base and is meant for end users. The most recent (`0.7.1`) comes from the developer branch and includes all new features, but is less tested.

The source code is provided for both branches, binary versions are also available for the stable branch.

In this scenario the developer branch (`0.7.1`) was chosen. To install the source code distribution from a zipped tar archive follow these steps:

1. do `tar -xzf dynamics-X.tar.gz` (where X stands for the version)
2. run `./configure`
3. do `make`
4. do `make install`

tools, daemons	are placed in	`/usr/local/sbin`
configuration files	-"-	`/usr/local/etc`

9.1.1 Network Time Protocol

It is highly recommended to install a NTP daemon on each machine involved (HA, FA and MN). Since timestamps are used as replay protection method synchronised clocks are inevitable.

During the tests `time.atnet.at` has been a reliable NTP server.

9.2 Basic Configuration

The basic configuration can be done by starting the setup scripts from Dynamics on each machine and entering the given values below respectively.

- **Mobile Node** (`dynamics-mn-setup`)

```
Mobile Node IP address       :      10.3.1.10
Home Agent IP address        :       10.3.1.1
Home network's address       :       10.3.1.0
Home network's prefix length :             24
```

  (Note that the script doesn't change the last two entries by itself) ✠

- **Home Agent** (`dynamics-ha-setup`)

```
Home Agent IP address        :       10.3.1.1
Mobile Node IP address       :      10.3.1.10
```

- **Foreign Agent** (`dynamics-fa-setup`)

```
hierarchical/simple FA (h/s) :              s
Foreign Agent IP address     :       10.3.2.1
```

  A RSA key also needs to be created for the FA. The RSA public key file can be generated with the following command: ✠

  `rsakeygen /etc/dynfad.key 128`

  (rsakeygen is included in the Dynamics distribution)

This should provide a simple working scenario. In the following sections you will find a more detailed explanation of the configuration process.

9.3 Advanced configuration

In the configuration files (`dynmnd.conf`, `dynhad.conf` and `dynfad.conf`) more parameters can be found to alter the daemon's behavior. Some examples of their use are listed in the following sections.

9.3.1 Tunnel modes

To choose the `Tunneling Mode` the following entries in the corresponding configuration file have to be altered:

- **Mobile Node** (`dynmnd.conf`)

```
# 1 = automatic, prefer reverse tunnel (bi-directional)
# 2 = automatic, prefer triangle tunnel (only CN->MN)
# 3 = accept only reverse tunnel
# 4 = accept only triangle tunnel
TunnelingMode < 1 | 2 | 3 | 4 >
```

- **Home Agent** (`dynhad.conf`)

```
EnableTriangleTunneling < TRUE | FALSE >
EnableReverseTunneling < TRUE | FALSE >
```

- **Foreign Agent** (`dynfad.conf`)

```
EnableTriangleTunneling < TRUE | FALSE >
EnableReverseTunneling < TRUE | FALSE >
ForceReverseTunneling < TRUE | FALSE >
```

Enabling only the tunnel mode which will be used (disabling the other) is always the safest way to make sure the daemons behave as expected. ✠

9.3.2 Agent Advertisements

The dispatch of the `Agent Advertisements` (AA) from the HA and the FA can be set with the following key words:

- **Home Agent** (`dynhad.conf`)

```
# ha_disc:
#   0 = do not allow dynamic HA discovery
#   1 = allow dynamic HA discovery with broadcast messages
#
# agentadv:
#   0 = do not send AA without agent solicitation
#   1 = send agent advertisements regularly
#
# interval: (in seconds)
#
INTERFACES_BEGIN
# interface  ha_disc  agentadv  interval  force_IP_addr
eth1         1        1         5
INTERFACES_END
```

- **Foreign Agent** (`dynfad.conf`)

```
# type:
#   1 = both upper and lower direction
#   2 = only upper direction (to upper FA / HA)
#   3 = only lower direction (to lower FA / MN)
#
# agent advertisements:
#   0 = do not send AA without agent solicitation
#   1 = send AA regularly
#
# interval: (in seconds)
```

```
#
INTERFACES_BEGIN
# interface  type  agentadv  interval  force_IP_addr
eth0         2     1         5         10.3.2.1
eth1         3     1         5
INTERFACES_END
```

The interval in which AAs are sent is directly proportional to the time needed by the HA and FA daemons to realize the change of location of the MN. Setting the interval to 5 seconds allows a fast recognition when arriving into a new area, though the administrative network traffic caused by the advertisement packets may reach unintended size.

9.3.3 Security Parameter Index and Shared Secret

By default a `Shared Secret` between MN and HA is established. Note that the `Security Parameter Index` (SPI) must be associated with the IP address from the MN inside the section `AUTHORIZEDLIST` of the HA configuration file.

- **Mobile Node** (`dynmnd.conf`)

```
# SPI < number >
SPI  1000

# SharedSecret < HEX number string or character string >
SharedSecret  "MN2HA"
```

The `timestamp tolerance` sets the accepted difference between MN and HA time in seconds. All packets exceeding this limit will be rejected. It is therefore important to make sure that the system clocks of all involved machines (especially the Laptop) are in sync.

- **Home Agent** (`dynhad.conf`)

```
AUTHORIZEDLIST_BEGIN
# < SPI | SPI-range      IP | network/netmask >  (n:n)
# SPI           IP
1000            10.3.1.10
AUTHORIZEDLIST_END

SECURITY_BEGIN
# Replay Protection Method:
#   0: none
#   1: timestamps
#   2: nonces
#
#       auth.   replay  timestamp     max          shared
# SPI   alg.    meth.   tolerance     lifetime     secret
1000    1       1       100           600          "MN2HA"
```

```
SECURITY_END
```

A `Shared Secret` between the HA and FA can be defined as follows:

- **Home Agent** (`dynhad.conf`)

```
FA_SECURITY_BEGIN
# SPI           FA IP           Alg.    Shared Secret
2000            10.3.2.1        1       "HA2FA"
FA_SECURITY_END
```

- **Foreign Agent** (`dynfad.conf`)

```
FA_SECURITY_BEGIN
# Agent type:
#   1 = FA
#   2 = HA
#   3 = MN
#
# SPI         FA IP          Agent   Alg.    Shared Secret
2000          10.3.1.1       2       1       "HA2FA"
FA_SECURITY_END
```

Similarly a `Shared Secret` between the MN and the FA can be added:

- **Mobile Node** (`dynmnd.conf`)

```
FA_SECURITY_BEGIN
# SPI           FA IP           Alg. Shared Secret
2001            10.3.2.1        1      "MN2FA"
FA_SECURITY_END
```

- **Foreign Agent** (`dynfad.conf`)

```
FA_SECURITY_BEGIN
# Agent type:
#   1 = FA
#   2 = HA
#   3 = MN
#
# SPI         FA IP          Agent   Alg.    Shared Secret
2000          10.3.1.1       2       1       "HA2FA"
2001          10.3.1.10      3       1       "MN2FA"
FA_SECURITY_END
```

Currently the only (default) authentication algorithm implemented is keyed MD5 (Alg. 1).

The section `AUTHORIZEDNETWORKS` contains a list of allowed IP addresses from which registration requests can be sent to the FA. For security reasons one should alter the default entry, which grants access to all.

- **Foreign Agent** (`dynfad.conf`)

```
AUTHORIZEDNETWORKS_BEGIN
# [ networkaddress ]/[ netmask ]
0.0.0.0/0.0.0.0
AUTHORIZEDNETWORKS_END
```

9.3.4 Network Access Identifiers

Although it is not absolutely necessary for this scenario (according to [3]), the usage of the Network Access Identifiers (NAIs) is recommended, since it reduced the occurrence of some strange authentication errors.

- **Mobile Node** (`dynmnd.conf`)

```
# HANetworkAccessIdentifier < string >
HANetworkAccessIdentifier "HN_MIP_CNDS"
```

- **Home Agent** (`dynhad.conf`)

```
# NetworkAccessIdentifier < string >
NetworkAccessIdentifier "HN_MIP_CNDS"
```

- **Foreign Agent** (`dynfad.conf`)

```
# NetworkAccessIdentifier < string >
NetworkAccessIdentifier "FN_MIP_CNDS"
```

10 Running the daemons

To run the daemons just execute the corresponding executable located in `/usr/local/sbin`

- **Mobile Node** (`dynmnd`)
- **Home Agent** (`dynhad`)
- **Foreign Agent** (`dynfad`)

Make sure that the flags mentioned in section 8 on page 18 are set correctly. A small script (`check_flags`) is being provided to print these flags in the following order:

`ip_forward`
`rp_filter` for eth0
`rp_filter` for eth1 (if configured)
`min_delay`

If you encounter problems (i.e. the daemons are not working properly) have a look at section 12.

11 Monitoring and debugging the daemons

Each daemon has a monitoring tool (`dynmn_tool`, `dynha_tool` and `dynfa_tool`) which allows to get detailed information about the current state of the program. Entering

```
st 1
```

on its command line will continuously display status information. For intense debugging purposes, the Dynamics specific environment variable should be set to print all debug information with the following command

```
export DYNAMICS_DEBUG=-
```

and start the daemons with the `--debug --fg` parameters. Moreover I used strace to see all network activities and catch the daemon's output by using the `-e trace=network,write` option:

```
strace -ff -tt -x -o/out.log -e trace=network,write dynmnd
--fg --debug
```

With tcpdump one can listen to the packets on the network. Basically only UDP (registration [request, reply]) and ICMP (agent advertisements) packets are of interest.

```
tcpdump -i eth0 udp or icmp
```

12 Troubleshooting

This checklist sums up the most important things to do in case of problems with this scenario.

- first of all read and verify the advices in `dynamics_config_checklist` under `dynamics-0.7.1/doc/`
- make sure that `rp_filter` is set to 0 (Section 9.2)
- (save your settings and re)start with the Basic Configuration (Section 9.2)
- use debbuging flags (Section 11) and search for the error message
- look up the error code returned by the daemons in the RFC2002
- compare the system clocks of MN and HA and check if the `timestamp tolerance` in the `dynha.conf` is accurately set (Section 9.3.3)
- choose different `Channel/Frequency` for each Access Point (Section 8)
- choose an equal `Network Name` for each Access Point (Section 8)
- read the other documents listed in [2]
- browse or subscribe to the Dynamics users' mailing list, available online under `http://www.cs.hut.fi/Research/Dynamics/mailinglist.html`

13 Conclusion

All files and programs mentioned in this document are available under `/project` on the Laptop or in my home directory (`/home/weyland/project`).

```
/project
/project/MIP
/project/MIP/bin          binary distributions of Mobile IP
                          software
/project/MIP/conf         important configuration files from
                          the test machines
/project/MIP/doc          general Mobile IP documents
/project/MIP/doc/dynamics Dynamics documents
/ptoject/MIP/doc/lucent   Lucent documents
/project/MIP/doc/results  project documentation, presentation,
                          source diagramms, tables, log-files
/project/MIP/src          source code distribution of Mobile
                          IP software
/project/src              source code of other used software
```

References

[1] D. Forsberg: *Dynamics - HUT Mobile IP-HOWTO*, October 1999

[2] B. Andersson, D. Forsberg, J. Malinen and T. Weckström: *Dynamics Agent installation and configuration files:*

- `INSTALL`
- `README`
- `dynamics_config_checklist`
- `dynfad.conf`
- `dynmnd.conf`
- `dynhad.conf`
- `debug_flags`

[3] Lucent Technologies: *ORiNOCO Manager Suite User's Guide*, August 2000. Latest version available online from `http://www.lucent.com/orinoco` or from my home directory under `/project/MIP/doc/lucent/ug_OM.pdf`