Abstract

Implementation, configuration and analysis of a Mobile IP network, as specified in RFC3220 of IETF. The IPv4 implementation with a mobility support was made using Software Dynamics version 0.8.1 developed by the Helsinki University of Technology. Among other aspects, macro and micromobility solutions are studied.

IP MOBILITY – A PRACTICAL APPROACH

D. Costa [a], F. Santos [a], J. Carvalho [b], L. Faria [a]

[a] *INESC Porto - UTM , Praça da República 93, 4050-497 Porto, Portugal*
[b] *INESC Porto - UOSE , Rua do Campo Alegre 687, 4169-007 Porto – Portugal*

I. INTRODUCTION

The portability, the mobility and the wireless accesses are more and more a necessity and a demand from users of data communications networks. The Internet Protocol (IP) does not sustain either portability or mobility, due to the fact that the packets routing to their destination is accomplished according to its IP address. These addresses are associated to a fixed network location, thus when the mobile node is in a network which is not its own can't maintain its address, which makes impossible for the mobility to be transparent. In order to skirt this IP limitation, the Internet Engineering Task Force (IETF) specified a protocol to solve the IP macromobility, the Mobile IP. The Mobile IP was in this manner designed to solve the problem of macromobility, portability, letting a user have connectivity to the Internet and work in a network which is not his, as if he were in his own. However, recently, with the wireless networks appearance, its fast growth, the constant change of the access point by the users, and the mobility demand by them again, had Mobile IP been thought over, having been presented several proposals to try to solve the micromobility problem, for instance the Cellular IP, the Hawaii and the Hierarchical IP. The micromobility protocols are designed for surroundings where the mobile nodes change frequently their connection point, and the basic Mobile IP, with its encapsulation mechanism, inserts an overhead in the network, in terms of delays, lost of packets and signaling.

By this article we propose ourselves to analyze the Mobile IP and Hierarchical IP behavior. On section II is made a generic description of the Mobile IP protocol. Following on, on section III we expose the Hierarchical IP and we describe its functioning. Thereafter it is presented the simulation model of hierarchic scenery on section IV, being presented results and the analyzed performance in section V. Finally, in section VI we will discuss some relevant aspects by presenting some evident conclusions.

II. MOBILE IP FUNCTIONING

The secret of Mobile IP is that all the functionalities for the processing and management of the information about mobility are enraptured in well defined entities, the Home Agent, Foreign Agent, and Mobile Node. The actual protocol, the Mobile IPv4, is completely transparent for the other network layers and also it does not require any changes in what comes to host and routers.

The Mobile IP allows Mobile nodes to have two IP addresses. The first one called Home Address, is static, that is to say it is permanently bestowed to Mobile Node, remaining unchangeable besides the connection point of Mobile Node to the Internet. This address is used to identify higher-level connections. The 2nd IP address is the Care-of-Address, this address identifies the actual location of the Mobile Node, therefore, when the Mobile Node changes its connection point, the Care-of-Address is managed by the Foreign Agent Entity.

The Home Address throws out a hint that Mobile Node is continuously available to receive data in its local network, through a network node known as Home Agent. Whenever the Mobile node is connected to a foreign network the Home Agent receives all the packets appointed to the Mobile Node and delivers them at the actual point of attach of

the Mobile Node, which means, in its Care-of-Address. Whenever the Mobile Node changes to another location, registers its new Care-of-Address in its Home Agent.
To deliver a packet at a Mobile Node, the Home Agent sends a packet to the Care-of-Address. This delivery needs the packet to be encapsulated inside another IP packet, so that the Care-of-Address appears as the destination IP address. This encapsulment is also called as tunneling. When the packet arrives at the Care-of-Address, the packet is decapsulated so that the Home Address appears as the destination IP. When the packet arrives at the Mobile Node, addressed to the Home Address, is processed correctly by the higher layers, which receive the IP layer packets in the Mobile Node.

III. HIERARCHICAL IP

The Mobile IP showed that it is unsuitable for the micromobility, because the FA constant changes take us to successive records in the HA, which makes the number of signaling packets increase in the network. If the HA is distant, the delays can become elevated which can result in a big percentage of lost packets, addressed to the MN.
The Hierarchical IP was deliberate to solve these problems, maintaining all the functionalities of the processing in well-defined entities. Home Agent, Foreign Agent and Mobile Node, to the resemblance of the Mobile IP, adding a hierarchy in FAs trees.

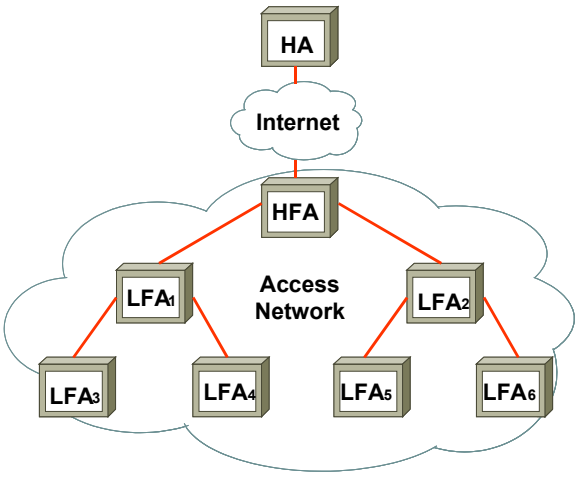

Figure 1

The main advantage of the Hierarchical IP over the Mobile IP relies on the fact that a MN can freely move among several LFAs (*Leaf Foreign Agents*), without having the register messages out of the access network, that is, after the 1st register in the HA, the MN only registers itself again in the HA when enters in a new access network.
This implies from a very instant, a minor traffic in the external network and a minor delay because the distances are smaller.
This is possible although the MN, as it tries to register in a new LFA, send a register message to the HA, but this one doesn't get there, once it is intercepted as it gets to the node, where the relating LFA has already an entrance at the index table for the MN, in worst case this is the HFA (Highest Foreing Agent). This answers with a message, pretending to be the register answer of the HA. Thus the MN moves among LFAs without having the HA noticed and the MN destinated packets delivery task is delivered to the foreign network.
All the nodes of the network have an up-to-date index table of the MN addresses, which are associated to the relative nodes.
Every time a MN connects to a new LFA, it sends a message from node to node until the old LFA with a purpose which is to remove the MN address from its index table,
Among several LFAs from the network and HFA the sent packets to the MN are conducted in tunnels until the last LFA.

IV. SIMULATION MODEL

The simulation scenery used to test the hierarchical IP is shown in the following figure.

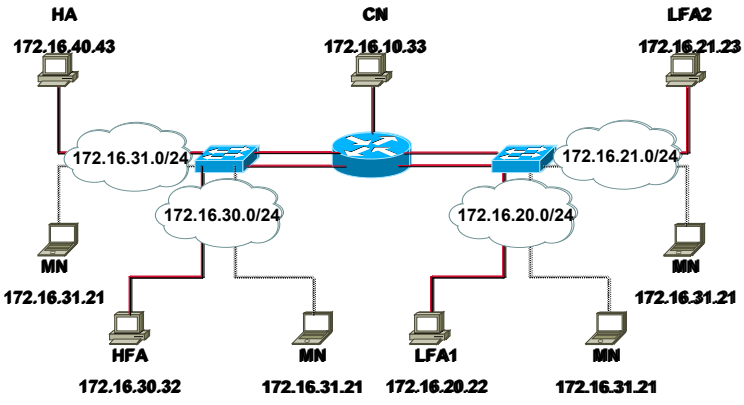

Figure 2

The tests were done with a network all of it built with a 100Mbps. All the PCs and the active network equipment (routers and

switches) have 100Mbps network interfaces cards.
Although we've made an hierarchic setting, we can also analyze the performance of macromobility, because when the mobile node is connected to the HFA network it is the same as having a non-hierarchic setting.
The software that implements the Mobile IP and the Hierarchical IP is the Dynamics version 0.8.1 developed by the Helsinki University of Technology.
To configurate the Mobile IP entities we used the respective setup tool for all the entities. This tool changes the IP addresses in the configuration files. We also change the MNDefaultTunnelLifetime from 300 to 30 on the file dynmnd.conf.

V. EXPERIMENTAL RESULTS

The practical results of the simulation described before are presented in the following tables. When in there is done a reference to the HA, HFA, LFA1, LFA2, it means that at that moment the MN is connected to the network of that node.

	CN → MN			MN → CN		
	MIN (ms)	AVG (ms)	MAX (ms)	MIN (ms)	AVG (ms)	MAX (ms)
HA	0.9	1.0	1.8	0.9	0.9	1.2
HFA	1.9	2.1	3.2	1.9	2.2	14.7
LFA_1	2.1	2.3	3.3	2.2	2.6	17.8
LFA_2	2.1	2.4	12.2	2.1	2.3	2.4

Table 1

FTP – 20 Mbytes		
	CN → MN	MN → CN
HA	5.05 s	18.68 s
HFA	22.26 s	22.16 s
LFA_1	22.75 s	23.19 s
LFA_2	22.70 s	22.89 s

Table 2

The tables 1 and 2 were obtained by accomplishing the command '`ping -c 50 172.16.31.21`' in this situation CN → MN and '`ping -c 50 172.16.10.33`' on the opposite side, MN → CN, both after the MN being stabilized, i.e., after being registered in that network, not having packets lost whatsoever.

When the MN is at home network, either the communication works as CN → MN, or the opposite way, there are no big differences, between the round trip time, which was something expected for in its Home Network the protocol is the normal IP, in what comes to the others, the difference is not also very high, for it was being done a reverse tunneling which imply that the packets path is equal on both senses.
The maximum timings, which sometimes appear way superior to the other tables stocks, may correspond to binding messages which happened in course of the 50 pings.
To see the difference between timings when the transfer is done from or to the MN, the best data are those file transfer protocol (FTP) ones, once they are transferred nearly 20MB which makes the results more reliable due to the average effect. Whenever the Mobile node is connected at a lower level the round trip time increases slightly which was something to be expected, because the number of tunnels increase as well.

The following tables (3 to 10) were obtained with the command '`traceroute 172.16.31.21`' in this situation CN → MN and '`traceroute` 172.16.10.33' in a different situation MN → CN and made being the MN in the network shown in the table.

CN → MN				
HA		1	2	3
1.	172.16.10.254 (R)	1.200 ms	1.147 ms	1.016 ms
2.	172.16.31.21 (MN)	1.735 ms	1.811 ms	0.875 ms

Table 3

MN → CN				
HA		1	2	3
1.	172.16.31.254 (R)	1.314 ms	1.710 ms	1.530 ms
2.	172.16.10.33 (CN)	0.966 ms	1.060 ms	0.886 ms

Table 4

CN → MN				
HFA		1	2	3
1.	172.16.1.254 (R)	1.221 ms	1.138 ms	1.019 ms
2.	172.16.31.31 (HA)	0.984 ms	0.989 ms	0.854 ms
3.	172.16.30.32 (HFA)	1.113 ms	1.191 ms	0.994 ms
4.	172.16.31.21 (MN)	2.626 ms	2.727 ms	2.448 ms

Table 5

MN → CN				
HFA		1	2	3
1.	172.16.30.32 (HFA)	0.615 ms	0.402 ms	0.331 ms
2.	172.16.31.31 (HA)	1.297 ms	1.298 ms	1.097 ms
3.	172.16.31.254 (R)	2.092 ms	2.149 ms	1.960 ms
4.	172.16.10.33 (CN)	1.850 ms	2.070 ms	1.813 ms

Table 6

CN → MN				
LFA_1		1	2	3
1.	172.16.10.254 (R)	1.235 ms	1.111 ms	0.989 ms
2.	172.16.31.31 (HA)	0.925 ms	0.987 ms	0.823 ms
3.	172.16.30.32 (HFA)	1.099 ms	1.175 ms	0.958 ms
4.	172.16.20.22 (LFA_1)	1.218 ms	1.337 ms	1.084 ms
5.	172.16.31.21 (MN)	2.094 ms	2.330 ms	2.001 ms

Table 7

MN → CN				
LFA_1		1	2	3
1.	172.16.20.22 (LFA_1)	0.587 ms	0.419 ms	0.330 ms
2.	172.16.30.32 (HFA)	0.820 ms	0.647 ms	0.577 ms
3.	172.16.31.31 (HA)	1.626 ms	1.558 ms	1.345 ms
4.	172.16.31.254 (R)	2.382 ms	2.450 ms	2.241 ms
5.	172.16.10.33 (CN)	2.130 ms	2.360 ms	2.050 ms

Table 8

CN → MN				
LFA_2		1	2	3
1.	172.16.10.254 (R)	1.235 ms	1.135 ms	1.025 ms
2.	172.16.31.31 (HA)	0.933 ms	0.995 ms	0.855 ms
3.	172.16.30.32 (HFA)	1.089 ms	1.205 ms	0.990 ms
4.	172.16.21.23 (LAF_2)	1.187 ms	1.243 ms	1.090 ms
5.	172.16.31.21 (MN)	2.831 ms	2.975 ms	1.969 ms

Table 9

MN → CN				
LFA_2		1	2	3
1.	172.16.21.23 (LFA_2)	0.588 ms	0.373 ms	0.295 ms
2.	172.16.30.32 (HFA)	0.695 ms	0.646 ms	0.529 ms
3.	172.16.31.31 (HA)	1.555 ms	1.554 ms	1.301 ms
4.	172.16.31.254 (R)	2.274 ms	2.422 ms	2.154 ms
5.	172.16.10.33 (CN)	2.230 ms	3.106 ms	2.025 ms

Table 10

With the traceroute done from the MN we can conceive that the software is ready accomplishing reverse tunneling once it always appeals to the HA to routing packets. At the tables regarding the command traceroute, the round trip time, in the case of router (R), seems to be absurd, because there is no encapsulation nor decapsulation and the round trip time is superior to the rest of the nodes where there's encapsulment. This is due to the fact that the routers are optimized to process normal packets and in case the packets can't be normally processed, which means, in case they are not sent instantly to the output port, which can be the TTL case = 0, they must be processed aside by the microprocessor router, which is far slower than the used PCs processor.

100 packets	MIN	AVG	MAX	Received
HA → HFA	1	1.8	4.9	47
HA → LFA_1	0.9	2.0	2.9	52
HA → LFA_2	0.9	2.0	3.1	52
HFA → LFA_1	1.9	2.3	9.8	40
HFA → LFA_2	1.9	2.1	13.7	65
LFA_1 → LFA_2	2.1	2.3	9.2	80

Table 11

The table 11 was obtain with the command 'ping –c 100 172.16.31.21', being the transfer interrupted at 10 packets passing the MN to the indicated local and waiting that the ping recovers.

From the analysis of the table we can see that the packets lost is normally higher than 50 packets or close to that, which tells us that this software packet is not indicated to solve micromobility cases, although in situations as the exchange of the connecting point between leaves (LFA1 and LFA2) the packets lost being considerably lower inn about 20 packets is still very elevated for any type of application and specially for the real time. These elevated losts are due to the fact that the MN cannot register in the network, to which it connects, whereas it doesn't occur binding timeout. This restriction is not contemplated in the RFC3220, where is mentioned that the MN shall be capable to register at any moment, and not just when the binding expires. This restriction is strictly due to the implementation of the used software.

These data have incongruences, because the packets losts are influenced by the binding timeouts and if they occur close the reconnection moment, the packets lost can be considerably lower. But to a global vision of the functioning, these data serve perfectly and to reduce this effect were realized three pings to each situation, being later found the average. The timings from the tables have no meaning to the time durability analysis of the handoff, once the ping only accounts the packets timings which are sent and received correctly.

VI. CONCLUSIONS

The used software fulfils wonderfully the purposes, to which the Mobile IP itself purposes, as so to say macromobility.

The hierarchical IP improves the Mobile IP performance in what comes to the micromobility level and the frequent exchange of the connection point. Yet it can't be used for the demands to the micromobility level, once it looses lots of packets, becoming thus very difficult a session continuance with the minimum QoS, and if the reconnections are too frequents, the connections may indeed fail.

VII. REFERENCES

[1] IETF, *"IP Mobility Support"*, RFC 2002, October 1996.
[2] IETF, “IP Mobility Support for Ipv4”, RFC 3220, January 2002
[3] Charles Perkins, *“Mobile IP"*, IEEE Communications Magazine, May 1997.
[4] Open Report, Mobile IP, 3/0362-FCP NB 102 88 Uen, Ericsson, 1999
[5] Charles E. Perkins, *“Mobile Networking Through Mobile IP”*, Sun Microsystems, 1998
[6] Andrew Campbell, Javier Gomez, Sanghyo Kim, Chieh-Yih Wan, Zoltan Turanyi and Andras G. Valko, *"Comparison of IP Micromobility Protocols"*, IEEE Wireless Communications, Feb. 2002.
[7] Jochen Schiller, "Mobile Communications", Addison-Wesley
[8] IETF,”Reverse Tunnelling for Mobile IP”, RFC 2344, May 1998
[9] Dan Forsberg, Jari T. Malien, Jouni K. Malien, Tom Weckstrom, “Dynamics – Hut Mobile IP v0.6”, 29 October 1999.
[10] http://www.cs.hut.fi/Research/Dynamics/