Performance Implications of the Multi Layer Mobility in a Wireless Operator Networks

Jouni Korhonen*
Helsinki University
Computer Science
Berkeley Summer School 2004
Email: Jouni.korhonen@helsinki.fi

***Abstract* – This paper studies Mobile IPv4 based handover delays in a live operator network in the Mobile Node and handover signaling point of view. In our measurements the Mobile Node always used a VPN to connect its Home Network and the WLAN authentication was EAP-SIM based. Measurements were done in both UMTS and WLAN networks. Our measurements show that in general deployments Mobile IPv4 handovers are too slow for services having time constraints. The Mobile Node software should be tightly integrated, and the access network infrastructure should be designed to allow fast and smooth horizontal and vertical handovers.**

I. INTRODUCTION

Recent developments in IP mobility and secure operator wireless networks have introduced network architectures, where the mobility and the security take place on several layers at the same time. The reason for multi layer IP mobility [11] is to ease the deployment of the secure IP mobility solutions to existing network infrastructures and make the migration gradual. One of the drivers for this development has been the recent introduction of multi-access capable mobile terminals. These terminals are equipped with two or more different network access technologies. Also there are more and more overlapping operator controlled networks. Another driver for the multi layer IP mobility solutions has been the security and known problems of the existing deployed IP security technologies. Current IP security solutions cannot handle IP address change during the user session. Also in the near future access networks will have both IPv4 and/or IPv6 capability. IP version migration must also be solved in these heterogeneous access networks. Mobile terminals should be able to roam from one access network to another independent of underlying IP version.

For a mobile operator a combination of a short range public access wireless networks and wide area cellular networks are an interesting area for new services. These services should also allow a secure corporate access for corporate users. Mobile operators also would like to combine the same AAA mechanism to all used accesses, which basically means all new accesses should use SIM for AAA. Especially WLAN access combined with SIM based AAA is a technology that mobile operators are looking forward to deploy.

One of the planned services for WLAN networks is Voice over IP. WLAN has also been seen as a convenient way of extending wireless coverage especially in indoors premises. 3GPP is standardizing interworking system between the 3G network and WLAN access [1][8]. For cellular operators 3GPP defined 3G-WLAN interworking system will be the way how WLAN access gets deployed. Before the standardized systems are ready for deployment there will be intermediate solutions, which this paper studies. Again the key common area between 3GPP standardized system and the intermediate systems are AAA and roaming between accesses.

The combination of SIM based AAA, IP level security, and IP mobility that is aware of different IP versions in access networks create a networking environment where signaling takes place on several layers. IP mobility and various tunneling mechanisms may also be used on several layers, which mean a considerable amount of header overhead. A Number of signaling messages must be exchanged between the mobile terminal and different network nodes before the first data packet is allowed to be sent or received.

This paper studies the performance implications of IP mobility solutions in an operator controlled networking environment as described earlier. The main focus and interest is on symmetric and asymmetric Voice over IP type traffic flows, and how their performance gets affected during handovers. The paper will also discuss the problem in a network architectural point of view. Performance measurements and signaling flows are captured from live operator networks. Real live network problems have strong influence in this paper. Transport protocol related problems are not analyzed too deeply, since those are already well studied [9]. For an operator it is usually more important to identify the biggest bottlenecks in the deployed infrastructure and optimize those before addressing transport protocol level issues. This paper will show that currently main bottlenecks in handover performance are located in the Mobile Node (MN) software, legacy access network deployments and AAA.

The rest of this paper is organized as follows: chapter II describes the testing environment and actual measurements, chapter III analyzes performed measurement, discusses about interesting findings during tests and also suggests improvements that could possibly improve handover performance. Finally we conclude this paper in chapter IV.

* Jouni Korhonen works for TeliaSonera as researcher in a R&D unit and can be reached at jouni.korhonen@teliasonera.com

II. SETUP OF TEST ENVIRONMENT AND MEASUREMENTS

This chapter describes the test environment and used measurements. Performed tests aim to measure the Mobile IPv4 handover time in Mobile Node point of view.

A. *Operator Environment - Mirroring Live Infrastructure*

One of the aims, when building the test environment, was to either completely use a live network infrastructure or make it as close to live network as possible. The final test environment used for the measurements is a mixture of a live network and laboratory environment. The Figure 1 illustrates the test network. Everything except the data traffic originating from the WLAN access network is using TeliaSonera's public live network.

The test network has a commercial Mobile IPv4 [3] implementation[†], which has been one of most up to date with standards and public ly available implementation. The cellular network provides both GPRS and UMTS accesses. The traffic form the GPRS/UMTS core is tunneled over a GRE tunnel to the APN-router. Tunneling has implication to IP MTU size due added tunneling headers. WLAN access points are latest Cisco Aironet 1200 models. RADIUS servers are running in SUN servers with operator grade commercial RADIUS server implementations. Access Controllers (AC) provide forced web-based WLAN login and access controlling function at IP level. DHCP-servers were based on the latest ISC[‡] DHCP-server dstribution and running in RedHat/Debian Linux servers. DHCP-server code was modified slightly and changes are described in greater detail in II.C.

Our network allowed also using EAP-based [5] authentication to the WLAN access network. When using EAP-based login the forced web-based login can be bypassed. Our WLAN access authentication and authorization was configured to use EAP-SIM [1][2], which made it possible to reuse our existing billing system and roam from WLAN access point to another without user involvement. As a side note, we completely ignore billing issues in this paper. Many WLAN hotspots are billed in time basis, which effectively defeats "*always on*" principle in network access. Due to current WLAN access network infrastructure deployment, introducing Foreign Agents (FA) to the WLAN access network would have been too complex to be justified in any way.

The Home Agent (HA) also included VPN-gateway software. Because of this deployment decision both Mobile IP and IPSec [10] tunnels get terminated inside the same physical server. The traffic just routed through different virtual interfaces inside the same server. Both Home Agent VPN-gateway and Correspondent Node servers were Linux servers running in a powerful enough PC-hardware.

B. *Mobile Nodes*

The Mobile Node used for the measurement was a powerful 1,8GHz Pentium based laptop running a fresh installation of Windows XP operating system. The laptop was equipped with an integrated 802.11 Wireless Protected Access (WPA) [1][7] capable WLAN card and a prototype UMTS terminal. The laptop also had a Gemplus SIM-card reader hooked to a USB port. The additional SIM-card was used for WLAN authentication as the laptop was unable to reuse the other SIM-card located in the UMTS terminal.

The software used during measurements included a commercial Mobile IPv4 client, a development version of a commercial EAP-SIM client and a commercial VPN client. The Mobile IPv4 and the VPN software were from the same vendor as the Home Agent. The EAP-SIM client retrieved its authentication data from the SIM-card on the USB reader. All required calculation was done in software in the client.

C. *Measurement Setup and Configuration*

The MN Mobile IPv4 client was forced to use UDP tunneling (NAT Traversal) [4]. In our test environment that would not have been required. In a general deployment scenario UDP tunneling is useful due the NAT capability and better cooperation with stateful FireWalls. The MN was also configured not to solicit FAs. Actually the MN software always prioritized registering to the HA before even trying the FA. HA'a address was configured to the MN and the MN home address was dynamic (based on the NAI extension). We also selected such MTU size that no IP packet fragmentation would occur.

The network traffic payload was generated using the *ttcp+* program, which is modified from the original *ttcp* to allow a client to trigger the server to start sending data. *ttcp+* was used for both TCP and UDP traffic generation. We used two types of traffic profiles:

- One constant bit rate downlink UDP flow. 56KBps for WLAN to WLAN handovers and 56kbps for UMTS to WLAN handovers.
- One constant bit rate downlink UDO flow and one downlink TCP flow. UDP flows were 56KBps for WLAN to WLAN handovers and 56kbps for UMTS to WLAN handovers.

Handovers were created by moving the MN between two WLAN Access Points (AP) and let MN 's WLAN driver to do the handover decision. The handover from UMTS to WLAN was arranged in a similar way. The WLAN interface was prioritized over the UMTS interface. So when the MN moved closer to an AP and WLAN's signal strength started to get better, eventually a vertical handover took place.

Measurement data was captured from three different locations in the test network. Capturing locations are shown in Figure 1. We

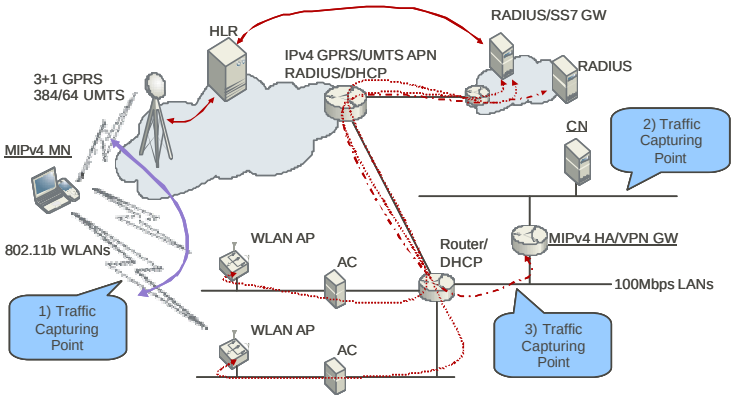

Figure 1: The test network architecture

[†] Secgo Mobile IP – http://www.secgo.fi
[‡] Internet Systems Consortium – http://www.isc.org

Table 1: WLAN to WLAN handover signaling times in seconds

	UDP+TCP			UDP		
	MAX	MIN	AVG	MAX	MIN	AVG
EAP	2,92	2,23	2,33	2,42	2,14	2,27
IDLE1	3,79	3,01	3,34	3,82	3,19	3,41
DHCP	1,85	0,8	1,53	6,37	1,14	2,08
IDLE2	3,07	2,28	2,71	3,18	2,34	2,75
MIP	5,99	0,22	0,73	1,22	0,11	0,35
Total	12,85	8,54	10,64	17,01	8,92	10,86

Table 2: UMTS to WLAN handover signaling times in seconds

	UDP+TCP			UDP		
	MAX	MIN	AVG	MAX	MIN	AVG
EAP	3,01	2,3	2,57	3,59	2,23	2,54
IDLE1	4,68	0,98	3,03	4,78	0,85	2,78
DHCP	0,87	0,24	0,57	0,74	0,08	0,43
IDLE2	3,26	3,26	2,97	3,35	2,74	3,02
MIP	0,21	0,09	0,16	0,39	0,11	0,19
Total	12,03	6,26	9,3	12,85	6,01	8,96

were mostly interested in analyzing the measured data from the locations 1) and 2). Especially our interest was to measure the handover time as seen by the MN in a signaling point of view.

During the measurements the VPN was allowed to complete its initial IPSec negotiation before starting tests and the MN was allowed to be attached to one access network. It should be noted that the no additional negotiation took place during the measurement.

We modified the ISC DHCP-server implementation so that is would work faster in situations when the MN did a handover from the WLAN to WLAN. In this particular case the MN tried to renew its existing lease using the old source IP-address in the DHCPREQUEST-message. Because the ISC DHCP-server implementation does not use MAC-addresses for its message routing but rather relies on IP-level routing, the DHCPACK got routed to old subnetwork or silently discarded. The DHCP-client kept retrying for 30 seconds until it timed out. After the timeout the DHCP-process started again with a DHCPDISCOVERY and the MN got a valid IP-address promptly from the DHCP-server. This poor DHCP performance could have also been avoided integrating the DHCP-client logic as a part of the Mobile IP software rather than relying completely on the Windows client.

III. ANALYSIS ON MEASUREMENTS

This chapter describes the measured results from both WLAN to WLAN and UMTS to WLAN handovers. This chapter also presents a summary of all measurements.

A. WLAN to WLAN Handover

The WLAN to WLAN handover signaling times break down as seen the by the MN is shown in Table 1. Measured times for a single UDP flow and combined UDP+TCP flows are in the same table. Although not shown in this table the WLAN association time was included into the EAP-negotiation time. Basic WLAN to WLAN handover and its affect to e.g. Voice over IP traffic is already well studied problem [6].

Average time spent in EAP-negotiation phase was 2.3 seconds. Deeper analysis showed that 2/3 of the total time was spent in the MN, either retrieving data from the SIM-card or calculating the key material. It turned out that the USB dongle and interface to the SIM-card was the bottleneck. The time spent in the DHCP phase is considerably long, around 2 seconds. The reason for this is the way the ISC DHCP-server and the DHCP-protocol works. As explained in the chapter II.C the Windows DHCP-client tried to renew its current release using the IP-address from the old subnetwork. This caused the DHCP-message exchange to be: DHCPREQUEST > DHCPNAK > DHCPDISCOVERY > DHCPOFFER > DHCPREQUEST > DHCPACK. The previous message exchange resulted in total to three roundtrips and the DHCP-server had to do also additional processing like releasing and creating a new lease. The Mobile IPv4 registration phase consists of one roundtrip and took average 0.5 seconds to complete.

The maximum handover time for the combined UDP+TCP traffic profile was 12,85 seconds and 17.01 seconds for bare UDP. The minimum for the combined UDP+TCP was 8,54 seconds and 8,92 seconds for the UDP alone. The average for the combined UDP+TCP traffic was 10,64 seconds and for the UDP alone 10,86 seconds. Studying the TCP-trace from the CN side showed that it took in average 15 seconds for TCP to recover from the temporary connectivity loss caused by the handover. The most notable delay factors were the *idle-time* periods (IDLE1 and IDLE2) between signaling phases. In average the *idle-time* periods took longer that the rest of the handover related signaling. We could not really trace down the cause for these *idle-time* periods but the strong indication was that they were Windows XP internals derivative.

B. UMTS to WLAN Handover

The UMTS to WLAN handover signaling times break down as seen by the MN is shown in Table 2. Measured times for a single UDP flow and combined UDP+TCP flows are in the same table. Although not shown in this table the basic WLAN association time was included into the EAP-negotiation time.

Average time spent in EAP-negotiation phase was 2.6 seconds. Deeper analysis showed that just like in the WLAN to WLAN handover case 2/3 of the total time was spent in the MN. The time spent in the DHCP phase is now considerably shorter, less than second. When the MN enters the WLAN network, the Windows DHCP-client does a normal DHCP address configuration procedure starting with a discovery of DHCP-servers. The Mobile IPv4 registration phase consisted of one roundtrip and took average 0.18 seconds to complete. One of the reasons why both the Mobile IPv4 and DHCP signaling were faster is due the WLAN link and the LAN (where the WLAN AP is connected to) not being congested. The background traffic consumed less bandwidth in the UMTS to WLAN case and also used a different route.

The maximum handover time for the combined UDP+TCP traffic profile was 12,85 seconds and for the UDP alone 10,85 seconds. The minimum for the combined UDP+TCP was 8,54 seconds and for the UDP alone 6,01 seconds. The average for the combined UDP+TCP was 10,64 seconds and for the UDP alone 8,96 seconds. Studying the TCP-trace from the CN size showed that it took in average 20 seconds for TCP to recover from the temporary

connectivity loss caused by the handover. The most notable delay factors were again those *idle-time* periods between signaling phases; just like in the WLAN to WLAN handover case.

C. Measurement Summary

Both handover test cases have similar results. The only big change happens in DHCP-signaling times. The UMTS to WLAN case is considerably faster than the WLAN to WLAN case because unnecessary DHCPREQUEST > DHCPNAK phase did not happen. An interesting finding was the role of the DHCP, which depending on the server or the client implementation (or configuration) could cause a very long additional delay.

Idle-time periods are an indication that a loosely integrated MN software may have unexpected and unexplained delay factors. In our cases we managed to trace down one delay factor to the EAP-negotiation phase, which was the SIM access interface. Even if all *idle-time* periods could be removed, still the general handover delay would be at its best in magnitude of several seconds – in the MN and signaling point of view. If transport level delays are also taken into account the total handover time that applications experience would be considerably longer. For example it took in average 15 seconds for TCP flows to recover from the WLAN to WLAN handover and 20 seconds from the UMTS to WLAN handover. For UDP flows the situation is not as bad as for TCP because there are no retransmissions or slow start. In any case measured handover delay times show that general purpose evolutionary deployment of Mobile IPv4 on top of an existing infrastructure defeats completely one of the use cases that is used to drive Mobile IPv4 – namely the Voice over IP. For services and application without any time constraints the used Mobile IPv4 infrastructure and deployment would work well.

D. Possible Handover Optimizations

There are number of optimization methods that could be applied to the Mobile IPv4 infrastructure and deployment described in this paper. Still having the evolutionary network upgrade approach in mind most of the optimizations should be applied to the MN. One possible network related optimization could be allowing WLAN APs to perform pre-authentication as discussed in [7].

On the MM side enhancing the SIM access could save at least a second. Tight integration of the MN software – at least in our Windows laptop case – could drop the handover time to half or even to one third of the current time. Unfortunately that would also mean more or less integrating the EAP-client, DHCP-client and the Mobile IPv4 into one component. Allowing the MN to use multiple interfaces simultaneously to exchange data would definitely help during handovers. Alas benefiting from multiple interfaces would require a network infrastructure to be able to prepare for handovers and allow multiple Mobile IPv4 tunnels/registrations [12]. Also actively monitoring layer 2 would allow triggering different signaling phases as soon as possible. And in general knowing the header overhead of a NAT capable Mobile IPv4 with a NAT capable VPN is something that should also be addressed.

IV. SUMMARY

This paper presented results of Mobile IPv4 handover performance measurements that were performed in a live operator network. Due the "reality check" aspect the test set up had restrictions that affected the handover performance. Both WLAN to WLAN and UMTS to WLAN handovers were measured – mostly in a handover signaling point of view. The Mobile Node was equipped with a Mobile IPv4 client, a VPN client and an EAP-client that used the EAP-SIM authentication method. Both the VPN and the SIM-based authentication provide adequate level of security for a nomadic user.

One of the desired applications for a corporate WLAN is the Voice over IP. Alone the WLAN to WLAN handovers showed that any service having real-time requirements, such as the Voice over IP – would not tolerate the handover delays measured for this paper. The fastest handover time for the WLAN to WLAN was over 8 seconds, the slowest was about 17 seconds and the average was slightly less than 11 seconds. The UMTS to WLAN handover delay times were at best about 6 seconds, the slowest was almost 13 seconds and the average was around 9 seconds.

There is clearly a need for optimizations. Unfortunately the evolutionary way of introducing new functionality into operator networks will not allow making any radical changes. It is not always even possible to affect the used access network infrastructure. This is the unfortunate case for example in interoperator roaming cases. We also noticed that the majority of the handover delay times were caused by *idle-time* periods between handover signaling phases and accessing the SIM in the Mobile Node. Thus the most obvious place to start optimizing handovers would be addressing the Mobile Node software, trying to integrate all required components tightly together instead of using separate client services.

REFERENCES

[1] "Intergration of 3G Wireless and Wireless LANs", Special issue, *IEEE Communication Magazine*, Vol. 41 No. 11, November 2003.

[2] H. Haverinen, J. Salowey, "EAP SIM Authentocation", IETF draft-haverinen-pppext-eap-sim-13.txt, Work in Progress.

[3] C. Perkings, "IP Mobility Support for IPv4", IETF RFC3344, August 2002.

[4] H. Levkowetz, S. Vaarala, "Mobile IP Traversal of Network Address Translation (NAT) Devices", IETF RFC 3519, April 2003.

[5] L. Blunk, et'al, " Extensible Authentication Protocol (EAP)", IETF draft-ietf-eap-rfc2284bis-09.txt, Work in Progress.

[6] Jon-Olov Vatn, "An experimental study of IEEE 802.11b handover performance and its effect on voice traffic", KTH, Royal Institute of Technology, Kista, SWEDEN, July 2003.

[7] B. Aboba, "IEEE 802.1X Pre-Authentication", Available at http://www.drizzle.com/~aboba/IEEE/11-02-TBDr0-I-Pre-Authentication.doc, June 17, 2002.

[8] 3GPP, "3GPP system to Wireless Local Area Network (WLAN) interworking; System description (Release 6)", TS23.234 V6.0.0 (2004-03).

[9] A. Gurtov, J. Korhonen, Effect of Vertical Handovers on Performance of TCP-Friendly Rate Control, submitted for publication, 2004.

[10] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol", IETF RFC 2401, November 1998.

[11] F. Adrangi, H. Levkowetz, "Problem Statement: Mobile IPv4 Traversal of VPN Gateways", IETF draft-ietf-mip4-vpn-problem-statement-02.txt, Work in Progress.

[12] K. El Malki, "Low Latency Handoffs in Mobile IPv4", IETF draft-ietf-mobileip-lowlatency-handoffs-v4-08.txt, Work in progress.