# Handover Latency Measurement of Mobile IPv6 in a Testbed Environment

Muhammad Arif Amin, Kamalrulnizam Bin Abu Bakar,
Abdul Hanan Abdullah, Mahesh Nair, and Rashid Hafeez Khokhar

Universiti Teknologi Malaysia, Johor, 81310 Skudai, Malaysia
`mamin@hct.ac.ae`
http://www.hct.ac.ae

**Abstract.** The emergence of wireless networking necessitates continuous time connectivity to support end-to-end TCP or UDP sessions. Wireless networking does not provide reliable connections to mobile users for real-time traffic such as voice over IP, audio streaming and video streaming. Handover latency in Mobile IPv6 poses many challenges to the research world in terms of disconnecting users while roaming. Many efforts have been made to reduce the handover latency with focus either on layer 2 or layer 3. This paper presents the handover procedure of Mobile IPv6 and investigates various factors affecting the delay during network switch over. In this paper, a testbed environment is presented that includes two different wireless LAN networks using Universal Mobile IP for Linux (UMIP) implementation and Cisco routers. The aim is to present handover latency caused by multiple signals at layer 2 and layer 3 and make recommendations on how to reduce the total handover latency experienced by the MIPv6 protocol.

**Keywords:** Handover, IPv6, WLAN, Mobile IPv4, Mobile IPv6.

## 1 Introduction

The exponential growth of wireless devices in the last few years demand continuous connectivity to the Internet. Mobile users require constant communication with others while moving from one place to another. To be able to reach others during motion, a node must have a unique public Internet Protocol (IP) address so that the traffic can be sent to others [1]. IP version 4 (IPv4) was originally proposed in the late '70s and was capable of allocating 32-bit addresses, which in total could provide approximately 4.3 billion addresses in the world. Since private IPv4 addresses does not follow the Internet hierarchy for address allocation, therefore it does not allow all the nodes to communicate on the Internet. To provide mobility to fixed nodes with IPv4, mobile IPv4 protocol [2] was introduced. However, due to the address space restriction, the protocol failed to gain popularity in the Internet world. The address space restriction in IPv4 led the Internet Engineering Task Force (IETF) to begin work on the new protocol called IPv6 [3], which is capable of providing 128-bit addresses to all IPv6-capable devices.

N. Meghanathan et al. (Eds.): CCSIT 2012, Part I, LNICST 84, pp. 285–299, 2012.
© Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2012

IPv6 follows the Internet hierarchical structure to assign IPv6 addresses to the nodes that allows any node to communicate with another node over the Internet. However, based on the experience with IPv4, the mobility functionality in IPv6 was kept from the beginning. Therefore, an extension to the IPv6 header was introduced to provide mobility to wireless users and thus named Mobile IPv6 (MIPv6) [4]. The potential use of the MIPv6 protocol is in smart phones, wireless laptops and many other devices. Typically, when the mobile node (MN) changes its location from one network to another, the IP address of the MN must change accordingly, this process is called the handover. However, with this process, the MN loses connectivity from the previous network, and the application in use disconnects. Although the TCP application can tolerate disturbance in connection and request retransmission, the UDP protocol cannot. Most of the real-time applications rely on UDP for fast communication. In this paper, we present experimental results in order to characterize and quantify the handover latencies of Mobile IPv6. Multiple experiments are set up to measure layer 2 and layer 3 handover delays independently in order to measure disruption due to the movement between different networks. In order to evaluate and quantify the results, an open source implementation environment is used, in particular Debian 5.0.1 and an open source MIPv6 application for Linux called Universal Mobile IP "UMIP 0.4" [5]. A deep analysis of the results are presented by comparing multiple methods that a mobile node may use to switch between networks.

Many articles propose new techniques and research to reduce the handover latency in MIPv6 by modifying the protocol, or propose methods to enhance either layer 2 or layer 3. This paper focuses the standard MIPv6 implementation on a Linux platform. Previous work has been done by many people to measure the performance and delays in MIPv6 using a testbed environment. In [6,7], the author used MIPL on RedHat 8.0, but the author did not implement route optimization. Another implementation done by [8], which includes Fedora Core and MIPL installation, movement detection, duplicate address detection (DAD)and effect of router advertisement (RA), is studied. However, the author used Linux machines as access routers and a Cisco router only for routing between networks. In [9], MIPv6 testbed is set up using FreeBSD and KAME to demonstrate the handover latency and the effect of RA is studied to reduce the handover latency. Using another Linux-based testbed for Linux [10,11], in which the authors have measured the handover latency, TCP and UDP protocols performance. A recent study includes performance of transport protocols in a testbed [12]. The rest of the paper is organized as follows: Section 2 presents the detailed Mobile IPv6 handover process and its components. In section 3 layer 2 experimentation and results are presented. Layer 3 experiments and results are presented in section 4 and the conclusion is presented in section 5.

## 2 Mobile IPv6 Handover Process

The Mobile IPv6 handover process provides the mechanism for users to roam between different networks; however, in order to roam freely a user should not disconnect from the current network. To achieve this, network entities should communicate with each other and be able to transfer information while the mobile node (MN) is moving from one network to another. A MN is allocated an IPv6 address from the home network called the home address (HoA). The MN is always addressable using this address by the communicating nodes called the correspondent node (CN); however, when it moves to a foreign network, the CN must form another IPv6 address called a care of address (CoA). Packets can still be routed to the MN by using a mechanism in which network entities such as access routers communicate with each other and forward traffic [13]. To perform the handover process, the access router must be configured with additional feature which allow packets to move continuously to the foreign network keeping the MN in contact.

- A MN must be able to detect the change in the network; i.e., the MN has moved to a new network.
- The MN must be able to inform its home network and other nodes communicating with it.
- The handover process should be performed efficiently so that upper layers do not disconnect.

A complete MIPv6 handover process consists of the layer 2 and layer 3 handover process [13]. However a layer 3 process cannot start unless the layer 2 process is completed by the MN. The Layer 2 process includes scanning, authentication and association to wireless access point. The Layer 3 process includes discovering new routers, address configuration, movement detection and then IP registration as shown in Figure 1.

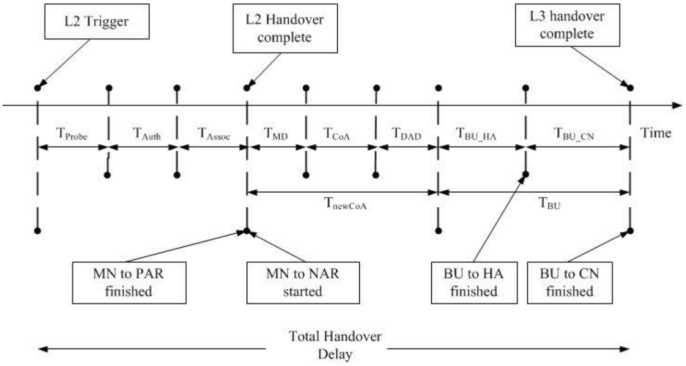

**Fig. 1.** Handover Delay Time Line for Mobile IPv6

The MIPv6 handover process mainly consists of the following components:

- Movement detection time ($T_{mvd}$): This is the time the MN uses to detect IPv6 router advertisements and neighbor discovery to find out if the MN has moved to a new network.

- Duplicate address detection time ($T_{DAD}$): This is the time take by the MN to form a CoA and perform duplicate address detection to confirm the uniqueness of the IPv6 address.
- Binding Update time ($T_{BU}$): This is the time taken by the MN to get acknowledgment from CN and HA after sending binding update (BU) signals.

The mathematical representation of the L2 and L3 handover is shown below.

$$T_{hand} = T_{L2} + T_{L3} \tag{1}$$

$$T_{L2} = T_{scan} + T_{auth} + T_{assoc} \tag{2}$$

$$T_{L3} = T_{mvd} + T_{DAD} + T_{BU} \tag{3}$$

$$Total_{hand} = T_{scan} + T_{auth} + T_{assoc} + T_{mvd} + T_{DAD} + T_{BU} \tag{4}$$

## 3 Layer 2 Handover Experiments

In the initial experimental setup, it is an obvious choice to link the IEEE 802.11-based mobile node in an environment that is independent of Mobile IPv6 protocol to test the link layer handover. The IEEE 802.11b wireless network specification has obtained significant acceptance in the industry and research; therefore, the specification is chosen as a layer 2 protocol over which to run Mobile IPv6. The aim of this first experiment is to evaluate the 802.11b handover time independent of Mobile IPv6, as a basis for later evaluating the component of handover time that occurs at the IP layer. In this experiment, the layer 2 and layer 3 handover delays are added to determine the total handover latency.

### 3.1 Methodology

Link layer or layer 2 handover in an IEEE 802.11b wireless network occurs when an 802.11b-based mobile node changes its point of connection from one network to another, usually characterized by a move from one access point to another. In order to experimentally quantify the average handover time of an 802.11b network, a simple IP network is built, consisting of two independent 802.11b Local Area Networks (LANs) linked by a regular Ethernet backbone. A single 802.11b mobile node is forced to move back and forth between the two access points, creating the kind of link layer move that would trigger a Mobile IPv6 handover event.

There are three methods for triggering a switch between access points:

1. Method 1: By decreasing the transmission power of the access point: In this method, the transmission power of the access point is decreased to which MN is currently connected. The MN will detect the degraded signal strength and switch to another access point, provided they both have the same SSID.

2. Method 2: Configure both access points with the same SSIDs and the same channel: In this method, both access points are configured with 100% transmission power. The MN will switch to another AP when not available. However, this is done by manually shutting down the wireless card of the access point to which the MN is currently attached.
3. Method 3: Configure both access points with different SSIDs and different channels: In this method, the MN will go through all the channels available before attaching to the other AP. This will consume more time and generate delay and more packet loss. To perform the test, the wireless card is manually shut down in the access point to which the MN is currently attached, forcing it to switch to another access point. This will require the MN to be configured with both SSIDs manually. However, a MN can also perform probes to detect another SSID in range and attach if open system authentication is used.

In this experiment, only methods 2 and 3 are used to obtain results and compare for consistency since it is not easy to switch the MN to another access point based on the power level in a lab environment. It is observed that when high-power access point is switched on, the MN immediately attaches to the access point regardless of forcing it to do so. However this experiment can be tested in a large area with no interference and the presence of other access points in the environment. In order to measure the handover latency at the link layer, a packet sniffing tool such as Wireshark [14] is used to record the time when MN disassociates from the old access point and associates with the new one.

### 3.2 TestBed Setup

A simple Ethernet network is established as shown in Figure 2. Two access points are connected to an Ethernet switch and are located close to each other. One node with the Debian 5.0.1 operating system is installed and configured with a wireless network card to act as a mobile node switching between access points. Another node with a wireless network card configured in monitor mode and installed with a packet sniffing tool called Wireshark is used to capture packets from the mobile node. The captured packets are then analyzed, and the individual delay related to the probe, authentication and association process is measured. The MN is initially attached with an old access point and then a switches over to the new access point. This is done by either reducing the transmitting power of an access point or by shutting down the old access point.The equipment used in the testbed setup is shown in Table 1.

**Method 2.** Triggering Handover by Configuring Both Access Points with the same SSIDs and same Channel:

In this method, both access points are configured with channel 6 and the SSID “Home”, the transmission power of both access points is kept to 100%. The configuration of the testbed is shown in Table 2. Initially, the mobile node is attached to access point 1 and SSID ’Home’, and then the handover is triggered by shutting down the wireless card of access point 1. Since the network is set up

**Table 1.** Layer 2 Experiment Equipment Details

| Hardware | Operating System | Interface | RAM | CPU |
|---|---|---|---|---|
| Cisco 2960 Switch | Cisco IOS 12.2 | 24 10/100 Ethernet | 64 MB | IBM PowerPC405 |
| 2 X Dell Laptop D505 | Debian 5.01 | 1 10/100 Ethernet, 1 Cisco WiFi 802.11b | 2 GB | 1.5 GHz |
| 2 X Cisco Access Points 1200 | Aironet 12.2(13)JA3 | 1 Ethernet, 802.11 b/g | 16 MB | IBM PowerPC405 200 MHz |

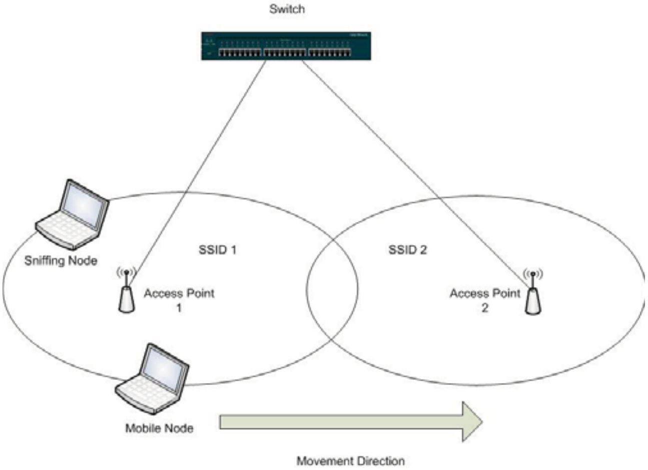

**Fig. 2.** Layer 2 TestBed Setup

with similar SSID and the same channle, the mobile node will switch to access point 2 in the area immediately. The handover delay is measured by sniffing the mobile node using wireshark program running on the sniffing node.

**Method 3.** Triggering Handover by Configuring Both Access Points with Different SSIDs and Different Channels:

In method 3, both access points are configured with the same power level, but with a different channel number and SSID. Access point 1 is set to channel 1 with SSID “Home” and access point 2 is set to channel 6 with SSID “Foreign”. The configuration of the testbed is shown in Table 3. Initially the mobile node is attached with access point 1 and SSID “Home” and then the handover is triggered by shutting down access point 1. Since the network is setup with a different SSID, the mobile node will perform the probe scan, authentication and

**Table 2.** Testbed Configuration: Method 2

| Access Point | Power Level | SSID | Power | Channel |
|---|---|---|---|---|
| Access Point 1 | 100 mW | Home | 100% | 6 |
| Access Point 2 | 100 mW | Home | 100% | 6 |

**Table 3.** Testbed Configuration: Method 3

| Access Point | Power Level | SSID | Power | Channel |
|---|---|---|---|---|
| Access Point 1 | 100 mW | Home | 100% | 1 |
| Access Point 2 | 100 mW | Foreign | 100% | 6 |

association process to switch to access point 2 in the area. The handover delay is measured by sniffing the mobile node by using wireshark program running on the sniffing node.

### 3.3 Results

**Method 2.** Triggering Handover by Configuring Both Access Points with the Same SSID and the Same channel:

Figure 3 shows a scan, authentication and association delay as measured with experiments. A total of 15 different tests were conducted to analyze the signal pattern and delays. The graph shows that the highest delay is caused by the scan process and the lowest by the association process. The maximum value for scan delay is 0.069 sec, and the minimum is 0.0003 sec and the mean scan delay is approximately 0.01678 sec. The sudden increase in the scan delay could also be due to the interference from the other access points. However, since both access points were kept on the same channel, the MN does not scan when moving to the other access point because the MN will initially look for an access point on the same channel. The authentication delay minimum is 0.0004 sec, the maximum is 0.0357 sec and the mean 0.0033 sec. There is only one sudden authentication increase during the experiment in test 4 when the access point delayed the response to the MN. The association delay minimum is 0.0002 sec, the maximum 0.0089 and the mean is 0.0026 sec. The association delay does not take much time when the MN is already authenticated. The association step makes the MN and the access point start exchanging data packets.

Based on the mathematical equation (2) in section 2, the total layer 2 handover delay is the sum of scan, authentication and association delays. The layer 2 handover delays are measured as minimum of 0.0027 sec, maximum 0.0722 sec and the mean is 0.022 sec. The layer 2 handover largely depends on the individual signals. If any signal causes more delay, it will create impact on the total handover. In some cases where multiple authentications are used to secure the wireless networks, the layer 2 handover delay increases drastically.

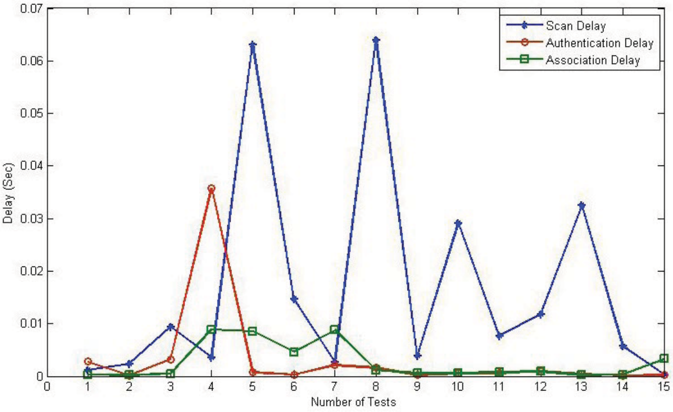

**Fig. 3.** L2 Signal Delays, Same SSID

The experimental result is shown in Figure 4 in which the minimum handover delay is 0.606 sec, the maximum is 0.687 sec and the mean is 0.630 sec. The large delay in the layer 2 handover depends on the scan delay, but if the scan delay is reduced, the total L2 handover can also be decreased. The Layer 2 handover constitutes part in the total handover delay in MIPv6; thus, if the layer 2 handover increases, then the total handover increases as well.

**Method 3.** Triggering Handover by Configuring Both Access Points with Different SSIDs and Different Channels:

Method 3 is similar to method 2, but with access points on different channels (access point 1 on channel 1 and access point 2 on channel 6). Figure 5 shows a plot of individual layer 2 signals. There is no difference in the authentication and association delay; however, the scan delay has increased compared to method 1.

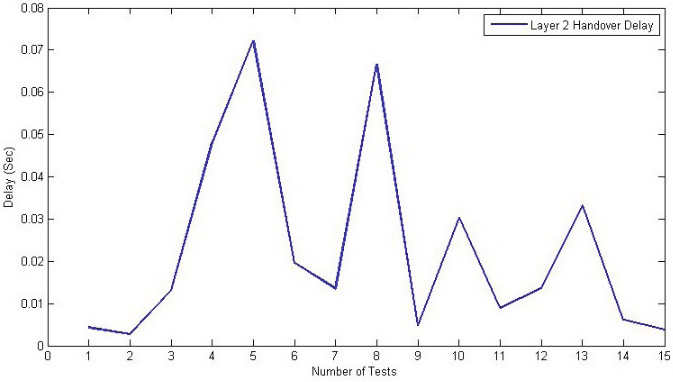

**Fig. 4.** L2 Handover Delay, Same SSID

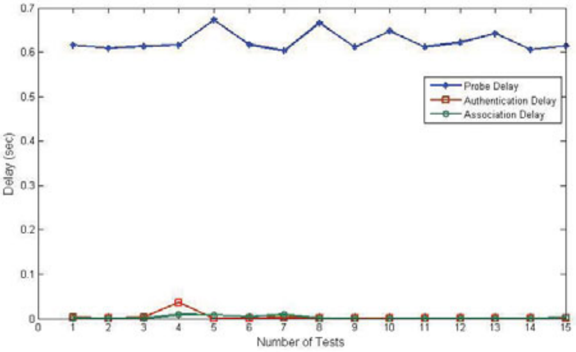

**Fig. 5.** L2 Signal Delays, Different SSID

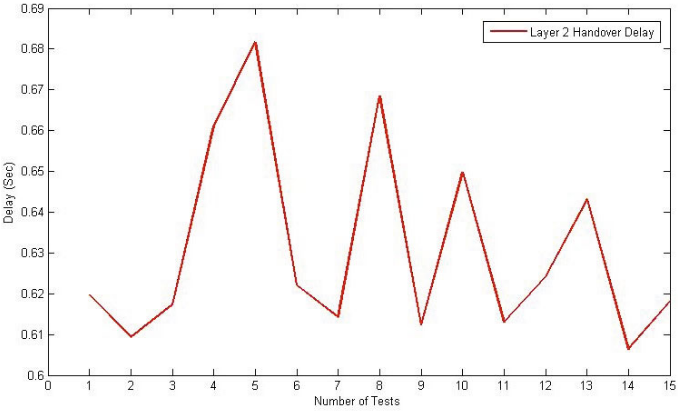

**Fig. 6.** L2 Handover Delay, Different SSID

The reason for the increase in the scan delay is because of scanning channels starting from channel 1 to channel 6. This has generated additional delay for the MN to send the authentication request signal. The increase in scan delay was measured from the time the MN has sent the de-authentication signal to the access point 1 and received a probe response signal from the access point 2. The minimum time for scanning is measured as 0.6003 sec, maximum at 0.6194 and the mean delay as 0.6079 sec. Since the scan delay has increased, it has impacted on the total layer 2 handover delay; thus, the minimum delay is 0.606 sec, maximum 0.6817 and the mean is 0.630 sec as shown in Figure 6.

## 4 Handover Delay on MIPv6

This section presents multiple experiments conducted to measure handover delays in standard Mobile IPv6 network using routers, access points, a mobile node and a correspondent node. The total handover delay in MIPv6 is the sum of the layer 2 and layer 3 delays, so the results obtained from the previous section allow an estimate of the contribution of the layer 2 handover time to the total handover time when using Mobile IPv6 over the 802.11b network.

### 4.1 Methodology

In order to experimentally measure the average handover delay of a Mobile IPv6 network, a physical network including IPv6 and Mobile IPv6 is built in a lab environment. Two different IPv6 networks are built using access routers configured to support IPv6 and MIPv6 functionality in order to send and receive signals from the mobile node. The access router at the home network is configured with home agent functionality, and the access router at the foreign network is configured with standard IPv6 functionality. The mobile node and correspondent nodes are installed with the Debian 5.0.1 operating system. The MN is running UMIP 0.4 application to support Mobile IPv6 functionality and is used to switch between the two different networks to measure the handover delay. Each network contains one access point configured with different SSID and auto channel selection to avoid interference. To trigger the handover, the MN is moved from ”Home” network to ”Foreign” and then back to ”Home”. In order to observe the effect of switching between the two networks and to measure the handover time, an Internet packet generator software called Distributed Internet Traffic Generator (D-ITG) is installed on the MN and correspondent node [15]. The software is used to simulate the different type of traffic; D-ITG support IPv4 and IPv6 protocols for simulation purpose. A large size UDP traffic is generated to measure various parameters such as MIPv6 delay, packet delivery rate and packet loss. The Mobile IPv6 handover time is determined by measuring the delay time when the MN loses communication and restarts communication with the CN. A sniffing computer at the foreign network is used to continuously sniff the MN traffic so that the tcpdump files can be used to investigate the details of the Mobile IPv6 handover procedure used by the UMIP implementation. A total of 15 tests were carried out to collect the results for analysis purpose and delay accuracy.

### 4.2 TestBed Setup

This section explains the testbed setup and relevant hardware in a lab environment for experimentation. It is important to discover the tools and equipment needed with compatibility with mobile IPv6 features. It has been discovered that Cisco 2600 routers with IOS version ”c2600-advipservicesk9-mz.123-11.T3” and above support IETF RFC 3775 for Mobile IPv6 protocol. These routers support Mobile IPv6 features such as Binding Update, Binding Cache, Neighbor discovery, Duplicate address detection and Binding acknowledgment. The routers are configured with two different IPv6 networks to support the IPv6 routing feature. Since all the operating systems do not support Mobile IPv6 protocol, a careful study has shown that an open source mobility application called UMIP

0.4 can be installed on a Linux Kernel to provide mobility functionality through Ethernet or wireless network interface cards. Therefore, both MN and CN nodes are installed with the Debian 5.0.1 operating system with Linux Kernel version 2.6.30.1 and a UMIP 0.4 application to enable Mobile IPv6 functionality. The equipment list is shown in Table 5, network configuration in table 6 and the network topology in Figure 7.

**Table 4.** Layer 3 Experiment Equipment Details

| Hardware | Operating System | Mobility Software | Interface |
|---|---|---|---|
| 3 X Cisco 2611XM | c2600 IOS 12.4 | IOS | 2 10/100 Ethernet |
| Dell Laptop D505 | Debian 5.0.1, Kernel 2.6.30.1 | UMIP 0.4 | 1 10/100 Ethernet, 1 Cisco WiFi 802.11b |
| Dell Laptop D505 | Debian 5.0.1, Kernel 2.6.30.1 | UMIP 0.4 | 1 10/100 Ethernet, 1 Cisco WiFi 802.11b |
| 2 X Cisco Access Points 1200 | Aironet 12.2(13) JA3 | None | 1 Ethernet, 802.11 b/g |

Figure 7 shows the network topology, in which access point 1 and router 1 are part of home network for the MN and access point 2 and Router 2 are part of foreign network. Router 1 acts as a home agent since it is configure as home agent. However, router 3 is connected with both router 1 and 2 with serial links to simulate WAN and is configured to do routing between all the networks. Hence, the routing table contains all the routes that exist in the network. This is required when the MN moves to a new network, it has to send the binding update signals to the home agent. A correspondent node is attached to router 3 for communication purpose, the MN will send all the traffic to CN while moving between the home and foreign network. The mobile node's IPv6 address at the home network is aaaa:0:1:0:aaa:ff:fe00:8 and the home agent's IPv6 address is aaaa:0:1:0:aaa:ff:fe00:2. Access point 1 is configured to broadcast SSID "Home" and access point 2 is configured to broadcast SSID "Foreign"; are configured to accept open authentication, so that no extra delays is encountered during the motion. The mobile node starts its movement from the home network and moves towards the foreign network and returns home.

**Experiment 1.** Moving the Mobile node from Home Network toward Foreign Network.

In this experiment, the mobile node is initially connected to the home network, and the packet generator is configured with a constant stream of UDP traffic of 1000 packets per second and 512 bytes of packet size. The correspondent node

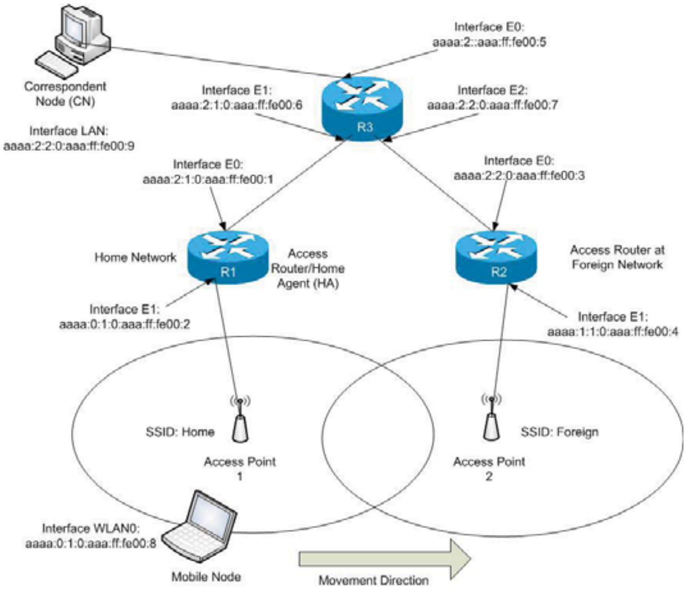

**Fig. 7.** Layer 3 TestBed Setup

is configured with the same packet generator in the receiver mode. The G.711 Voice specification traffic with 100 packet/sec and 120 bytes/packet is configured on the MN using the DITG generator to send VoIP traffic to the CN. When the MN is moved from the home to foreign network, it initially performs many tasks such as layer 2 handover, movement detection, IPv6 address configuration and then duplicate address detection. Once all these steps are done, then it starts to send binding, Home Test and Correspondent Test signals to the home agent and correspondent node, respectively. This is to inform the HA and the CN about the new IPv6 address so that they can continue communication. However, this process causes delays since the packets sent to the CN are lost until the MN gains re-connectivity. During the trial, the router advertisement interval is set to the standard 30-70 ms.

**Experiment 2.** Moving the Mobile node from Foreign Network toward the Home Network.

In experiment 2, the method used as exactly as experiment 1 except that the MN was moved from foreign network towards the home network while executing the same VoIP application. However, this process will have less latency, since the MN is returning home and has most of the information configured on it such as the home IPv6 address and the home agent address. Figure 8 shows the total MIPv6 delay for both experiments 1 and 2; it is observed that a maximum of 7 sec and minimum of 4.8 sec delay with the average of 5.6 sec has occurred when moving the MN from the home to the foreign network compared to maximum of 2.16 sec and minimum of 0.932 sec and average of 1.43 sec delay from the foreign to the home network. This is because when the MN moves from the home to the foreign network it performs Layer 2 handover which constitutes 0.6 sec, and then movement detection (MvD), IP address configuration (IPad), duplicate address detection (DAD) and binding update (BU). All these processes constitute individual delays, in which the DAD is the highest. However, it is also difficult for a MN to detect it has changed the network; this can be done only by listening to the router advertisements (RA) from the new router and sensing that it has moved. This becomes difficult for the MN to decide, because it can still receive RA from the old router as well from the new router.

### 4.3 Results

The analysis of trial 1 shows that the MIPv6 has higher delays and greater packet loss compared to trial 2.

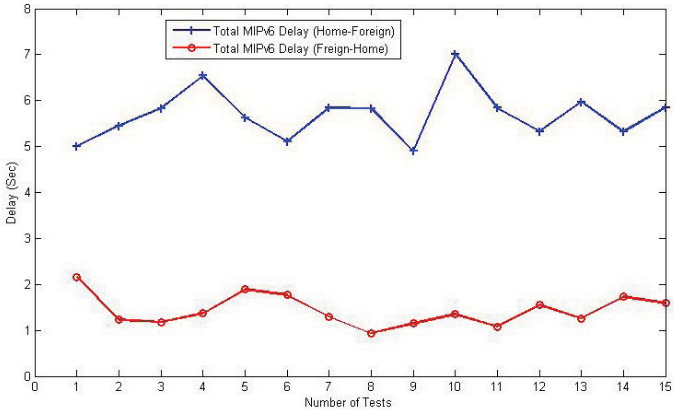

**Fig. 8.** Total MIPv6 Delay

The packet delivery rate is shown in Figure 9 and packet loss in Figure 10. It is observed that the packet delivery is much lower when the MN moves from home to foreign network. This is due to the large delay and large packet loss. Both the home router and MN are not configured to buffer packets; as soon the MN changes its location, the Home router stops sending packets to the CN and the MN. The maximum packet delivery rate from the home to the foreign is 13.4 packet/sec and minimum 3.2 packets/sec with average of 8.4 packets/sec. However, from the Foreign to the home network, the maximum is 62 packets/sec and the minimum is 31 packets/sec with an average of 43 packets/sec. It has been observed that at certain events when the MIPv6 delay is higher than the usual, even after receiving the minimum of three RA signals by the MN, it does not configure the IPv6 address. However, it starts sending the neighbor solicitation and router solicitation signals to the routers. This produces additional delay on the MN, and occurred only once or twice during the test runs.

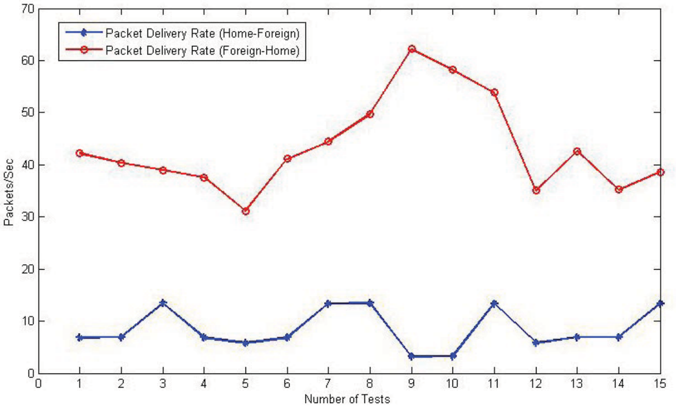

**Fig. 9.** Packet Delivery Rate

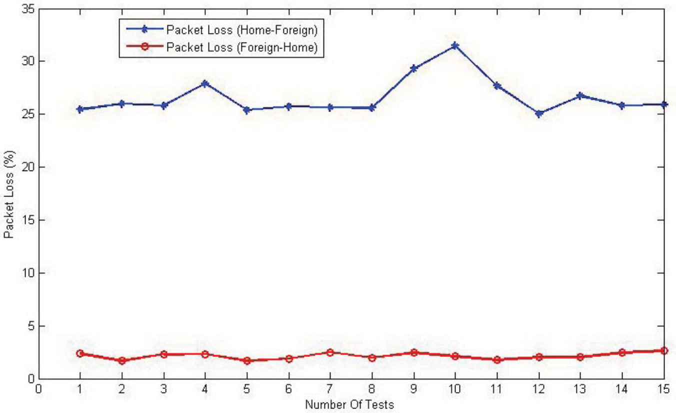

**Fig. 10.** Packet Loss

## 5 Conclusion

The handover process in Mobile IPv6 is a composition of layer 2 and layer 3 delays. Theoretically, the layer 2 delays consist of scanning, authentication and association. The layer 3 handover process consists of movement detection, IPv6 address configuration, duplicate address detection and binding update process. A layer 3 handover cannot begin until the layer 2 handover process finishes. Both layer 2 and layer 3 constitute delays that add up and produce a large handover delay for the MIPv6 protocol. In this paper, layer 2 and layer 3 delays have been analyzed individually through experimentation in a lab environment. A testbed is set up with routers, access points, a mobile node and a correspondent node. The mobile node is moved from one network to another while executing VoIP traffic and the delays are measured at the MN and the CN.

The test results show that the MIPv6 protocol experiences large handover delays, when the MN moves from the home network towards the foreign network compared to the reverse direction, the percentage difference of 42% is observed. The same is true for the packet delivery and packet loss. The packet delivery ratio between both movements is 1 to 5, where 5 is from the foreign to the home network and 1 is from the home to the foreign network. This is mainly because of large packet loss during movement. It has been observed that a large number of RS signals are sent by the MN rather than listening to the RA signals and detect a new network. In the future more realistic experiments will be performed to measure handover latency, in particular changing parameters such as the router advertisements and router solicitation. A number of applications such video, audio streaming and different standard of VoIP will be used to measure the packet loss and throughput.

## References

1. Postel, J.: Internet Protocol, RFC 791 (Standard), Updated by RFC 1349 (September 1981)
2. Perkins, C.: IP Mobility Support for IPv4, RFC 3344 (Proposed Standard), Updated by RFC 4721 (August 2002)
3. Deering, S., Hinden, R.: Internet Protocol, Version 6 (IPv6) Specification, RFC 2460 (Draft Standard), Updated by RFCs 5095, 5722, 5871 (December 1998)
4. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6, RFC 3775 (Proposed Standard) (June 2004)
5. Umip: Usagi-patched mobile ipv6 for linux, `http://umip.linux-ipv6.org/index.php?n=main.homepage`
6. Werapun, W., Unakul, A.: An experimental mobile ipv6 for linux testbed system, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, NCS 2003 (September 2003)
7. Antoine, S., Wei, M., Aghvami, A.H.: Mobicom poster: impact of mobile ipv6 handover on the performance of tcp: an experimental testbed. SIGMOBILE Mob. Comput. Commun. Rev. 7, 31–33 (2003)
8. Busaranun, A., Pongpaibool, P., Supanakoon, P.: Handover performance of mobile ipv6 on linux testbed, Specialized Systems Consultants, Inc. (2004)
9. Stewart, L., Armitage, G., Banh, M.: Implementing an ipv6 and mobile ipv6 testbed using freebsd 4.9 and kame, Centre for Advanced Internet Architectures, Technical Report 040331A (March 2004)
10. Chen, J.-L., Lee, Y.-F., Chang, Y.-C.: Mobile ipv6 network: implementation and application. Int. J. Netw. Manag. 16, 29–43 (2006)
11. Xie, G., Chen, J., Zheng, H., Yang, J., Zhang, Y.: Handover latency of mipv6 implementation in linux. In: Global Telecommunications Conference, IEEE GLOBECOM 2007, pp. 1780–1785 (November 2007)
12. Dhraief, A., Chedly, Z., Belghith, A.: The impact of mobile ipv6 on transport protocols an experimental investigation. In: 2010 International Conference on Communication in Wireless Environments and Ubiquitous Systems: New Challenges (ICWUS), pp. 1–8 (October 2010)
13. Koodli, R., Perkins, C.: Mobile Internetworking with IPv6. John Willey & Sons (2007)
14. Wireshark, `http://www.wireshark.org`
15. D-itg: Distributed internet traffic generator, `http://www.grid.unina.it/software/itg`