Inter-Domain Mobility with LISP-MN – A Performance Comparison with MIPv6

Musab Isah*, Chris Edwards**
School of Computing and Communication,
Lancaster University,
(*m.isah, **c.edwards)@lancaster.ac.uk

***Abstract*—In this work, we aim to evaluate Locator Identifier Separation Protocol-Mobile Node's (LISP-MN) performance in an inter-domain mobility scenario for both multi-interface and single interface Mobile Node (MN) with focus on throughput, handover delay, service disruption time and packet loss. To serve as the benchmark for performance, we compare LISP-MN with the IETF standardised MIPv6. We implement the 2 protocols on a laboratory testbed comprising all the nodes necessary for their operation. For multi-interface MNs, LISP-MN shows a better response in soft handover scenarios in terms of throughput and packet loss. MIPv6 on the other hand shows shorter handover delay with lower service disruption time in a hard handover scenario. Both protocols demonstrate poor performance for a single interface MN due to the long handover delay experienced. Although LISP-MN's handover control messages doubled that of MIPv6, our experiments show that it takes a similar time as MIPv6 to complete the handover message exchange.**

Keywords - inter-domain mobility, vertical/horizontal handover, lisp-mn, mipv6, heterogenous/homogenous mobility, loc/id split.

I. INTRODUCTION

Mobile devices are increasingly becoming the primary source of access to the Internet as the number of connected devices exceeded the world population in 2014 [1]. One feature that is likely to be prevalent in future mobile networks is the heterogeneity of the wireless network technology and the consequent frequent vertical handovers – a change in connectivity source between the different radio-access technologies, which was made possible by the multi-interface capability of today's mobile devices. Users will also have the ability to move between several available independent Wi-Fi networks in places such as train stations, airports, and shopping malls. For example, Gao et al. [2] have shown that 20% of mobile nodes have at least ten IP address changes per day, which suggests roaming between networks under different domain and/or administrative control. As we see today, there will be a large convergence of many wireless access technologies, e.g. cellular, wireless broadband access networks, wireless sensor networks, and Wireless Local Area Network. Users will have several wireless networks available to which their devices connect to and disconnect from automatically depending on the devices' network needs, configuration and subscription. The most common scenario today is a change of Internet connection between 3/4G cellular and home/office/public Wi-Fi.

This research is funded by Petroleum Technology Development Fund (PTDF), Nigeria. Reference PTDF/E/OSS/PHD/IMM/11.

The current Internet routing and addressing architecture was not designed to achieve any such level of mobility. This is because the IP address is used to define both the location and identity of a network device interface. The need to decouple this semantic was known even before the Internet was created [3]. Furthermore, this decoupling is identified as an important component towards finding the solution to the problems of scalability, multihoming, and inter-domain traffic engineering faced by the Internet today [4]. Although IPv6 provides enough addresses to identify the billions of devices on the Internet for the purpose of end-to-end connectivity, the challenges outlined above would not be solved with the current Internet architecture. Separation of location and identity of a mobile device ensures that changing a point of attachment to a network or changing the active interface on a device does not affect ongoing sessions, since the transport (and upper) layer sockets are bound to the device's *identity* and routing is achieved using the device's *locator*. It was on this premise that Locator Identifier Split (or simply Loc/ID [5-7]) protocols, such as *Locator Identifier Separation Protocol-Mobile Node* (LISP-MN) [7, 8], were conceived.

On the other hand, *Mobile IPv6* (MIPv6 [9]) and its extensions – such as Mobile IPv6 Fast Handovers and Hierarchical MIPv6 – are designed mainly to enable mobility on the Internet through the use one IP address for routing while a *mobile node* (MN) is on the home link and a different address when the MN moves to a foreign network. A mobility anchor at the MN's home network maintains the relationship between the two addresses. This is in contrast to most Loc/ID protocols which have no concept of home network and the *identifiers* and *locators* are usually mapped using a global mapping system. It is currently debated [7, 10-12] if the MIPv6-based mobility approaches shall be maintained in future networks or whether we will see the adoption of one of the proposed Loc/ID mechanisms – which include the LISP-MN, Host Identity Protocol (HIP), Site Multihoming by IPv6 Intermediation (SHIM6), Identifier Locator Network Protocol (ILNP) etc. A new Personal IPv6 (PIPv6) address is also proposed [13] to be used as node identity in mobile and vehicular ad hoc networks, as well as wireless sensor networks. There are many advantages in adopting Loc/ID protocols in the future networks including improved routing scalability, support for multi-homing, and support for traffic engineering as well as simplified renumbering. There is also the potential for having MNs providing services (as remote servers) on the move. In this work, we aim to evaluate LISP-MN performance in the inter-domain mobility scenario for both multi-interface and single interface MNs with a focus on throughput, handover delay, service

disruption time and packet loss. Our main contributions with this work include:

- to the best of our knowledge, this is the first performance evaluation of the LISP-MN protocol on a laboratory testbed that focuses on inter-domain mobility for both heterogeneous and homogenous environments;
- investigating the impact of the protocol's handover management process on TCP traffic; and
- providing a critique of LISP-MN suitability for future wireless network environments by comparing it with the IETF standardised MIPv6.

For multi-interface MNs, our experiment shows that LISP-MN has a better response in *soft handover* (SH) scenarios in terms of throughput and packet loss. MIPv6 on the other hand shows shorter handover delay with faster session resumption time in a *hard handover* (HH) scenario. Both protocols demonstrate poor performance for a single interface MN due to long handover delay experienced.

The rest of the paper is arranged as follows: Section II presents an overview of LISP, LISP-MN and MIPv6. Section III details the experiment while performance evaluation & discussion is presented in IV. Section V looks at the related work and Section VI is the conclusion.

II. OVERVIEW OF PROTOCOLS

A. Locator Identifier Separation Protocol

To understand LISP-MN, it is necessary to provide a brief on its parent protocol. LISP [14, 15] is a map-and-encapsulate tunneling protocol that enables the separation of *identity* and *location* of a host in the IP network. The protocol provides 2 address spaces (both IP addresses), the *endpoint identifier* (EID) serving as the node *identity*, and *routing locator* (RLOC) to determine the position of the node on a network. The EIDs are obtained from the EID block space, and are independent of the RLOCs. EIDs are used for intra-domain routing (routing within an autonomous system). RLOCs, on the other hand, are globally routable addresses used for inter-domain routing and assigned to the border router named *egress/ingress tunnel router* (ETR/ITR), which marks the entry and exit point of a domain. The ingress and egress functionality may be collocated on a single tunnel router in a domain and simply referred to as xTR. While the ITR maps the destination EID of an outgoing packet to its corresponding RLOC by sending a *map-request* to the mapping system, the ETR on the other end receives and delivers packets destined to one of its EID prefixes.

A distributed database tree (DDT) – conceptually similar to DNS – termed LISP-DDT [16] is the current mapping system in use. As shown in Fig. 1, to communicate with HOST_B in LISP network, HOST_A resolved the IP of HOST_B using the DNS, as per a normal IP session (1); EID_B1 is returned in the process (2). The packet is then forwarded towards the default gateway, which is the ITR (3). The ITR sent a *map-request* to the map-resolver asking for the RLOC(s) of the ETR serving the requested EID (4). The map-resolver queried the LISP-DDT for the mapping and received a response (with the map-server address) by either the map-server or the DDT root itself (5 and 6). The map-resolver sent the *map-request* to the map-server and received a *map-reply* response; and the *map-reply* is delivered to the originating ITR (7, 8, and 9). The ITR then encapsulated the packet to its destination (10). Note that the map-server must have been earlier delegated (0) to respond to *map-request* on behalf of the destination ETR. Hence the map-server publishes the EID prefixes in the mapping system on behalf of the tunnel routers that it is serving. The ETR, on the other hand, de-encapsulates incoming packets destined to an EID within its control and forwards accordingly. Replies would normally be reverse-tunneled to their destination. To optimise performance, the ITR caches some routes to speed up the routing process and avoid querying the mapping system every time a communication channel is to be established.

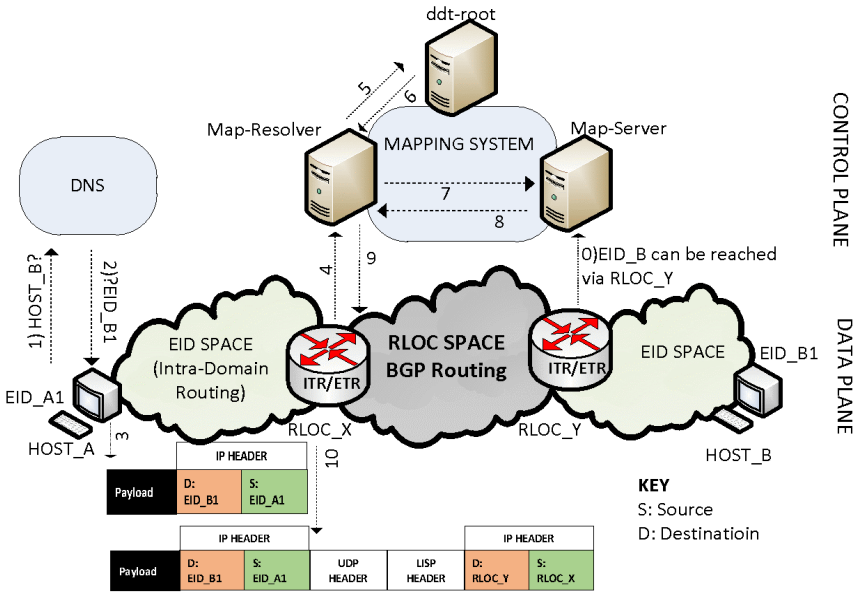

Fig. 1. LISP Protocol Data and Control Planes

The caches are refreshed after a pre-defined timeout to avoid storing stale routes. For communication with a non-LISP-domain (when ITR receives no response from the mapping system), the ITR may simply forward the packet to the destination without encapsulation. But since ingress filtering is usually deployed by many ISPs, the ITR would encapsulate and forward the packets to a *proxy* ETR (PETR), which then forward the packet to its final destination. The PETR would normally be located in networks that do not have ingress filtering and must have the sending domain's EID prefix pre-configured in its database. Replies from non-LISP-domains are sent to the *proxy* ITR (PITR) serving the LISP-domain, which encapsulates the packet and forwards it to its destination.

B. *LISP-MN*

LISP-MN [8] is an approach defined to enable mobility with the LISP protocol. An MN is equipped with a lightweight version of ITR/ETR functionality and behaves like a single LISP-domain. The mobile device is configured statically with an EID which is used by the transport and application layer to identify communication sessions. The map-server serves as the mobility anchor and tracks the location of the MN at any given time. For communication with non-LISP *correspondent node* (CN), the MN forwards and receives all packets via the PETR and PITR respectively.

Once an MN comes online or moves to a new network, it configures a new IP address (RLOC) and sends a *map-register* message to its map-server in order to register the RLOC (its location). The server will authenticate the EID and reply with a *map-notify* message confirming that the EID-RLOC registration has been successful and an up-to-date mapping is published on the *mapping system*. The MN will also send a *solicit map request* (SMR) message to its PITR - and to any LISP-based CN to invoke a mapping update. Consequently, the PITR and the CN would send a *map-request* to the MN, to which the MN replies with a *map_reply* containing the MN's new RLOC. This ensures that the PITR and LISP-based CN have an up-to-date mapping of MN's location.

As highlighted in [8], LISP-MN can be deployed to work in five distinct mobility cases:

Scenario 1: LISP-MN establishing a communication session with a stationary node in a LISP-domain;

Scenario 2: LISP-MN establishing a communication session with a non-LISP-domain;

Scenario 3: LISP-MN establishing a communication session with another LISP-MN;

Scenario 4: Non-LISP-domain communicating with LISP-MN

Scenario 5: LISP-domain communicating with a LISP-MN.

Although these scenarios have been expanded to 9 in [17], a gradual implementation of LISP-MN to work side-by-side with the legacy Internet would mean scenario 2 becomes the common implementation for the fact that the different servers on the Internet that form the bulk of the CNs today are mostly located in non-LISP-domains and are themselves not LISP-capable. As such LISP-enabled devices are likely to be the ones establishing a communication session with these CNs and rarely the other way round. We focus on this scenario in our work for these reasons and Fig. 2 shows the data plane operation of the scenario.

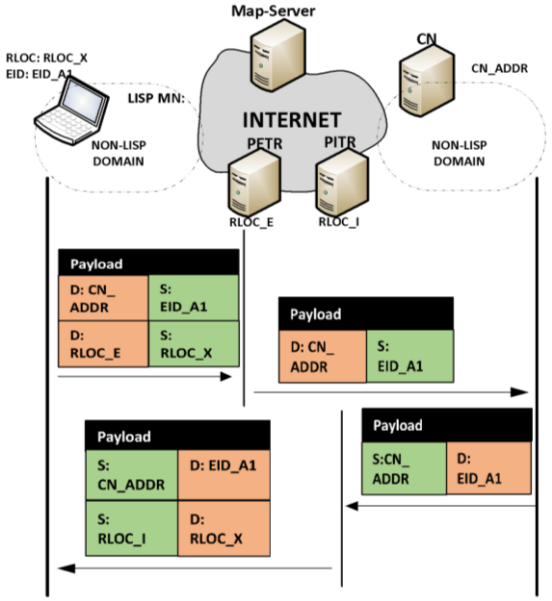

Fig. 2. LISP-MN Data Plane Operation

As shown in Fig. 2, an MN would encapsulate all outgoing packets to the PETR (with the exception of management protocols such as dhcp), and the router de-encapsulates the packets and forwards using conventional Internet routing. Replies are sent by the CN using the MN's EID but would be delivered by the Internet routing infrastructure to the PITR serving the EID, which forward the packets to the MN. The PITR would advertise reachability of the MN's EID prefix in the default-free zone, to enable communication between LISP and non-LISP-domains. The PITR would learn of any change in the MN's location by either contacting the map-server or through the SMR message explained earlier.

C. *MOBILE IPv6*

MIPv6 is an IETF mobility protocol and like other mobility protocols, it is targeted at maintaining communication sessions during an MN's handover by using a non-mutable IP address, termed *home address* (HoA), for end-to-end connectivity. While in the home network, the MN uses the HoA as it would a normal IP address and all communication and routing of packets are done using the same address. On a foreign network, the MN acquires/configures a new IP address called a care *of address* (CoA). The relationship between the two addresses is maintained using a mobility anchor, termed a *home agent* (HA), at the MN's home network. Henceforth, the HoA is used in forming the transport (and upper) layer sockets and the CoA is used for routing.

To register the CoA, the MN sends a *binding update* (BU) to the HA and the HA authenticates the message and replies with a *binding acknowledgement* (B_Ack) indicating *binding* completion. The protocol specification mandates the

TABLE I. Analytical comparison of LISP-MN and MIPv6

FEATURE	LISP-MN	MIPv6
Registration	Five control messages: *map-register, map-notify* with map-server; SMR, *map-request and map-reply* with PITR	Two control messages: BU and B_Ack with HA
Tunneling	Add 56 bytes to an IPv6 packet by using UDP encapsulation	Uses IP-in-IP tunneling and adds 40 bytes IPv6 header
Routing	Packets sent via the PETR, replies via the PITR	Packets sent and received via the HA
IP version agnosticism	IPv4-in-IPv6 encapsulation capable	Not available
Traffic Engineering (TE)	Dynamic TE possible with multiple locators	Not available
Media Redirection	Can use more than one proxy tunnel router	Can also use multiple HAs
Triangular routing	Avoided in LISP-based networks but necessary in the scenario being evaluated.	Direct communication can be achieved with route optimisation.

authentication of the *binding messages* using security mechanisms such as *Internet key exchange* or IPsec. As shown in Fig. 3, packets to and from the MN, while on a foreign network, are routed via the home network. On receiving these packets, the HA tunnels them to the MN's current location with its IP address as the source and MN's CoA as the destination. The MN replies to the CN with the help of *reverse-tunneling*, by sending the packet back to the HA for onward delivery to the CN. Routing can further be optimised when communicating with CN with the MIPv6's *route-optimisation* (RO) feature, and direct communication can be achieved between the two nodes using the MN's CoA as source. And as shown in the figure, a *return routability test* is necessary to achieve such a level of optimisation.

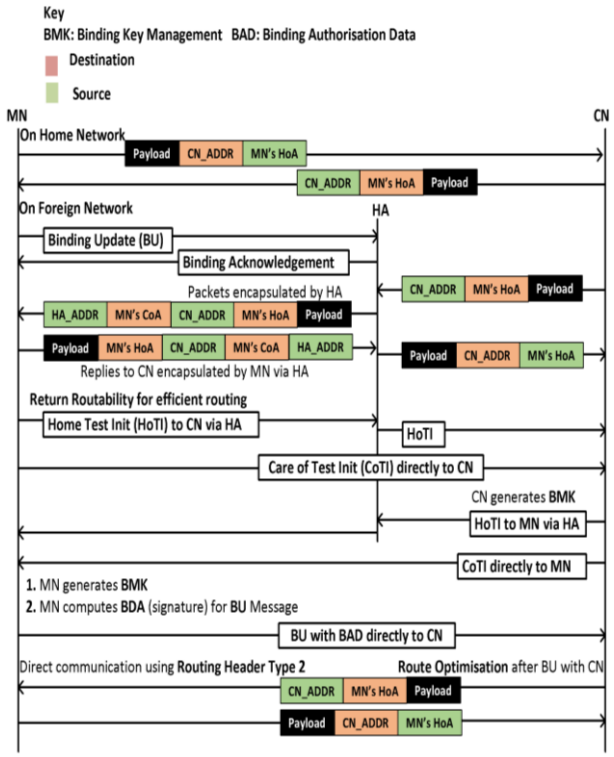

Fig. 3. MIPv6 Control and Data Plane Operation

For analytical comparison of MIPv6 with LISP-MN, see Table 1.

III. EXPERIMENT

A. Testbed Setup

We set up an Ethernet network as shown in Fig. 4. The MN has access to the network via AR1 and AR2, through which it connects to the relevant nodes to enable communication. The testbed reflects an IPv6 public network with dynamic routing configured to enable reachability. All the 8 nodes in the testbed are desktops running Ubuntu 14.04; with the minimum of Pentium (R) 3.20 GHz CPU and 4GB RAM. The wireless links on the MN are set to 2.7Mbps uplink and 3.3Mbps downlink to represent average Wi-Fi throughput.

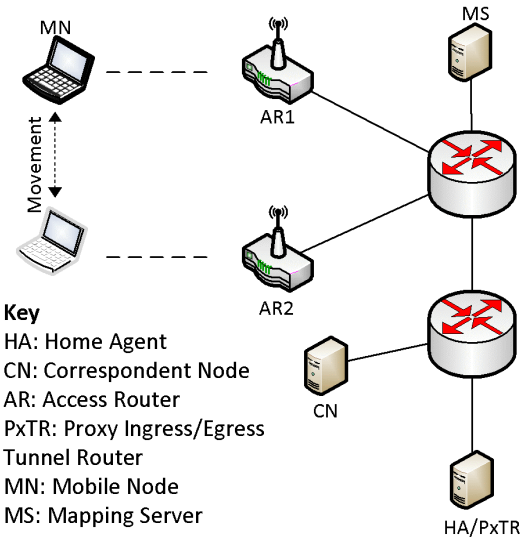

Fig. 4. Testbed Architecture

To ensure that all network and system parameters are the same for both protocols in the testbed, LISP-MN and MIPv6 are run on the same device and the HA and *Proxy Ingress/Egress Tunnel Router* (PxTR) are collocated as well.

LISP-MN Configuration

For LISP-MN, we use the LISPmob [18] implementation to configure the LISP components, which include the MN, map-server/map-resolver, and PITR/PETR. LISPmob was initially developed by Cisco but is currently maintained by Barcelona Tech University.

MIPv6 Configuration

For MIPv6, we use the UMIP code [19], for both the MN and HA. UMIP was originally developed by the USAGI project and is currently maintained by *umip.org*.

B. Mobility Scenario

We take a common Heterogeneous Mobility Scenario (HeMS) today where a user leaves an area – home, office or a public space – with her mobile device whilst connected to the available Wi-Fi within the vicinity and streaming a video over TCP. As she walks away from the place and loses connectivity, the device automatically switches to her 3/4G cellular service for continuous streaming. We also look at the reverse of this scenario, where the user comes into a place with a pre-configured Wi-Fi link available and her connectivity immediately switches back to the Wi-Fi. We also look at a Homogeneous Mobility Scenario (HoMS) where a user, streaming a video over a Wi-Fi network in places like a train station, switches over to another Wi-Fi network.

At the beginning of the HeMS experiment we bring the two interfaces up with first priority interface in active state. The MN establishes a communication session with the CN using the active interface, and as soon as the interface goes down, all communication is switched over to the second interface. We termed this event a *hard handover* (HH) because of the abrupt loss of connectivity. The MN brings up the first priority interface again after a period of time to force a switch-over to it, and we termed this event a *soft handover* (SH) because both of the two interfaces become active for some time before a switch is finally made to the interface with high priority. As for HoMS, the MN switches connection between AR1 and AR2 while communicating with the CN.

IV. PERFORMANCE EVALUATION AND DISCUSSION

We take measurements for *throughput*, *handover delay* and *service disruption time* (SDT) by using the *tcpdump* program to capture and analyse the TCP traffic sent by a *server-client* program on the CN to the MN. We also used *iperf* to send datagrams from the CN during a handover event to measure the amount of packet loss caused by the handover. Noise traffic, in the form of TCP and ICMPv6 packets, were also running as competing traffic on the links. The results presented below are averages of ten runs for each experiment.

A. Throughput

Both LISP-MN and MIPv6 achieved similar link utilisation as seen in Fig. 6 and 7 for a TCP session, in both HeMS and HoMS. The main difference is the reaction of the mobility protocols to handover events in a HeMS. As can be seen in Fig. 6, LISP-MN suffers a significant drop in the HH event, going as low as 82kbps from 1917kbps compared to MIPv6's 1410kbps drop from a similar peak. This deterioration in LISP-MN throughput is caused by the 635.5 milliseconds (ms) delay in handover. Conversely, LISP-MN performs better in the SH event maintaining the same throughput level throughout. MIPv6, on the other hand, experienced a drop in throughput during the SH.

The main reason for MIPv6's drop in throughput during the SH is the fact that the protocol uses the just brought up interface to exchange the handover messages and then immediately switches communication session to the new interface. The

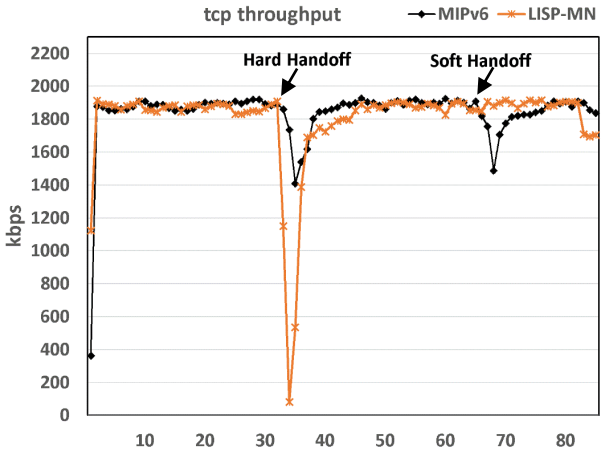

Fig. 5. LISP-MN vs MIPv6 TCP Throughput in Heterogeneous Mobility Scenario

abrupt switch of interface (on average 15ms after handover messages) causes a little slowdown in the packet delivery and the likelihood of dropped packets by the previous access router, hence the drop in throughput. LISP-MN in contrast, uses the current interface to exchange the handover messages with the

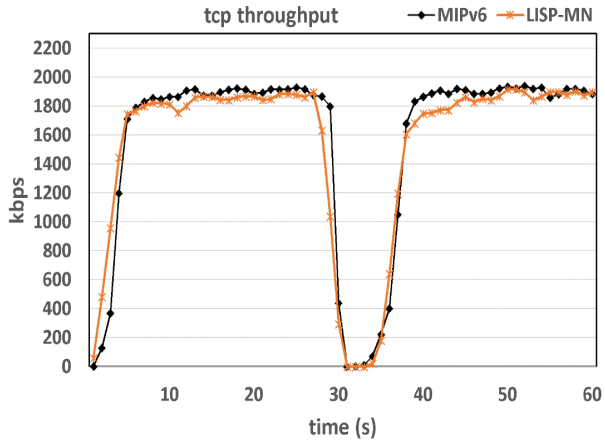

Fig. 6. LISP-MN vs MIPv6 TCP Throughput in Homogeneous Mobility Scenario

PITR and continues to use the interface – for up to 113 ms – before the eventual handover to the target interface.

As the result of the long handover delay and break in data transmission in HoMS, both protocols experienced a significant drop in throughput going down below 5 kbps for 3s in LISP-MN, and 2s in MIPv6. This drop causes packet losses and will have an adverse effect on the performance of loss-sensitive applications such as VoIP and video conferencing. But whilst MIPv6 takes 9s between the start of the handover event and TCP throttling to full capacity after the switch to the new link, LISP-MN requires 14s to reach its peak due to its longer SDT.

B. Handover Delay and Service Disruption Time

In this section we analyse, the different elements that are involved in a handover scenario to understand the contribution to the delay of each entity involved in the handover event. Handover delay is the time during which an MN cannot exchange packets with its CNs. Because of the very low handover delay for SH in HeMS, no service disruption was experienced and very few packets were lost in the process. It is a different case in a HH event and for this reason, we will focus our analysis on the HH event. We define handover delay as the time between the first priority interface going down until the last handover message is sent (map_reply from the MN, in the case of LISP-MN) or received (B_Ack from the HA, in the case of MIPv6) by the second interface. We will also look at the consequent SDT from the handover delay, which is measured from the time we receive the last TCP packet on the first link (the link serving the first priority interface) to the time we receive the first packet on the second link. For clear understanding, we define three different delay variables D1, D2 and D3 below and presents their results in Table II.

D1: From current interface down until first handover message is sent using the new interface – map_register for LISP-MN and BU for MIPv6.

D2: From the first handover message until the last handover message – map_reply for LISP-MN and B_Ack for MIPv6.

D3: From last handover message until data session resumes.

Note that the addition of D1 and D2 produces the handover delay, and the addition of the 3 delay periods produces the SDT.

TABLE II. Handover delay and Service Disruption Time in a Hard Handover

Protocol	**Milliseconds**				
	D1	***D2***	***Handover delay***	***D3***	***SDT***
LISP-MN	532	104	636	401	1037
MIPv6	3	104	107	5	113

MIPv6 shows better performance in handover delay with only 107ms to LISP-MN's 636ms although it takes a similar time to send LISP-MN's five control messages against MIPv6's two. The 636ms delay by LISP-MN is owed to the protocol's delayed response to change in interface, D1, up to 532ms. This causes packet losses as we will subsequently see in the following section. Nevertheless, introducing D1 is a conscious decision by the LISPmob project – the developers of the LISP-MN protocol – so that fluctuations in an MN's interface (several interface changes can happen in a short amount of time on mobile devices' interfaces) would not trigger a false-positive handover event leading to a failed handover.

MIPv6 has faster response to a handover event and hence low packet loss but with high possibility of failed handovers due to high handover blocking probability - a situation where by the handover duration is higher than the MN's residence time on the target network and as such a handover cannot be completed. Frequent failed handovers will also cause a flooding of control messages into the network. The LISP-MN delayed response, on the other hand, mitigates these failed handover events thereby preventing any unnecessary handover traffic being sent into the network; but it does so at the expense of more packet losses. Furthermore, D1 and D3 makes LISP-MN soft handover event smoother, with no visible drop in TCP throughput (Fig. 6) throughout the event as opposed to the MIPv6 which shows a sign of throughput drop during the SH event.

For HoMS, we define handover delay as the time between disconnecting from the current link until the last handover message is sent/received. SDT remains the time between the last data packet before handover until the first packet after. D2 and D3 are the same as defined earlier while D1 is from disconnecting the current link until the first handover message is sent using the new link. The results are given in Table III. There is 3156ms – more than 3 seconds – delay before the first handover message is sent by both protocols. The long delay involves the necessary layer 2 verification messages using the EAPOL protocol as well as layer three address configuration processes. These processes include movement detection (i.e sending Router Solicitation and receiving Router Advertisement), CoA configuration and duplicate address detection, bringing the handover delay for LISP-MN to 3269ms and 3499ms for MIPv6.

We can see that in this scenario, LISP-MN responded to the change of state in the interface as soon as the layer three address configuration process was completed by starting the handover process even earlier than MIPv6. Because D1 (3156ms) is by far higher than the 'cooling-off period' of 532ms we noticed in the HeMS, LISP-MN needed not to wait any further to start the mobility signaling as soon as the IP address is configured on the interface. Both protocols experienced a long delay of more than three seconds and consequently long SDT. D3 for both protocols is quite high and likely caused by TCP on the CN withholding packets for lack of acknowledgements during the handover period causing the significant SDT.

Table III. Handover delay and Service Disruption Time for a Single Interface Mobile Node

Protocol	**Milliseconds**				
	D1	***D2***	***Handover delay***	***D3***	***SDT***
LISP-MN	3156	111	3269	2371	5640
MIPv6	3172	327	3499	1842	5341

The long delays in both protocols will affect many applications not only delay and loss sensitive types but also 'best effort' type applications such as Internet browsing and (most non-multimedia) mobile apps, as users will experience slow loading of webpages and apps' response respectively. This may also cause network congestion when MN's keep sending TCP retransmission requests over the network after a handover event.

C. Packet Loss

All packets destined to the MN at the point of handover would be dropped by the network unless a rule exists to buffer or tell the ARs what to do with such packets. These dropped packets are counted as lost packets and are directly proportional

to handover delay. We measured the loss by sending UDP streams from a CN using the *iperf* tool, and initiate a handover event on the MN during the period. Table IV shows the performance of LISP-MN and MIPv6 in HeMS.

HH with LISP-MN results in the most loss in packets (up 92 datagrams of the 5,530 sent during the one minute period) as expected due to the long handover delay experienced. The protocol performs better in the SH recording only 10 datagrams lost, just 0.18% over the period of 1 minute. MIPv6 recorded very low packet loss for both HH and SH, the loss experienced by both protocols in the SH event can be tolerated by even loss-sensitive applications such as video conferencing and VoIP.

Table IV. Packet Loss in Heterogeneous Mobility Scenario

	Datagrams lost (in one minute)	
Protocol	***Hard Handover***	***Soft Handover***
LISP-MN	92 (1.6%)	10 (0.18%)
MIPv6	30 (0.55%)	24 (0.42%)

The same amount of packet loss is recorded for both protocols in a HoMS with 8.4% for LISP-MN and 8.3% for MIPv6 over the period of one minute as shown in Table V. These losses are caused by the long delay experienced by the protocols at the point of handover. The level of loss is enough to perturb many best-effort type applications, and will break loss-sensitive ones. The high drop in throughput we see in Fig. 7 and for the HH event in Fig. 6 are as a result of many packets being lost as the MN's interface is reconfigured and mobility control messages are exchanged.

Table V. Packet Loss in Homogeneous Mobility Scenario

Protocol	**Datagrams lost (in one minute)**
LISP-MN	468 (8.4%)
MIPv6	466 (8.3%)

V. RELATED WORK

An analytical review of LISP-MN was performed by Menth et al. [17] in which the use case scenarios of the protocol were expanded from the original five presented in [8] to nine. The authors also found that an MN does not always have to tunnel its packets to the PETR in order to hide its non-routable EID as proposed in the protocol's original specification [14]. Guided by a study which showed that up to 31% of ISPs do not perform ingress filtering of addresses, the authors proposed that rather than tunneling all LISP packets, an MN should perform a filter-check when coming on to a network, and should forward its packets san encapsulation if ingress filtering is not enabled on the network, thereby reducing the tunneling cost on the links.

Klein et al. [20] also analysed LISP-MN performance behind NAT and found that although MNs can start a connectivity process from behind the NAT, they cannot receive any reply therefrom. This is because the map_register message sent to the map-server contains the MN's private IP address, and as such when a remote ITR request for the RLOC of the MN in other to send packets to it, the private address is returned by the map-server which cannot be used for routing on the public network. The authors proposed an indirection through a NAT Traversal Router (NTR) that intercepts map_register messages from the MN (the MN is configured with NTR address as map-server and use a different port number from the one in the protocol's specification) and registers its (the NTR's) IP address with the MS, as the MN's RLOC. The NTR also saves the NAT device's IP address and the relevant port numbers in order to be able to send replies to the MN. Subsequently, packets destined to the MN are sent via the NTR, which forwards them to the NAT device using the stored information for eventual delivery to the MN.

Although we have not found any work on LISP-MN that compares the protocol to MIPv6 and/or its extension in inter-domain mobility scenarios, there is however a body of work where such comparisons are performed with other Loc/ID protocols. For example, Muslam et al. [21] proposed a network-based Loc/ID mobility solution termed mobility-enabled HIP and compared its performance with HIP, Micro-HIP and PMIPv6 using the OMNET++ simulator. Handover latency, packet loss and signaling overhead were used as metrics for evaluating the protocols' performance for both inter and intra-domain mobility scenarios. OMNET++ was also used by Mugga et al [22] to compare the handover latency and rehoming performance of the HIP protocol with MIPv6, Multiple Care-of-Address Registration (MCoA), and the Stream Control Transport Protocol (SCTP) . They found HIP to perform better in both mobility and multihoming scenarios than the other protocols because of its low signaling overhead and its soft handover feature that significantly reduced the rehoming time by a large factor. Phoomikiattisak et al. [23] used an overlay network laboratory testbed to evaluate the ILNP (version 6) protocol's soft handover feature in an inter-domain mobility scenario as a proof of concept experiment. Three different network conditions – LAN, MAN and WAN – were emulated to measure end-to-end delay and loss induced by the protocol during communication. An analytical comparison of the protocol with MIPv6 was also presented in the paper based on complexity and scale, signaling overhead and security.

VI. CONCLUSIONS

In this work we focused on the very important area of inter-domain mobility in both heterogeneous and homogeneous network environments, and how LISP-MN in particular handles such an important event in comparison to the standardised MIPv6. We considered a scenario where a LISP-capable MN communicates with a non-LISP CN in the legacy Internet. Going by the results and the analytical comparison presented in this work, we draw the following conclusions:

- That LISP-MN is suitable in soft handover – where a handover event involves two interfaces in active-active state before the change in connectivity. It is also more suitable where interface fluctuations and unstable wireless links are frequent because of its ability to hold on to its current connection for a defined period before initialising a handover event.
- MIPv6 on the other handover performs better in scenarios where wireless links are stable and

fluctuation of a mobile device's interfaces are not envisaged. It is also the ideal protocol where hard handovers are frequent in a network.

- Both protocols on their own are suboptimal when a single interface mobile device moves between domains, and both will require an external support, such as buffering incoming packets on the access routers during the handover and then forwarding the packets to the mobile device's target network.

Although both protocols have their strengths and weaknesses as highlighted, the IP version agnosticism (IPv4-in-IPv6 encapsulation for instance) and dynamic traffic engineering with multiple locators features of LISP-MN that are not available with MIPv6 give the former a slight advantage in the inter-domain mobility environment. We believe that these additional features provide a stronger argument for adopting LISP-MN in the future heterogeneous wireless networks.

REFERENCES

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2014–2019 White Paper," *Cisco Systems Inc.,* San Jose, CA, 2015.

[2] Z. Gao, A. Venkataramani, J. Kurose, and S. Heimlicher, "Towards a Quantitative Comparison of the Cost-Benefit Trade-offs of Location-Independent Network Architectures," in *ACM SIGCOMM'14*, Chicago, IL, Aug., 2014.

[3] A. Rodríguez Natal, L. Jakab, M. Portolés, V. Ermagan, P. Natarajan, F. Maino*, et al.*, "LISP-MN: Mobile Networking Through LISP," in *Wireless Personal Communications,* vol. 70, issue 1, May 2013, pp. 253-266.

[4] T. Li, "Design Goals for Scalable Internet Routing," *IETF RFC 6227*, May, 2011.

[5] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB Workshop on Routing and Addressing," *IETF RFC 4984*, Sep. 2007.

[6] W. Köpp and A. Klein, "Locator/Identifier Split," in *Proceedings of the Seminars Future Internet (FI) and Innovative Internet Technologies and Mobile Communications (IITM)*, Munich, Germany, Jul. 2011, pp. 62-70.

[7] W. Ramirez, X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, A. Martinez, and M. S. Siddiqui, "A survey and taxonomy of ID/Locator Split Architectures," in *Computer Networks,* vol. 60, 26 February 2014, pp. 13-33.

[8] D. Farinacci, C. White, D. Lewis, and D. Meyer. LISP Mobile Node, *IETF Internet Draft, draft-meyer-lisp-mn-12,* Jan. 2015.

[9] C. Perkins, J. Arkko, and D. Johnson, "Mobility Support in IPv6," *IETF RFC 6275,* Jul. 2011.

[10] C. So-In, R. Jain, S. Paul, and J. Pan, "Future wireless networks: key issues and a survey (ID/locator split perspective)," in *International Journal of Communication Network and Distributed System* vol. 8, no.1/2, 2012, pp. 24 - 52.

[11] H. Tuncer, S. Mishra, and N. Shenoy, "A survey of identity and handoff management approaches for the future Internet," *Computer Communications,* vol. 36, issue 1, 1 December 2012, pp. 63-79.

[12] R. Moskowitz and P. Nikander. "Host Identity Protocol (HIP) Architecture"*, IETF RFC 4423*, May 2006.

[13] I. Ganchev and M. O'Droma, "New personal IPv6 address scheme and universal CIM card for UCWW," presented in *7th International Conference on ITS Telecommunications (ITST)*, Sophia Antipolis, France, 2007, pp. 1-6.

[14] D. Farinacci, D. Lewis, D. Meyer, and V. Fuller, "The Locator/ID Separation Protocol (LISP)," *IETF RFC 6830*, Jan. 2013.

[15] D. Meyer, "The Locator Identifier Separation Protocol (LISP)," *The Internet Protocol Journal,* vol. 11, no 1, March 2008, pp. 23-36.

[16] V. Fuller, A. Jain, D. Lewis, and V. Ermagan, "LISP Delegated Database Tree," *IETF Internet Draft, draft-ietf-lisp-ddt-03.txt*, Apr. 2015.

[17] M. Menth, D. Klein, and M. Hartmann, "Improvements to LISP Mobile Node," in *22nd International Teletraffic Congress (ITC),* Amsterdam, The Netherlands, Sep. 2010, pp. 1-8.

[18] LISPmob*,* "*An open-source LISP implemantation for Linux, Android and OpenWRT*" [Online] 2015. Available: http://lispmob.org [Accessed: May 31, 2015].

[19] UMIP "*Mobile IPv6 and NEMO Basic Support implementation for Linux*". [Online] 2014. Available: http://www.umip.org/ [Accessed: Jun. 12, 2014].

[20] D. Klein, M. Hartmann, and M. Menth, "NAT traversal for LISP mobile node," in *Proceedings of the Re-Architecting the Internet Workshop*, Philadelphia, PA, 2010, pp. 1-6.

[21] M. M. Muslam, H. A. Chan, L. A. Magagula, and N. Ventura, "Network-based mobility and Host Identity Protocol," in *IEEE Wireless Communications and Networking Conference (WCNC)*, Paris, France, Apr. 2012, pp. 2395-2400.

[22] C. Mugga, D. Sun, and D. Ilie, "Performance Comparison of IPv6 Multihoming and Mobility Protocols," in *The Thirteenth International Conference on Networks* (ICN), Nice, France, 2014, pp. 166-171.

[23] D. Phoomikiattisak and S. N. Bhatti, "Network layer soft handoff for IP mobility," in *Proceedings of the 8th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks,* Barcelona, Spain, 2013, pp. 13-20.