LISP-ROAM: Network-based Host Mobility with LISP

LISP-ROAM: Network-based Host Mobility with LISP

Andrea Galvani
Embrane Inc.
andrea@embrane.com

Alberto Rodriguez-Natal
Universitat Politècnica de Catalunya
arnatal@ac.upc.edu

Albert Cabellos-Aparicio
Universitat Politècnica de Catalunya
acabello@ac.upc.edu

Fulvio Risso
Politecnico di Torino
fulvio.risso@polito.it

ABSTRACT

The LISP-MN protocol is an extension of the Locator/ID Separation Protocol (LISP) that support end-host IP mobility and that, to operate, requires updating the software of the mobile terminal. However in several scenarios this is a major roadblock to effectively deploy mobility. On the one hand the operator must support the implementation over a wide range of devices and on the other hand, end-host mobility does not provide sufficient control to the operator itself. In this paper we present LISP-ROAM, a LISP extension to support network-based end-host mobility. With LISP-ROAM, end-hosts remain completely unmodified while the network provides the mobility support by assigning the same IP address regardless of their network attachment point. The paper describes the protocol and presents an experimental evaluation of the performance of LISP-ROAM implemented on top of LISPmob, an open-source LISP implementation.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General; C.2.2 [**Computer-Communication Networks**]: Network Protocols

Keywords

LISP, Network-based mobility

1. INTRODUCTION

The recent LISP-MN [2] proposal leverages the capability of the Locator/ID Separation Protocol (LISP) [3] to decouple Endpoint Identifiers (EIDs) from Routing Locators (RLOCs) to provide mobility services. However, LISP-MN assumes the installation of additional software on the mobile terminal, which modifies the normal behavior of its TCP/IP stack. As a result, despite the many proposals that aim at introducing the support for host mobility in the Internet, those services gained attraction only in some particular environments. Consequently, most of the hosts are still forced to re-initialize all the established connections when moving from a network to another (e.g., WiFi to 4G), and, often, even when switching to a different technology within the network of the same operator, such as from 4G to 3G.

This paper presents LISP-ROAM, a LISP extension to support network-based end-host mobility that, differently from other proposals, allows end-hosts to stay unmodified. Mobility support is provided by some additional/modified network components (DHCP servers, authentication services, LISP xTR and servers) that cooperate to assign always the same IP address regardless of the host network attachment point, even when the user terminal attaches to a different provider. This would allow user terminals equipped by a standard TCP/IP protocol stack to change their network attaching point without impacting their network experience and without dropping any active transport-level session, and it would offer to network providers a better control over mobility services provided to their customers.

The paper describes the protocol, which has been engineered in order to minimize the technical/administrative burden of its deployment within the network of an operator, and it presents an experimental evaluation of the performance of LISP-ROAM implemented on top of LISPmob [6], an open-source LISP implementation.

This paper is organized as follows. Section 2 provides some background about LISP, while the LISP-ROAM protocol is described in Section 3 and its implementation in Section 4. The experimental validation of our proposal is presented in Section 5; related work is summarized in Section 6 and Section 7 draws some conclusive remarks.

2. LISP

The Locator/ID Separation Protocol (LISP) decouples identity from location on the current IP addresses by creating two separate namespaces, Endpoint Identifiers (EIDs) and Routing Locators (RLOCs). Packets are routed based on EIDs on the edge sites and on RLOCs on the transit networks. To connect disjoint EID spaces through the RLOC space, LISP follows a map-and-encap approach. EIDs are mapped to RLOCs and LISP Ingress/Egress Tunnel Routers (xTRs) are deployed on the EID-RLOC edges. xTRs encapsulate EID packets into RLOC packets that can be routed through the transit network. LISP introduces a distributed Mapping System to keep EID to RLOC mappings. The Mapping System is composed by Map-Servers, that store

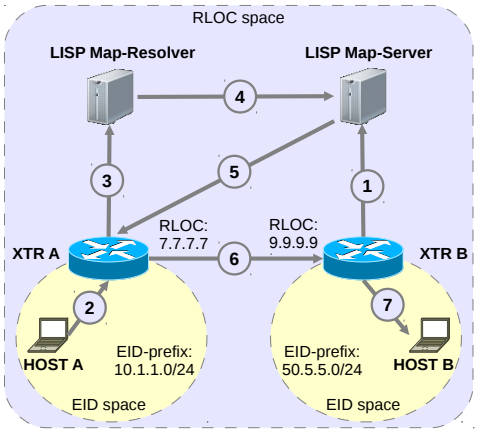

Figure 1: LISP overview

mappings, and Map-Resolvers, that locate the appropriate Map-Server for a specific mapping.

Figure 1 depicts basic LISP operation. xTR B has a designated Map-Server where it registers its EID to RLOC information via a Map-Register message (1). When xTR A receives traffic addressed to host B (2) it request the RLOC of host B via sending a Map-Request to a Map-Resolver (3). This request is routed within the Mapping System to the appropriate Map-Server (4). Normally, the Map-Server would forward this request to the xTR whose mapping has been requested, however xTR B has authorized its Map-Server to respond to Map-Request on its behalf. Therefore the Map-Server sends a Map-Reply (5) with the mapping information. xTR A receives the message and caches the mapping information for future use. From now on, all packets addressed to Host B on xTR A will be encapsulated towards xTR B (6). xTR B will receive these packets, decapsulate them and forward the decapsulated traffic towards Host B (7).

3. LISP-ROAM

3.1 Overview

LISP-ROAM originates from the observation that the TCP/IP stack can tolerate a temporary loss of connectivity without dropping the established transport-layer sessions (up to 25 seconds on most common operating systems according to our tests).

LISP-ROAM exploits this property by extending LISP to assign always the same IP address to the mobile terminal even if it attaches to a different network, so that the host perceives the handover as a transient disconnection from the network. This requires the network infrastructure to be updated in order to recognize the user terminal when it connects to a network (e.g., through a WiFi access point) and to assign always the same address to it. Furthermore, the network infrastructure has to be dynamically reconfigured to handle that address, which may not belong to the "foreign" network address space, and then update the proper LISP bindings to redirect all the traffic to the new location of the user.

Following sections will present the main components of this architecture and the detailed operations of the LISP-ROAM proposal.

3.2 Architectural assumptions

The main scenario targeted by LISP-ROAM is a confederation of Internet Service Providers that collaborate to offer a new mobility service to their costumers, specifically to enable all the subscribers to roam across every Wi-Fi network of the participating ISPs without dropping their active connections. It is assumed that:

- All the ISPs participating in the service establish a trust agreement in order to allow foreign users connect to their networks and update their location, e.g., by sharing the secrets needed to update users' location.
- Each user is tied to her "home" domain, which is the ISP she is subscribed to, and is considered "foreign" in all other domains.
- Each user has a fixed EID (i.e., IPv4 or IPv6 address), assigned when she subscribes to the service.
- Mobile user terminals connect to edge networks served by LISP-ROAM compatible routers; following LISP naming, we refer to edge routers as xTRs.
- All participating networks implement the 802.1x standard on the access side.

3.3 User authentication

User authentication is required to identify the subscriber that is attaching to a given access network and assign to the mobile device always the same IP address. Upon discovering the user identity[1], the system has to detect the user's EID and her home domain, which is needed to update the user's location in the home LISP Map-Server.

A simple approach would be to rely on the device's MAC address - which *should* be globally unique - binding it to user's home domain in a distributed mapping system, which looks similar to the method proposed in [9]. However, as MAC addresses can be easily spoofed, this method could be used only with networks in which users are reasonably trusted, or in which security is not the main concern.

In LISP-ROAM the 802.1x standard is active in all access networks, so that the user will be not only identified with her personal credentials (i.e., `user@domain`), but acharge of serving the access networks (e.g., WiFi access points) with 802.1x authentication; this behavior is already turned on by default in many services that support roaming, such as the pan-european `eduroam` network.

LISP-ROAM requires each ISP to maintain a RADIUS service[2] as a backend for 802.1x, where we store the credentials of the users subscribed to that domain. The RADIUS database is extended with additional attributes, namely the user's EID and the secret key of the home LISP Map-Server, which are needed to update the user's location that is always bound to a server in her home domain.

Although 802.1x networks typically have the access points to directly contact the RADIUS server to authenticate the

[1]The term *identity* does not necessarily mean that the user has to provide her personal credentials, although LISP-ROAM works this way. In general we foresee only the necessity to recognize the user and different criteria (e.g., the MAC address of the connecting host) can be used, if deemed appropriate.

[2]In principle, other solutions such as LDAP or Active Directories may be appropriate as well, provided that the necessary adaptations are made.

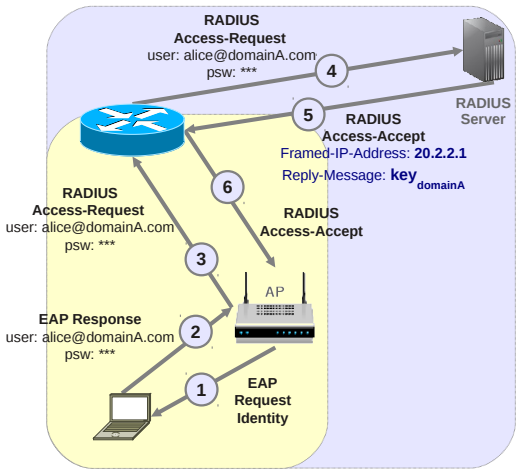

Figure 2: RADIUS Authentication

user, in our case we require the xTR to be in the path between the access device (e.g., access point) and the RADIUS server, as we need to intercept some additional information about the authenticating user that is contained in the `Access-Accept` message. This can be achieved by configuring the access point to refer to the xTR that is in charge of the network when asking for authentication, i.e. each `Access-Request` message is routed through the xTR of the network to the RADIUS Server of the domain, as shown in Fig. 2. If the user provides the correct credentials, the xTR receives an `Access-Accept` message, which will contain also the user's EID and the key of her home Map-Server. This message is then sent to the access point that allows the user to join the network.

3.4 Configuring the network

As depicted in Fig. 2, the RADIUS server writes the EID associated to the connecting user in a RADIUS attribute attached to the `Access-Accept` message. Hence, the xTR automatically learns user's EID as soon the authentication procedure is successful.

For each user, the xTR brings up a new virtual interface with a /30 IP address and configures the address reserved for the default gateway from the /30 pool. The xTR has an *alias* IP address for each interface created, and it will be the default gateway for every user, directly receiving all the traffic generated by them.

The whole operation is transparent to the user device, which will now try to configure its network interface through DHCP. This implies that the xTR has to be the authoritative DHCP server of the network in order to attract DHCP Requests, keeping the DHCP bindings also of the foreign users in the network. Apart from the network interfaces, the DHCP server (embedded in the xTR) needs to be properly configured to return to the user the proper network configuration, which allows the user to connect to the Internet through the new default gateway configured on the xTR.

3.5 Updating the user's location

Once the user's device has been configured with the correct IP parameters, we need to update the proper EID-to-RLOC mapping. For this, the xTR sends a Map-Register message containing the new EID-to-RLOC binding to the home Map-Server, using the appropriate key (written in the `Access-Accept` message returned by the RADIUS server) for authentication. The address of the user's home Map-Server is obtained by querying the LISP Mapping System. This is not needed if the user is in her home network.

The xTR sends a Map-Request, asking for the foreign user's EID, and receives a Map-Reply directly from the Map-Server of the foreign user's domain. In this way, the xTR can read the source IP of the Map-Reply packet to learn the foreign Map-Server's address. In fact, LISP-ROAM assumes that xTRs have registered their EID-prefix(es) using the "proxy Map-Reply" flag (`P-bit`) in the Map-Register message, leaving to each Map-Server the responsibility to answer Map-Requests on its behalf.

A Map-Register message is sent by the xTR to the Map-Server of the foreign user's domain. This message will be authenticated with the correspondent key and will contain the user's EID and xTR's RLOC(s).

At the end of this phase, the EID-to-RLOC is updated and the user can be reached through the current xTR.

LISP-ROAM does not depend on how authentication keys are managed by the domain, although we suggest not to have just one key per Map-Server; instead, it would be better to have a specific key for every user or modifying the Map-Server in order to manage dynamically generated keys.

3.6 Updating established connections

The final processing step consists in two actions: (*i*) notify the previous xTR about the new location of the user, and (*ii*) update the host cache of all the corresponding nodes with the new user location.

With respect to the former, when a Map-Server receives a Map-Register message for a currently existing binding, it sends a Map-Notify message not only to the new xTR, but also to the previous xTR, hence allowing the xTR to detect that a user moved away from its network.

With respect to the latter, in LISP-ROAM the Map-Cache of the xTR is extended to keep also the addresses of the corresponding nodes of the mobile user, hence the previous xTR can also notify them about her new location.

As soon as this message is received by the correspondent nodes, these will automatically trigger a Map-Request to their Map-Resolver and have their mapping updated. In the end, the LISP tunnel between the two hosts (correspondent node and mobile user) will be redirected to the new location and the connection will continue smoothly.

3.7 Overall Architecture

The overall architecture and the required components of LISP-ROAM are shown in Fig. 3. Connectivity is provided by a set of xTRs, each one responsible for one or more EID-prefixes. Each user is assigned a static EID that belongs to the home domain, independently from her current location. The RADIUS service stores users' credentials and other information related to the users. Finally, the LISP Map-Server will always keep the EID-to-RLOC binding of each user, even when she is connected to a foreign network.

4. IMPLEMENTATION

To validate LISP-ROAM in a real scenario, we implemented our solution in a WiFi home gateway based on the OpenWRT [7] operating system. The network components

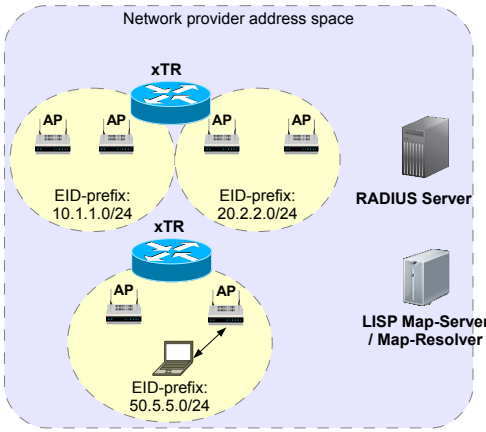

Figure 3: Network provider topology

of the architecture were as standard as possible, keeping modifications needed at minimum, in order to make the solution easy to implement and facilitate future developments.

In our prototype the xTR and the WiFi access point are collapsed in the same machine, meaning that the hosts will connect directly to the xTR. This reduces the steps required for user authentication since the access point directly contacts the RADIUS server.

The LISP support to the OpenWRT operating system was provided through LISPmob [6] software, an open-source implementation of LISP that has been modified to support LISP-ROAM.

4.1 RADIUS extension

Our RADIUS servers are based on the FreeRADIUS implementation, which have been properly configured to authenticate users with the EAP-TLS protocol.

The RADIUS database has been extended with the two attributes presented in Section 3.3. The RADIUS server automatically embeds all the attributes associated to a specific user in the `Access-Accept` message; hence, each time the user is correctly authenticated we are also able to learn her assigned EID and the secret needed for updating the Map-Server.

When a RADIUS server detects a user coming from another domain, it automatically acts as a RADIUS proxy; for instance, if the username contains a domain (`username @ foreigndomain`) that is different from the one of the current ISP, the `Access-Request` message will be relayed to the server belonging to that domain. This means that each RADIUS server must know the address of all the other participating RADIUS Servers, which is already a standard configuration in many confederated domains.

4.2 User data cache

Our xTRs were also enriched with an additional caching mechanism. When the user connects for the first time to a network, she is always considered "unknown" by the xTR, meaning that the xTR does not have any of the data related to her. After all the steps of our protocol are performed, the xTR stores user's data in cache, so when the user connects the next time she will be considered "known" and some of the processing steps can be avoided.

5. EXPERIMENTAL VALIDATION

This section presents the experimental validation of the LISP-ROAM architecture, particularly on characterizing the latency introduced when the user switches between different Wi-Fi networks (*handover* event).

5.1 Test bed architecture

The testbed, depicted in Fig. 4, includes several components as follows. The mobile host is a laptop running an unmodified Linux Kubuntu 12.04. xTR A and B are wireless routers running our modified version of LISPmob on top of OpenWRT Attitude Adjustment 12.09. Instead, xTR C, which serves the correspondent node, runs an unmodified version of the same OpenWRT image plus the vanilla version of LISPmob. The correspondent node is a virtual machine with Ubuntu Linux 12.10, directly connected to xTR C. Finally, our servers (RADIUS, LISP Map-Server and Map-Resolver) are dedicated machines either installed in the university lab or part of the University network (the latter).

The mobile host is about three meters away from xTR A and B, which represent the WiFi access points used to connect to the Internet. The WAN interfaces of all the xTR are connected to the same network and their public IPs (RLOCs) are part of the same pool.

The procedure used to validate our approach includes the following steps:

1. User (`alice@domainb.com`) connects to her home Wi-Fi network (**LISP-B**), served by xTR B;
2. She is assigned her EID 10.1.2.121 and the /30 netmask through the local DHCP service
3. User starts a data connection with the correspondent node (10.1.3.165), which is served by xTR C
4. At a certain point, the user decides to move to the foreign Wi-Fi network (**LISP-A**), under xTR A, obtaining a new IP configuration through the local DHCP service
5. The user moves back and forth from her home/foreign network in order to test all the possible connections scenarios, as shown in Table 1.

Having verified that LISP-ROAM works in all those conditions, our next measurements aim at evaluating the duration of the "off" period, i.e., the time in which data sent by the correspondent node cannot be delivered to the mobile host node because the network reconfiguration is still in progress.

5.2 Handover latency

We consider as start of the handover the instant in which the user arbitrarily decides to change Wi-Fi network (e.g., moving from the home to the foreign domain or vice versa). The handover will be considered completed when the connections are resumed, i.e., the data restart flowing in the network.

Results mainly depend (*i*) on the type of the network the user is connecting to (home/foreign) and (*ii*) whether the user is known/unknown to the xTR. In fact, some steps of

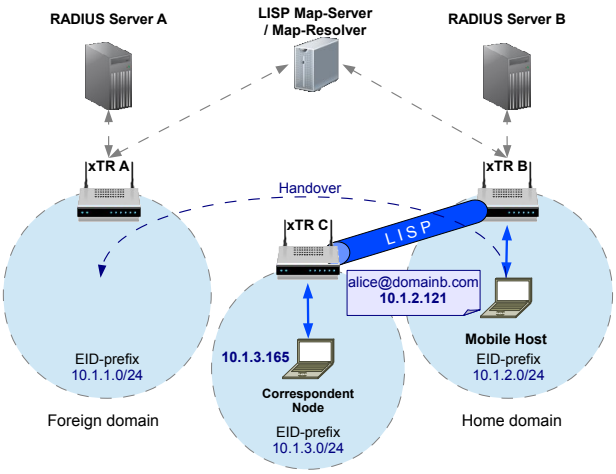

Figure 4: Test bed architecture

the procedure are skipped in case the user is known to the xTR, e.g. the address of the home Map-Server is not needed and the DHCP server already knows the binding associated to the user's EID, as detailed in Table 1.

Fig. 5 shows the latency introduced by the handover, detailing the results of 15 experiments in which we focused on two scenarios, the "best case" (i.e., known user connecting to a home network) and the "worst case" (i.e., unknown user connecting to a foreign network), whose results are shown by the two colors of each bar. As it can be seen, in the worst case scenario the average latency value is on the order of 5 seconds - while is lowered to 3 seconds in the best case scenario. In order to investigate the most prominent components of the latency, we sliced the time needed for the handover according to the following equation:

$$T_{handover} = T_{EAP} + T_{DHCP} + T_{loc} + T_{CN}$$

where:

1. T_{EAP} represents the time elapsed between the EAP `Identity Request` and the final EAP `Success` message, both sent by the xTR. This range includes the RADIUS communication between the xTR and the RADIUS server.
2. T_{DHCP} represents the time elapsed between the DHCP `Discover`, sent by the mobile host, and the DHCP `ACK` message, sent by the xTR.
3. T_{loc} is the time needed for retrieving the address of the home Map-Server of the user (if needed) and registering her new location
4. T_{CN} is the sum of the time elapsed between the user's location update and the notification sent to the previous xTR plus the update of the correspondent node's Map-Cache.

Table 2 represents the average value of each time slice, calculated on the same base of the experiments of Fig. 5. In this case the value for $T_{handover}$ are slightly lower than the ones depicted in Fig. 5 because we omitted the time needed

Table 1: Possible connection scenarios

	Type of user	
	Unknown	**Known**
Foreign	All steps performed	No Map-Server retrieval No DHCP update
Home	No Map-Server retrieval	No Map-Server retrieval No DHCP update

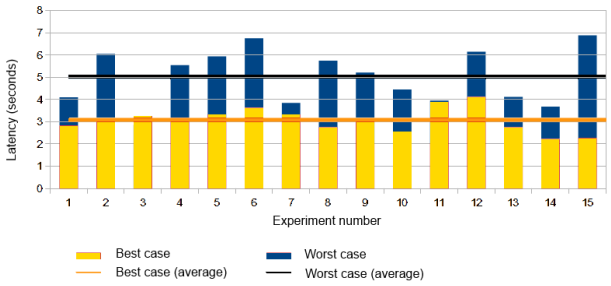

Figure 5: Latency measurement

for physically attaching the host to the network. This delay can not be measured looking at the packet captures and strongly depends on the hardware characteristics of the machines (host and access point).

5.3 Handover throughput

While the latency is important to characterize our approach, perhaps the most important parameter for end users is how the throughput of the (TCP) active network connections is impaired by the handover. For this, we started a TCP data transfer that consumed an average bandwidth of ~65000Kbps from our mobile host and we measured the impact on the overall throughput with transfers of different duration. We repeated 15 experiments, using both the best and worst-case scenarios.

Table 3 shows how the throughput is significantly decreased when short transfers are involved while the difference becomes negligible smaller for sessions around 2 minutes. Fig. 6 clearly depicts the trend of the throughput for a TCP session, which is reset to zero during the handover occurred in the worst case scenario.

6. RELATED WORK

At the time of writing, many solutions for host mobility have been proposed: Mobile IP[10] is the most notable one, PMIPv6 [9] that is the network-based version of MIP and SHIMv6 [5]. These technologies strongly rely on IPv6 and have been adopted mostly in mobile networks.

In recent times, LISP has been involved in the design of mobility solutions. LISP-MN [2] is a host-based implemen-

Table 2: Latency breakdown

T slice	Worst case scenario	Best case scenario
T_{EAP}	2.78 s	1.33 s
T_{DHCP}	1.25 s	0.03 s
T_{loc}	0.11 s	0.01 s
T_{CN}	0.52 s	0.52 s
$T_{handover}$	4.66 s	1.89 s

Table 3: Throughput breakdown

Data transfer duration	Worst case	Best case
10 s	22442 Kbps	27962 Kbps
20 s	42155 Kbps	43149 Kbps
60 s	55726 Kbps	57115 Kbps
120 s	60529 Kbps	60797 Kbps

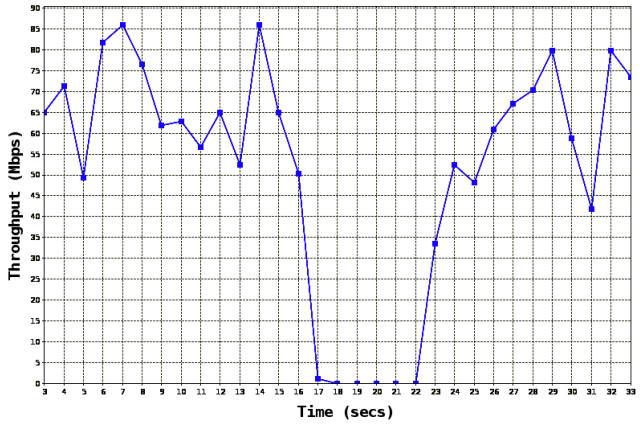

Figure 6: Throughput of a TCP session

tation that allows a mobile device to seamlessly keep its connections alive while roaming through different networks.

LISP VM Mobility [1] instead represents a network-based approach, and it presents a solution for moving virtual machines between geographically distant sites without dropping active connections. The migration is done keeping the virtual machine unaware, forcing the network components to detect the new host in the network by checking the generated traffic.

Gohar et al. [4] propose a solution (which bases on an earlier proposal in [8]) assuming that the mobile host moves within a domain under the control of the same network operator, hence it is able to receive always the *same* IP address. In fact, in that case we can assume the presence of a provider-wide IP address allocation service that returns always the same address to that client, e.g., based on the SIM card or the MAC address. Under those conditions, the edge router can dynamically activate a LISP tunnel on behalf of the user, hence avoiding any additional software/configuration on the user terminal.

We assumed that the user terminal can move across multiple Wi-Fi networks that belong to different domains. However, LISP-ROAM abstracts from the infrastructure / topology / configuration of the networks, in order to allow a future extension including also mobile operators. The scenario of this work is highly different from the one of [1], since we are considering a host that is actively connecting to a new network, performing all the steps needed (e.g., authentication, IP assignment, etc.).

7. CONCLUSIONS

The solution proposed in this paper achieves user mobility through Wi-Fi networks, and could be scaled to bigger scenarios. However, some choices made in the proposal can lead to future discussions and improvements.

The /30 virtual network interface approach is easy to implement but it allows to use only one address out of four, hence decreasing sensibly the overall amount of usable EIDs. We assume that this mechanism is fair to be used for prototyping, but other techniques should be considered for further developments (e.g., Proxy-ARP). Furthermore, having fixed EIDs puts a big constraint in our proposal, since the user cannot use more than one device at once (with the same credentials), and it forbids the re-usability of the EIDs.

The principal flaw of the proposal is the latency introduced during the handover. Part of this latency is due to the time needed to physically attach to the new network and obtain an IP address, which is the common scenario, and cannot be decreased. The additional delay added by this solution, instead, is because of the actions needed to identify and authenticate the user while roaming, to update her location and to notify her peers of the move. Overall, the authentication part can be considered the real bottleneck of the solution, since it is the one that actually requires most of the time, and has to be repeated for every connection.

Further works can be addressed to optimize or find alternatives to the 802.1x authentication. Also, future developments should involve 3G/4G networks in the scenario to design a joined architecture being able to keep connections while roaming on different network supports. Since the proposal keeps a high level of abstraction, it can be applied to mobile scenarios using, for instance, a SIM-based authentication mechanism.

Acknowledgments

This work has been partially supported by the Spanish Ministry of Education under scholarship FPU2012/01137.

8. REFERENCES

[1] Cisco Systems. “Locator/ID Separation Protocol (LISP) Virtual Machine Mobility Solution”. white paper, 2011.

[2] D. Meyer D. Farinacci and C. White. *“LISP Mobile Node”, draft-meyer-lisp-mn-09*, 2013.

[3] D. Meyer D. Farinacci, V. Fuller and D Lewis. *“The Locator/ID Separation Protocol (LISP)”, RFC 6830*, 2013.

[4] M. Gohar and S. J. Koh. “Network-Based Distributed Mobility Control in Localized Mobile LISP Networks”. *IEEE Communications Letters*, 16(1):104–107, 2012.

[5] E. Nordmark and M. Bagnulo. *“Shim6: Level 3 Multihoming Shim Protocol for IPv6”, RFC 5533*, 2009.

[6] The LISPmob open-source project (online). http://lispmob.org.

[7] OpenWRT. OpenWRT documentation: http://wiki.openwrt.org.

[8] S. T. Kim S. I. Choi and S. J. Koh. *“Network-based Mobility Control in LISP Networks”, draft-sichoi-lisp-ar-00.txt, work in progress*, 2013.

[9] I. Soto, C. J. Bernardos, and M. Calderon. “PMIPv6: A Network-Based Localized Mobility Management Solution”. *The Internet Protocol Journal*, 13(3):1–32, 2010.

[10] W. Stallings. “Mobile IP”. *The Internet Protocol Journal*, 4(2):2–14, 2001.