LISPmob: Mobile Networking through LISP

Albert Cabellos, Alberto Rodríguez Natal, Loránd Jakab, Vina Ermagan,
Preethi Natarajan and Fabio Maino

{acabello,arnatal,ljakab}@ac.upc.edu

{vermagan, prenatar, fmaino}@cisco.com

1.- Introduction

The current Internet architecture was not designed to easily accommodate mobility because IP addresses are used both to identify and locate the hosts. The fact that separating identity from routing location is an important design principle of inter-domain networks was known even before the Internet was created [1], but unfortunately the current architecture does not implement it. Such separation would seamlessly provide mobility and multihoming, among other desirable features, to the Internet. As a result, this is still an important research topic, and many solutions centered around this idea have been proposed [2,3,4,5,6]. Besides this, mobility also requires a location management system that maintains bindings between the identity and the location of the mobile client.

The Location/ID Separation Protocol [1] has been recently proposed to provide an incrementally deployable solution to such separation in the Internet. LISP considers two different types of addresses: Endpoint Identifiers (EIDs) and Routing Locators (RLOCs). EIDs identify hosts, and are assigned independently of the network topology while RLOCs identify network attachment points, and are used for routing. This allows that EID remain unchanged even if a topological change occurs, such as a handover. In transit, packets are encapsulated; the outer header contains RLOCs while the inner EIDs. LISP also introduces a Mapping System (MS) [2], a distributed database that maps EIDs to RLOCs.

LISP, as architecture, provides two important features to the Internet. First it truly splits location from identity, which is a requirement to provide native mobility and multihoming. With LISP, mobile clients can be seamlessly equipped with multiple wireless interfaces, and handover from different points of attachment, or among interfaces. Secondly, it provides a new level of indirection. A hostname lookup in DNS returns an EID, a second lookup is required to the Mapping System to find the associated RLOC. With LISP, this MS acts as a location management system. But unlike in traditional mobility protocols, such as Mobile IP [4], LISP's MS is distributed and federated. Mobile IP´s location management system (the Home Agent) is deployed at the mobile client's service provider. Interestingly, LISP´s MS avoids mobile service provider lock-in.

LISP is, at the time of this writing, being standardized at the IETF. Within the LISP architecture [1], LISP-MN [3] specifies the mobility functionality, and LISPmob (http://lispmob.org) an open-source implementation. In this white paper we describe the LISP-MN architecture and we discuss how it works and we review the advantages of LISP-MN in front of other mobility protocols.

2.- How LISP-MN works

2.1.- Overview

The Locator/ID Separation Protocol (LISP) [1] specifies an architecture for decoupling host identity from its location information in the current address scheme. This separation is achieved by replacing the addresses currently used in the Internet with two separate name spaces: Endpoint Identifiers (EIDs), and Routing Locators (RLOCs). Host applications bind to host's EID, which is used as the address for transport connections. RLOCs are IPv4 or IPv6 addresses used for routing through transit networks. In order to reach a host, identified by its EID, one must first find the current location (RLOC) of the host. LISP introduces a publicly accessible Mapping System that is designed to serve the EID-to-RLOC mapping information. Once the RLOC associated to an EID is discovered, packets with headers from the EID namespace are encapsulated in a second header from the RLOC space, and are routed to the destination, where the LISP header is removed before delivering packet to the destination host. LISP introduces special gateway routers called Tunnel Routers that perform the LISP encapsulation, and decapsulation at each site's ingress and egress points.

Separating the host identity (EID) from its locator (RLOC) enables seamless endpoint mobility by allowing the applications to bind to a permanent address, the host's EID. The location of the host can

change many times during an ongoing connection. Each time, the LISP tunnel routers will encapsulate the packets to the new RLOC, preserving the connection session from breaking. LISP-MN [3] leverages this feature to build a mobility architecture and protocol based on LISP. In LISP-MN a mobile node (MN) is typically statically provisioned with an EID that it uses for all its connections. Each mobile node essentially behaves as a LISP site in accordance to the LISP architecture. Packets (except for management protocols such as DHCP) are LISP encapsulated by the mobile node, and routed based on the RLOCs to the destination site. In the event of a handover, MN receives a new RLOC and updates its EID-to-RLOC mapping in the associated mapping system to maintain reachability at its new location.

The LISP-MN architecture leverages four existing LISP components: (i) A Mapping System, (ii) the LISP-MN, (iii) LISP Internetworking components (iv) LISP NAT-traversal
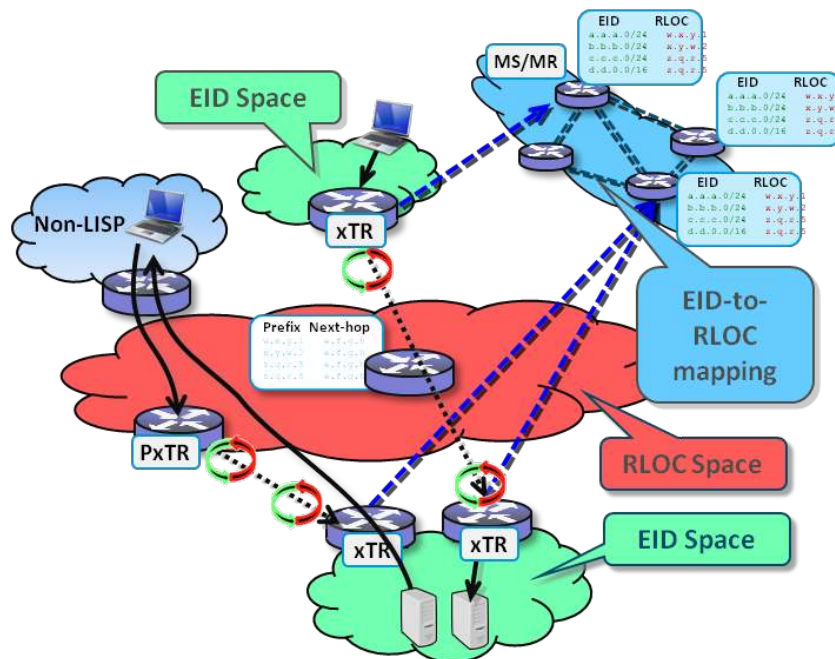

Figure 1.- An overview of the LISP architecture

2.2.- The LISP Mapping System

The LISP Mapping System [2] (figure 1) is a central aspect of the LISP-MN architecture and it is a publicly accessible service that publishes location information associated with EIDs (EID-to-RLOC mappings). Main elements of LISP mapping system are Map Servers and Map Resolvers. These are on the boarder of, and the interface to the mapping system. EID-to-RLOC mappings are stored in Map Servers, and each Map Server is associated with a partition of the EID name space, and stores the location information for those EID prefixes. Therefore, each LISP mobile node is associated with a specific Map Server where it registers its EID-to-RLOC mapping, and updates it according to its movement. In this context, Map Servers have assigned a set of prefixes (EIDs) and delegate them to either LISP tunnel routers or mobile nodes.

Map Resolvers are used as an interface to the mapping system for looking up EID location information; they have a similar functionality as DNS resolvers have in today's Internet: LISP mobile nodes send EID lookup requests (Map Request) to the mapping system through Map Resolvers.

LISP mapping system is designed to route EID lookup request from a Map Resolver to the Map Server responsible for that EID. At the time of this writing LISP has deployed a mapping system based on a BGP at the LISP Pilot Network [5]. The system is an overlay that connects Map Servers and Map Resolvers. In this overlay, Map Servers announce the EID prefixes that they serve through BGP. Map-Resolvers send EID requests onto the overlay, where they are routed according to the requested EID, and eventually reach the associated Map Server that stores the EID's location information.

2.3.- The LISP-MN

The lightweight tunnel router is the implementation of LISP-MN on the endpoint or mobile node. Mobile node tunnel routers are used to encapsulate outgoing packets in a LISP header based on RLOCs before leaving the mobile node, and to remove the LISP header from incoming packets before sending them to

upper layers ultimately reaching the destination application. The LISP-MN protocol is then, best understood, as the concatenation of three different phases: (i) *Registering EID and obtaining an RLOC* (ii) *Signaling EID-to-RLOC bindings and transmitting data-packets* (iii) *Handover*. In the following sections we review these phases in detail.

a)   Registering EID and obtaining an RLOC

Each LISP-MN is configured with at least one EID. As we have discussed before an EID is either a normal (/32) IPv4 or (/128) IPv6 address, identifies the node uniquely and remains static independently of its location. If the node has also a DNS entry, this entry returns the EID. This address is typically assigned by the Map-Server provider and in LISPmob, it is configured in a static file.

In order to connect to the Internet, the LISP-MN also needs at least an RLOC. RLOCs are obtained by traditional mechanisms such as DHCP or configured manually and are location dependent. This means that as the mobile node roams across providers, it will obtain a different RLOC in each location.

For each new RLOC obtained by the LISP-MN, the node has to inform about the new EID-to-RLOC binding to its Map-Server. In order to do so LISP defines the Map-Register signaling message that includes the EID and the RLOC. The node may include multiple RLOCs if the node is multihomed and LISP supports any combination of IPv4 and IPv6 for EIDs and RLOCs. The LISP-MN and the Map-Server share a pre-configured key (again configured in a static file in LISPmob), this key is used to sign the Map-Register to ensure authentication. Once the Map-Server receives a valid Map-Register containing an EID-to-RLOC mapping it will it make it accessible throughout the Mapping-System.
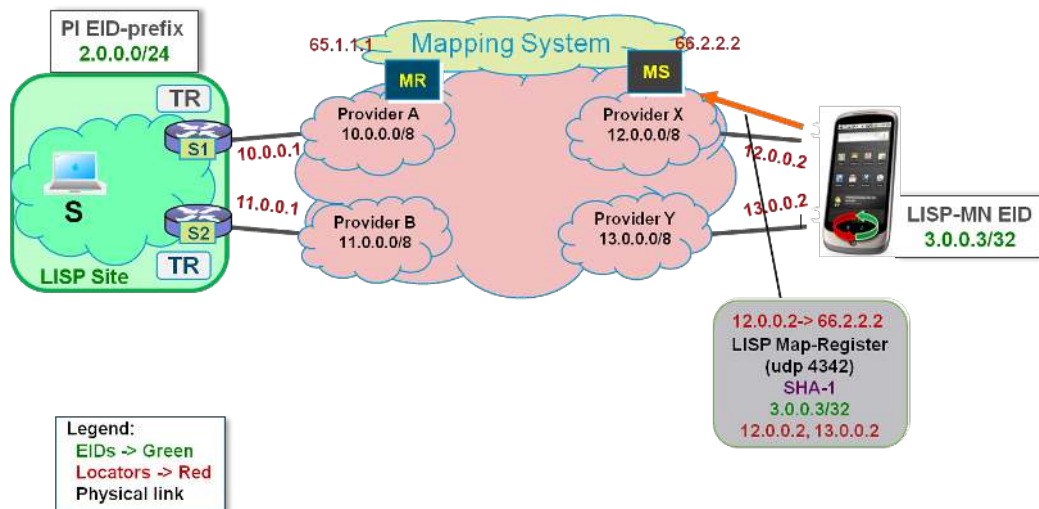

Figure 2.- Registering an EID-to-RLOC bindings

Figure 2 shows a LISP-MN configured with the EID 3.0.0.3/32 and two RLOCs from two different providers (X and Y): 12.0.0.2 and 13.0.0.2. The MN registers these two bindings 3.0.0.3/32→12.0.0.2 and 3.0.0.3/32→13.0.0.2 into its Map-Server (identified with the address 66.2.2.2). The MN and a MS have a pre-configured key and this Map-Register message is signed and hence, authenticated.

b)   Signaling EID-to-RLOC bindings and transmitting data-packets

Once a valid EID-to-RLOC has been registered into the Map-Server the mobile node can start sending and receiving data packets. In this section we describe how a static node located in a LISP site establishes a connection with a mobile node. In section 2.4 we also present the case when the peering site is non-LISP.

As figure 3 shows, the static node first retrieves the EID of the node (3.0.0.3/32) by querying the DNS and then transmits a packet addressed to this EID just as in the plain Internet. The packet is routed until it reaches the LISP tunnel router. Upon reception, the tunnel router checks whether it knows the EID-to-RLOC binding or not. For this purpose each LISP node includes a data-structure called Map-cache which stores such information.
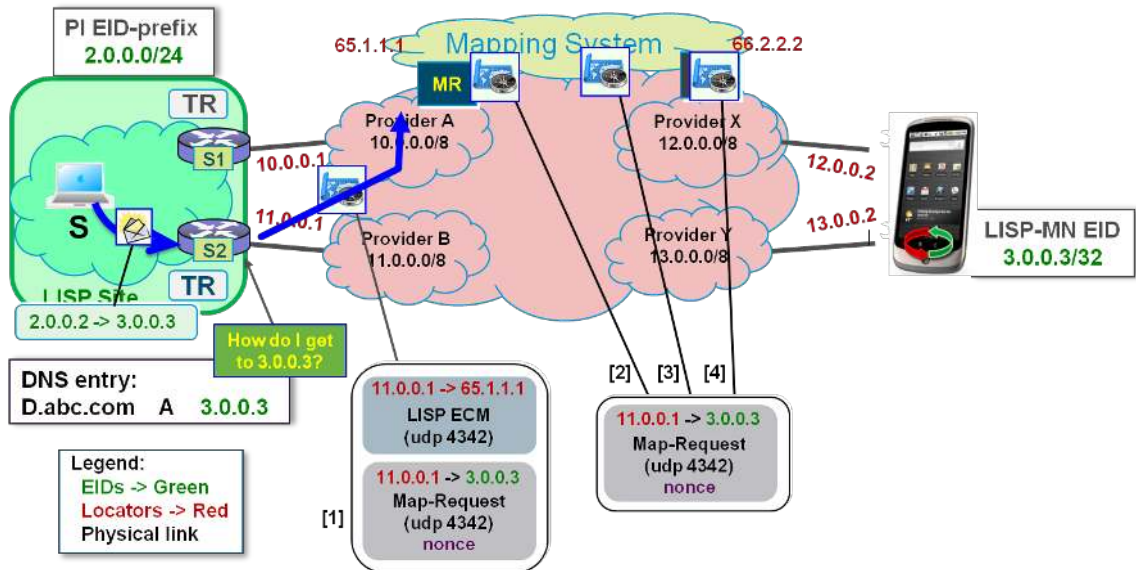
Figure 3.- Signaling an EID-to-RLOC binding (Map-request)

If the tunnel router´s Map-cache does not contain the binding for the destination EID, it will trigger a Map-request message. This message is used to query the Mapping-system for a particular binding. The Map-request is sent to the Map-resolver, which is typically co-located with the Map-server. In turn, the Map-Resolver forwards the Map-Request message through the Mapping-system that routes it, according to the destination EID, until it reaches the Map-Server that provides mapping services to the MN.
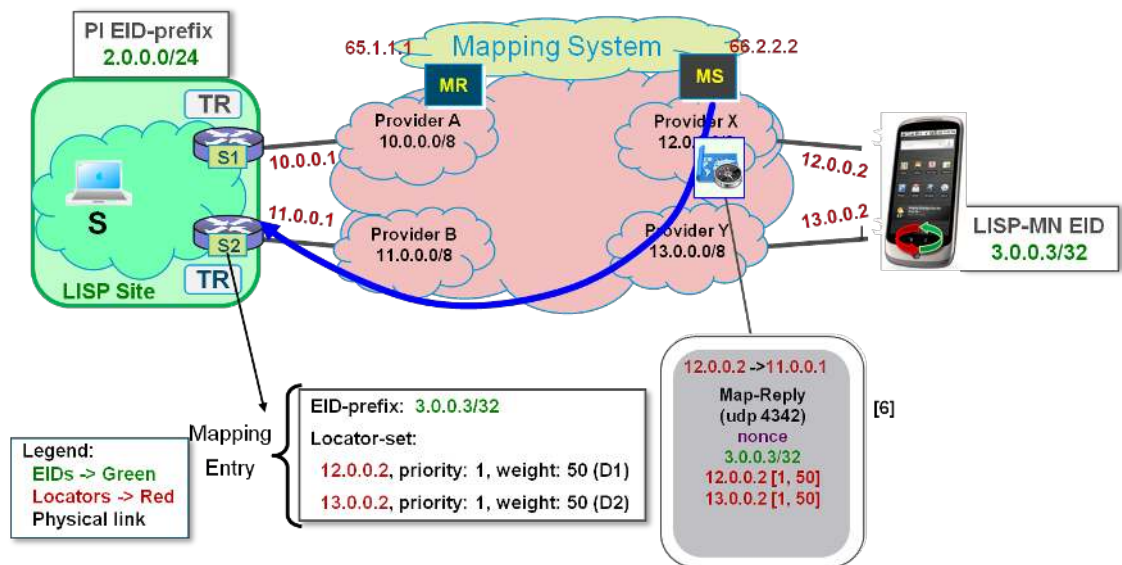
Figure 4.- Signaling an EID-to-RLOC binding (Map-reply)

The Map-Server then constructs a reply for the Map-request using another LISP defined message, the Map-reply. This message mainly contains the EID of the MN (3.0.0.0/32), the set of RLOCs that provide connectivity to the MN (12.0.0.2 and 13.0.0.2) and the priorities and weights of each locator, this is used for ingress load-balancing. Finally the Map-reply also contains a Time-To-Live (TTL) that defines the amount of time for which this particular mapping is valid, and a nonce to avoid unsolicited Map-replies.

The Map-reply is sent directly to the tunnel router that will install this binding in its Map-cache and will use it to encapsulate packets towards the MN until the TTL expires. At this point it will request a fresh binding.

Typically, as in TCP, the node will reply with another data-packet addressed to the EID of the node (S). Since the Map-cache of the MN does not have the binding for such EID it will trigger a Map-Request querying for the RLOC of the LISP site (10.0.0.1 and 11.0.0.1). This Map-Request, as in the previous case, will be routed through the Mapping-System towards the Map-Server servicing the site that in turn, replies with a Map-Reply. The mobile node, upon reception of this message will install this binding in its

Map-Cache and will start to encapsulate data packets directly to the locators of the LISP site according to whatever policies have been defined (weight and priorities). Such data-packets are decapsulated at the tunnel router and forwarded, as in the plain Internet, to the final destination S.
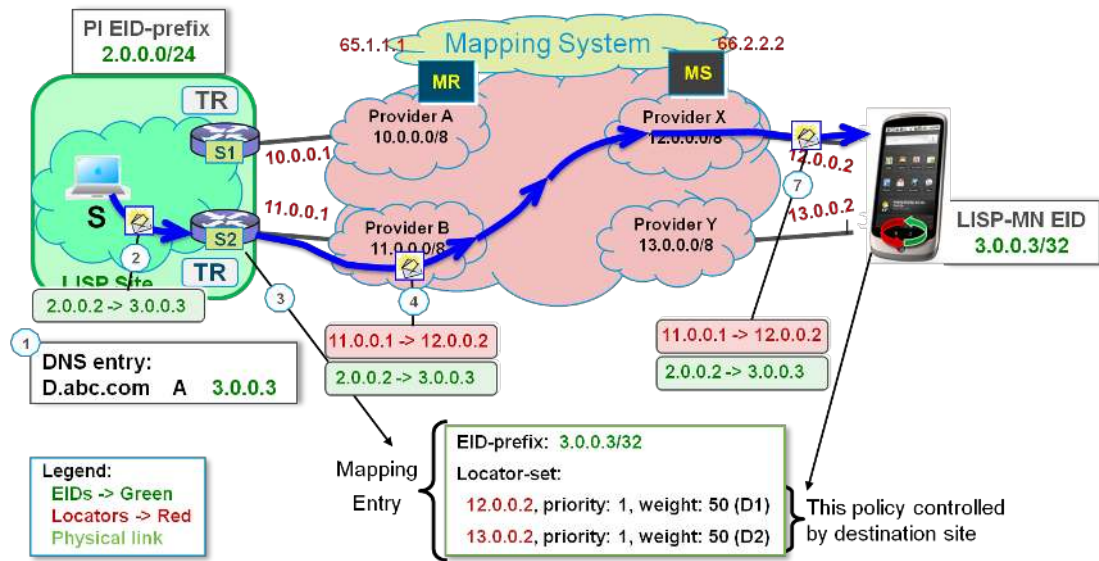

Figure 5.- Encapsulating and decapsulating data-packets

c)   Handover

When the mobile node changes its point of attachment it will regain connectivity in a new subnetwork, possibly serviced by a new provider. In this case it will first obtain a new RLOC and, as described in section 2.3.a, will send the new EID-to-RLOC binding on its Map-Server.

In order to resume existing connections the MN has to update all the EID-to-RLOC bindings stored in the Map-Cache of the routers with which it is communicating. To do so the MN will send a special signaling message called Solicit-Map-Request (SMR) to all these routers/mobile nodes. Upon reception of such message the peering router/mobile node will trigger a Map-Request addressed towards the EID of the soliciting MN. This message will be in turn forwarded by the Mapping-System until it reaches the Map-Server servicing the MN that will reply with a Map-Reply containing the updated RLOC(s).
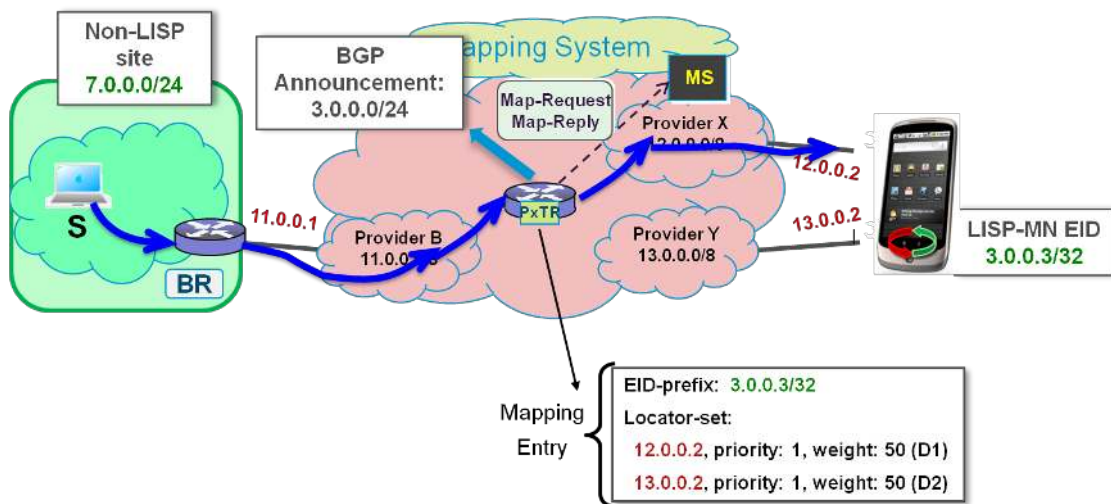

Figure 6.- Inter-networking with non-LISP sites

2.4.- The LISP Inter-networking components

The LISP internetworking components are proxies that facilitate communication with legacy nodes that are not LISP enabled. Mainly, the PxTR is a LISP-enabled router that attracts traffic by announcing, by means of BGP, EID prefixes. Then it queries the Mapping-system to obtain the corresponding RLOC bindings and encapsulate the data-packets towards them. Additionally, PxTRs also decapsulate packets

sent by LISP-enabled sites and nodes towards the non-LISP Internet. This is done to avoid ingress-filtering issues.

As shown in figure 6, the PxTR is announcing at the BGP DFZ an aggregated EID prefix that covers the one configured at the MN. In this scenario we are assuming that this entire prefix is used either by LISP sites or LISP-MNs. By means of this BGP announcement, PxTRs attracts traffic addressed to the EID of the MN (3.0.0.3/32) and, upon reception of a data-packet it queries the Mapping-system to obtain the locator set (12.0.0.2 and 13.0.0.2) and proceeds to encapsulate packets. In turn, the PxTR is also used to decapsulate data-packets addressed to non-LISP sites.

2.5.- LISP NAT Traversal

LISP control and data messages are UDP encapsulated and they use the destination port 4342 and 4341 respectively. However, the source port for these packets is random and therefore, and therefore NAT-boxes do not allow incoming packets. To avoid this issue a new network element has to be deployed: NTR (NAT Traversal Router). This entity acts as a proxy between the NAT box and the MN´s incoming packets. The operations of a MN behind a NAT are as follows.

First the MN must check weather is behind a NAT box. To obtain this information the MN sends a special signaling packet (*NAT Info Request*) to its Map-Server. In turn, the Map-Server replies with a list of available NTRs and the actual source address and port of the MN. With this, MNs are aware that they are (or not) behind a NAT box. In this case, the MN sends a Data Map Register message to the NTR with source port 4341. This message is an encapsulated Map-Register which opens the port 4341 in the NAT table and it is used by the NTR to infer the translated RLOC and port of the MN. Then, the NTR forwards the Map-Register to the appropriate Map-Server. Please note that this Map-Register contains the RLOC of the NTR, not the one of the MN, so all MN´s incoming packets are received by the NTR which, in turns, forwards them to the the translated RLOC and port of the MN. This way the NTR acts as a signaling and data-proxy for the MN.
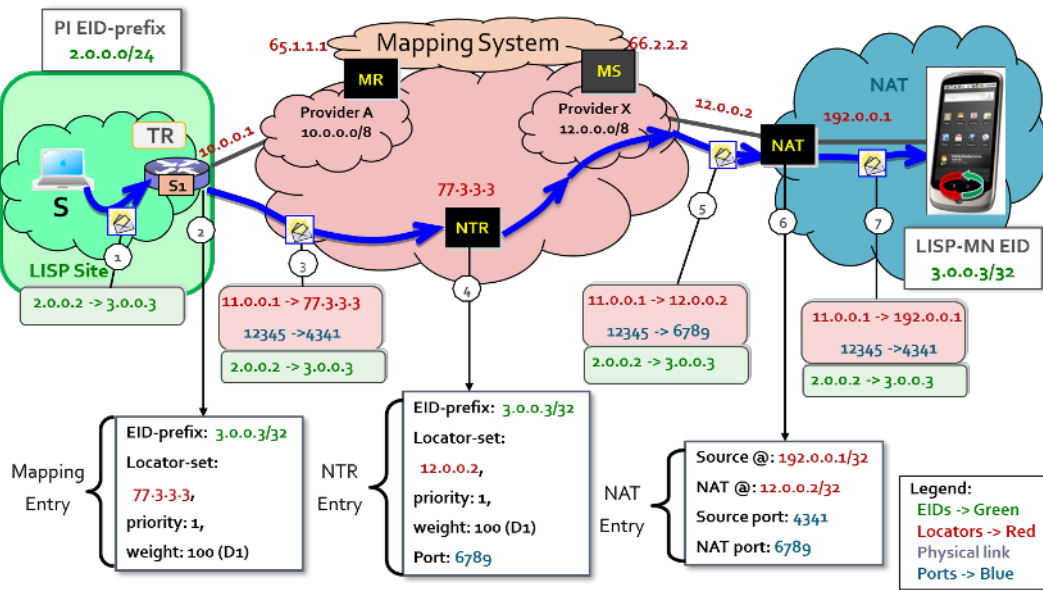

Figure 7.- MN behind a NAT box

Figure 7 shows an example of the packet flow when the MN is behind a NAT. The NTR register its RLOC (77.3.3.3) as the RLOC for the MN's EID (3.0.0.3). In its NAT table it stores the port (6789) and IP (12.0.0.2), since this is the real address and port received in the Data Map Register.

3.- Analysis of LISP-MN

Advancement of wireless access technologies has led to tremendous increase in the adoption of mobile devices and demand for mobile internet. Consequently, a plethora of solutions and protocols have been proposed in the past decade (see [6] and the references therein), with some standing out more than others such as Mobile IP [4]. In spite of the current diversity in mobility protocols and solutions, LISP-MN has a set of technical and strategical advantages with respect to existing solutions that make it an appealing alternative.

- LISP is designed for separation of host identity from location or routing address. This separation enables simplified multihoming. In LISP multihoming is even more native to the architecture: each EID prefix can be mapped to more than one RLOC, and each RLOC can be assigned specific priority and weight. LISP-MN inherits these benefits from the LISP architecture.

- LISP-MN separates the control plane functionality from the data plane, allowing each to scale independently. Since LISP-MN does not require Home Agent or Foreign Agent network elements in the data plane, it avoids triangle routing at the data plane level, for both IPv4 and IPv6 address families. Data packets always follow the shortest path and hence, LISP-MN incorporates natively route optimization support.

- The Mobile IP family of protocols is indeed a set of protocols that provides basic and advanced functionality to mobile nodes. For instance there are two separate versions for IPv4 and IPv6, and additional protocols for network (prefix) mobility. Furthermore, advanced features such as fast handovers [7] or multihoming [8] are again defined in separate documents. As a result, a developer willing to adopt Mobile IP must read and implement different protocols defined in several documents. In LISP-MN all the features are defined in a single document, and thus only require a single implementation. Currently LISP-MN supports IPv4, IPv6, network mobility and multihoming natively.

- Separation of control plane from data plane in LISP-MN, facilitates the decoupling of end-point identity from the mobility service provider. The sole functionality of the control plane is to locate a mobile node, much like DNS locating a service or a host name today. Similar to DNS, LISP control plane has a distributed and federated mapping system. The cross domain and distributed nature of the LISP mapping system renders LISP-MN as a more transparent and open solution compared to existing alternatives. In LISP communication at the data plane does not depend on a specific mobility service provider. Mobility becomes a native feature of the network architecture: one step closer to making internet for mobile nodes as seamless and open as the internet is today for static hosts.

References

[1] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "Locator/ID Separation Protocol (LISP)," draft-ietf-lisp-16, Internet Engineering Task Force, Nov. 2012, work in progress.
[2] V. Fuller, D. Farinacci, D. Meyer, D. Lewis, "LISP Alternate Topology (LISP+ALT)" draft-ietf-lisp-alt-05, Internet Engineering Task Force, Oct. 2010, work in progress.
[3] D. Meyer, D. Lewis, D. Farinacci "LISP Mobile Node", draft-meyer-lisp-mn-06.txt, Internet Engineering Task Force, Oct 2011, work in progress.
[4] C. Perkins, "IP Mobility Support" RFC 3344, 2002
[5] (online) www.lisp4.net
[6] W. M. Eddy, "At what layer does mobility belong?" in IEEE Communications Magazine, Number 42, Vol 10, 2004.
[7] R. Koodli, "Fast Handovers for Mobile IPv6 (FMIPv6)" RFC 4068, 2005
[8] G. Tsirtsis et al. "Flow Bindings in Mobile IPv6 and Network Mobility (NEMO) Basic Support", RFC 5648, 2011

Terminology

Endpoint ID (EID):   This is the address that a LISP MN uses as identity, and thus for transport connections.

LISP Mobile Node (LISP-MN): A LISP capable fast roaming mobile handset.

Mapping-System (MS): A publicly accessible distributed database that stores EID-to-RLOC bindings.

Map-cache: A data structure which contains an EID-prefix, its associated RLOCs, and the associated policy. Map-caches are typically found in xTRs and LISP-MNs.

Map-register: A LISP signaling message used by a LISP-MN to register its EID-to-RLOC binding in a Map-Server.

Map-reply: A LISP signaling message that conveys an EID-to-RLOC binding.

Map-request: A LISP signaling message used to query for a particular EID-to-RLOC binding.

Map-Server: A network infrastructure component which learns of EID-prefix mapping entries and publishes them in the Mapping-system

Map-Resolver: A network infrastructure component which accepts Map-Requests and finds the appropriate EID-to-RLOC mapping by consulting the Mapping-system..

Proxy Tunnel Router (PxTR): PxTRs are used to provide interconnectivity between sites that use LISP EIDs and those that do not. They act as a gateway between the Legacy Internet and the LISP enabled Network

Routing Locator (RLOC): This is the address that a LISP-MN uses as location, and it is a routable address that can be used to reach a mobile node. A mobile node may be multihomed and have multiple RLOCs.

Tunnel Router (TR): A TR is a LISP-enabled router that has at least one RLOC configured, and encapsulates and decapsulates data-packets.