A STUDY OF LOCATOR ID SEPARATION PROTOCOL

A THESIS IN
Electrical Engineering

Presented to the Faculty of the University
of Missouri-Kansas-City in partial fulfillment of
the requirements for the degree

MASTER OF SCIENCE

by
VARUN JAIN

B.E., Mumbai University, 2006

Kansas-City, Missouri
2010

A STUDY OF LOCATOR ID SEPARATION PROTOCOL

Varun Jain, Candidate for the Master of Science Degree

University of Missouri-Kansas City, 2009

ABSTRACT

Locator ID/Separation Protocol (LISP) aims at solving the issues in the current Internet Routing Architecture. The growth of the BGP routing table and Forwarding Information bases on core routers is very high. In addition, the number of BGP messages that are currently being processed by the BGP routers is a worrisome issue. Locator/ID Separation Protocol (LISP) is a recently proposed approach that provides a solution to these problems. By employing LISP, it is anticipated that significant scaling benefits can be achieved among which are the reduction of routing table sizes, traffic engineering capabilities, mobility without address changing. We present an analysis of how much these improvements are. Furthermore, a detailed study of this protocol is carried out and is compared against other solutions that are proposed along with an analysis of LISP as well.

The faculty listed below, appointed by the Dean of the School of Computing and Engineering, have examined a thesis titles "A Study of Locator/ID Separation Protocol", presented by Varun Jain, candidate for the Master of Science degree, and hereby certify that in their opinion it is worthy of acceptance.

Supervisory Committee

Dr. Deep Medhi
Department of Computer Science and Electrical Engineering

Dr. Cory Beard
Department of Computer Science and Electrical Engineering

Dr. Kenneth Mitchell
Department of Computer Science and Electrical Engineering

CONTENTS

ILLUSTRATIONS

Figure Page

LIST OF TABLES

Table Page

GLOSSARY

CIDR:	Classless Interdomain Routing
BGP:	Border Gateway Protocol
FIB:	Forwarding Information Base
IANA:	Internet Assigned Numbers Authority
PA:	Provider Aggregatable
PI:	Provider Independent
LISP:	Locator ID Separation Protocol
RLOC:	Routing Locator
EID:	Endpoint ID
ITR:	Ingress Tunnel Router
ETR:	Egress Tunnel Router
AFI:	Address Family Indicator
MTU:	Maximum Transmission Unit
IHL:	Inner Header Length
UDP:	User Datagram Protocol
NAT:	Network Address Translation
TTL:	Time To Live
TOS:	Type Of Service
ICMP:	Internet Control Message Protocol
SMR:	Solicit Map Request
IPv4:	Internet Protocol version 4
IPv6:	Internet Protocol version 6
IGP:	Interior Gateway Protocol

SCTP:	Stream Control Transmission Protocol
DFZ:	Default Free Zone
PTR:	Proxy Tunnel Router
DNS:	Domain Name System
NERD:	Not-so-novel EID to RLOC database
HTTP:	HyperText Transfer Protocol

ACKNOWLEDGMENTS

I would like to take this opportunity to express my heartfelt gratitude to my thesis advisor Dr. Deep Medhi. I thank him for his most valuable guidance and encouragement. I appreciate his generosity and valuable time that he spared for me to discuss and clarify the issues that came up during the course of this work.

I am very thankful to Dr. Cory Beard and Dr. Kenneth Mitchell for serving as members of my thesis committee.

I would like to offer my gratitude to my parents Mr. Sudhir Jain and Mrs. Rajni Jain, and also to my brother Mr. Vaibhav Jain for their love encouragement and support.

Finally, I would like to thank all my teachers and professors in my school and college and all my friends who have always been there for me.

CHAPTER 1

INTRODUCTION

1.1 Introduction

IP addresses consists of two parts: the *network address* (which identifies a whole network or subnet), and the *host address* (which identifies a particular machine's connection or interface to that network). The address space was initially divided into different classes based on the bit boundary for the network address part for unicast routing; for example Class A specified the network address based on the leftmost 8 bits, Class B based on the leftmost 16 bits, and Class C based on the leftmost 24 bits. This division is used in traffic routing in and among IP networks using implicit address blocks. When the Internet started growing in the late 1980s, the allocation of Class C came into the picture; the problem was that because of 24-bit network address, Class C allocation can severely impact the routing table size in the core of the Internet. To alleviate this problem, the concept of Classless Interdomain Routing (CIDR) was introduced. CIDR allows the use of explicit address blocks. CIDR is based on Variable Length Subnet Masking (VLSM). CIDR helps control the routing table size by aggregating addresses together and advertising them as a single route wherever possible with the help of the varying length of the subnet mask.

The current Internet routing architecture is under tremendous amount of load and is questionable whether it is scalable in the long term. The current size of the BGP routing table stands at 600,000 entries which is a very large number while the size of the forwarding information bases (FIB) on core routers is not getting any smaller and

currently stands at 305,334 both of which are increasing super-linearly[1] (see Figure 1). Because of the massive size of the routing table, the amount of space required to store them in memory on the routers is becoming large and is expensive. Moreover, the time taken to go through the entire table impacts latency. Also, another rising problem is the number of BGP messages getting processed at the core routers. All of these issues when considered together are starting to pose a difficult challenge to the growth of the Internet in the long term.

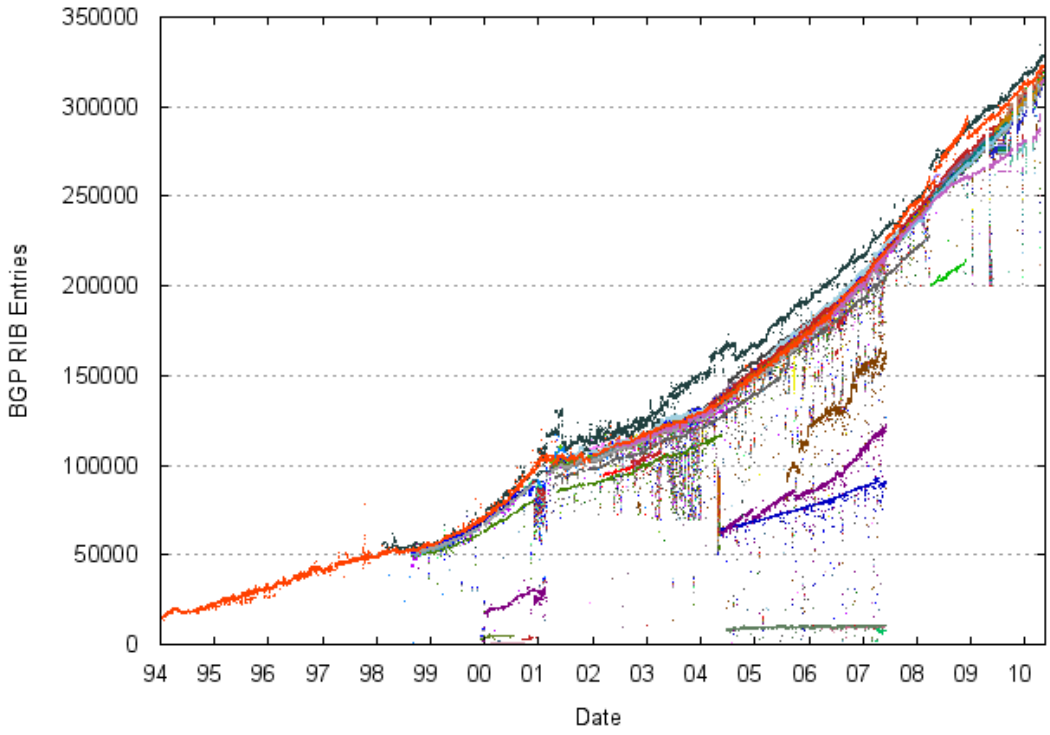

Figure 1. Growth of Routing Table[2]

[1] http://bgp.potaroo.net/
[2] http://bgp.potaroo.net/

LISP is an Internet routing protocol that is being developed at CISCO Systems for the past decade. It deals with improving the current condition of the Internet routing architecture. It is expected to alleviate the problems of the constantly increasing RIB and FIB size and consequently also on the number of BGP messages getting processed. LISP aims to achieve this with minimal changes to the existing hardware and software resources that are available.

1.2 Scope of Thesis

In this work, I undertook a detailed study of the LISP protocol. Chapter 2 summarizes the assumptions that are made in the routing architecture. Chapter 3 gives an introduction to LISP and definitions of the new terms introduced. Chapter 4 follows with the working of LISP, packet flow sequences and the study about the header formats. In Chapter 5 we discuss various messages associated with LISP. We also discuss locator reachability, mapping and clock sweeps. Chapter 6 discusses the interworking of LISP with non LISP sites and the concept of LISP-NAT. Chapter 7 is where we present the NERD which is the database system to be used with LISP and its record formats.

To date, there is little work on LISP and mobility. I present LISP with respect to mobility of the endpoints and how it will affect the scenario of mobile nodes. This is documented in Chapter 8. In Chapter 9, I present an analysis on LISP and compared it to the current routing architecture. Also, competitive comparisons with other protocols that are under consideration were conducted which is presented in Chapter 10.

CHAPTER 2

ASSUMPTIONS IN ROUTING ARCHITECTURE

In this section, we will discuss some of the implicit assumptions about the Interdomain routing architecture and take a look at some of the alternative to these assumptions.

2.1 Interdomain Routing Architecture Assumptions

2.1.1 IANA-based address allocation

IP addresses are manually assigned by the IANA or the other regional authorities which maintain a pool of IP addresses and allocate them in two different formats. They are:

- Provider Independent (PI) addresses
- Provider Aggregatable (PA) addresses

Providers who receive PA address also maintain their own registry and allocate sub-prefixes in their own PA space. The providers have network operators with the help of which they manually assign addresses from the received prefix. Hence, the only automation that the Internet managed was in the success of DHCP and auto-configuration. As the IP addresses are being manually configured, renumbering them is a big problem which gets compounded by the complex renumbering techniques which cannot be fully automated [1].

2.1.2 IP Address is both a locator and an identifier

Since the beginning, IP addresses have been considered to be connected to a single layer 2 interface. Hence, it serves as a locator. But it also serves as an identifier.

2.1.3 ASes are visible entities in the Interdomain routing system

The inter domain routing system is used to distribute prefixes. For the system, each prefix is associated with one AS. There have been mechanisms deployed to allow an AS to aggregate several prefixes before it announces a larger IP prefix, but those mechanism are not used by ISPs that much. ASes are very reliable in the current inter domain system because the AS-path is used as a loop avoidance mechanism in BGP [1].

2.1.4 Interdomain routing convergence

Whenever a link breaks, the protocol converges to let the entire Internet know of an alternate path. Recent measurements show that this convergence is slow; this is expected to be worse with more address prefixes to handle.

2.1.5 Traffic engineering

The interdomain routing architecture was designed to provide best-effort service, but due to traffic congestions, routing policies and other issues, this has been tweaked to achieve better load balancing and various traffic engineering objectives.

2.1.6 Security is not considered to be a strong concern

When Internet architecture was originally designed, security was not considered. The evolution of Internet has shown us that security has become a strong concern.

2.2 Alternative Interdomain Routing Architecture

2.2.1 Separating IDs and Locators

The separation of identifiers and locators will solve several problems. There are different ways to achieve this. One is in which the locator will identify the routers/middleboxes and the other one is in which it will identify the end systems [1].

2.2.2 Automatic Allocation of Locators

CIDR is found to have limitations mainly because of two factors:

- The growth of multihoming forced ISPs to advertise more and more specific prefixes which lead to an increased size of the FIB and the number of BGP update messages.
- There is pressure from enterprise networks to use PI address as IP address since renumbering is a costly affair.

A possible way to achieve this is that when a client network attaches to a provider network, a protocol should be used by the provider network to announce to the client network the prefix that the client network should use.

Six – One Router protocol is one of the many other protocols that are trying to solve this problem.

2.2.3 Removing ASes from the Interdomain routing system

Distribution of locators is one of the main objectives of interdomain routing architecture. ASes are not required to be a visible entity. ASes are made visible because they employ path-vector based interdomain routing which is used to avoid looping. But this also directly results in a fraction of the BGP messages that are exchanged. Newer approaches should be explored to avoid having them as visible [1].

2.2.4 Interdomain routing convergence

Link failures are common events that affect ASes both internally and externally. Studies have shown that it is better to react locally instead of globally. There are solutions that protect peering links in transit and stub ASes. Many solutions are being developed by the IETF to protect interdomain links.

2.2.5 Traffic Engineering

Traffic engineering is an essential requirement for both transit and stub ASes. This is often achieved by tweaking BGP advertisements. But, there are different types of traffic engineering, like:

- Selecting the path with the lowest delay to reach the given destination.
- Selecting the path with the highest bandwidth to reach the given destination.
- Forcing the outgoing packets to use a given peering link.
- Forcing incoming packets to use a given peering link.

With a locator-id split, the traffic engineering problem can be solved at different levels. The primary step is to map identifiers to locators. Tuning the mapping

mechanism is the most efficient and the most scalable way to allow networks to engineer their incoming packets flows.

CHAPTER 3

LOCATOR ID/SEPARATION PROTOCOL

3.1 Introduction

LISP is proposed to reduce scaling issues of a single numbering space for both host transport session identification and network routing. LISP is a network-based map-n-encap protocol which implements separation of Internet addresses into EIDs and RLOCs. Since end-systems operate the same way they do today in the LISP, it requires no changes to host stacks and existing database infrastructures. The existing network layer is "jacked up" and a new network layer is inserted below it. The LISP "jack-up" is depicted in the following Figure 2.

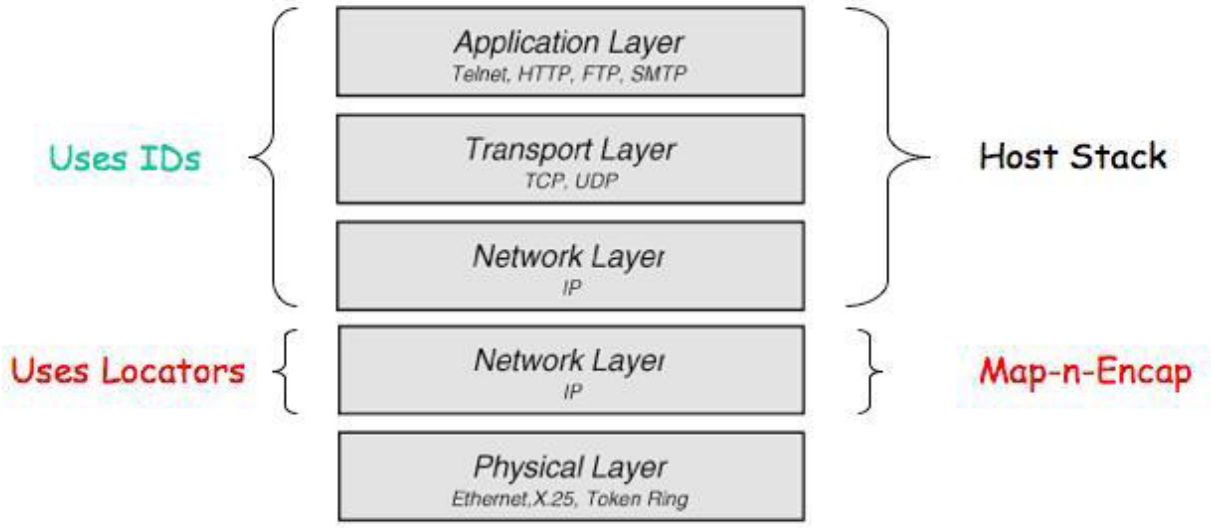

Figure 2. LISP Jack-up in Network Layer[3]

[3] http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_11-1/111_lisp.html

3.2 Design goals

This protocol is still under development and is one of the many proposals for the problem at hand. Some of the design goals of this protocol include:

- No need for changes to the hardware and software at the end-systems (hosts).
- Minimal changes to the Internet architecture.
- Can be deployed incrementally.
- No router hardware changes. Essentially, minimize the number of routers which need to be modified. Most customer site routers and core routers don't require changes. Also, minimize the amount of software changes in the routers that will get affected.
- Minimize the packet loss that may occur during EID - to - RLOC mapping.

3.3 Variants of LISP

There are four different variants of LISP which differ along the lines of strong to weak dependencies on the topological nature and the possibility of routability of EIDs. These variants are [2]:

- LISP 1: Uses EIDs that are routable and through the RLOC topology for bootstrapping EID-to-RLOC mappings. This was just a prototype stage version and has been deprecated now and is not used anymore.
- LISP 1.5: Uses EIDs that are routable for bootstrapping EID-to-RLOC mappings. This routing is via a separate topology.
- LISP 2: Uses EIDs that are not routable and EID-to-RLOC mappings are implemented with the help of DNS.

- LISP 3: Uses non-routable EIDs which are used to lookup keys for an EID-to-RLOC mapping database. There are various examples for employing such a database. One such example is Distributed Hash Tables (DHTs).

3.4 Definition of terms

In this section, we give a brief description of all the various terms associated with LISP [2].

- Provider Independent Address (PI Address): A block of address assigned from the pool, where the blocks are not associated with any particular location in the network. Hence, it is not aggregatable topologically in the routing system.
- Provider Assigned Address (PA Address): A block of address assigned to a site by a service provider. Generally, each PA address block is a sub-block of the service provider's CIDR block and is aggregated into the larger block before it is advertised into the global Internet.
- Routing Locator (RLOC): RLOC is the IPv4/IPv6 address of an ETR. It is the output of the EID-to-RLOC mapping lookup. An EID can map to one or more RLOCs. RLOCs are typically, numbered from topologically-aggregatable blocks that are assigned to a site at each point to which it attaches to the global Internet. Where the topology is defined by the connectivity of provider networks, RLOCs can be thought of as PA addresses.
- Endpoint ID (EID): EID is a 32-bit IPv4 or 128-bit IPv6 value used in the source and destination address fields of the inner most LISP header. The

host will obtain a destination EID the same way it obtains a destination address today. The source EID is obtained using the existing mechanism used to set a host's local IP address. An EID-prefix block is allocated to the site. Each host is allocated the EID from this block. An EID can be used by a host to refer to other hosts. In order to scale the mapping database, the EID blocks can be assigned in a hierarchical manner independent of the network topology. EIDs cannot be used as LISP RLOCs.

- EID-prefix: EID-prefixes are associated with the RLOCs addresses which make up the "database mapping". Generally, a power of two block of EIDs is allocated to a site by an address allocation authority. EID-prefix allocations can be sometimes broken up into smaller blocks if the RLOC set is to be associated with the smaller EID-prefix. A globally routed address block is not an EID-prefix. However, such globally routed address blocks can be removed from global routing and used as an EID-prefix. If a site has been explicitly allocated an EID-prefix, they cannot use it as a globally routed prefix assigned to RLOCs.
- End-system: Is an IPv4 or IPv6 machine that originates packets with a single IPv4 or IPv6 header. It supplies an EID for the destination address for the address field of the IP header when communicating globally.
- Ingress Tunnel Router (ITR): ITR is a router which accepts an IP packet with no LISP header appended to it. The router treats the destination address in the IP header (inner header) as an EID and performs an EID-to-RLOC mapping. The router then prepends one more IP header (outer header) with its globally-routable RLOC in the source address field and

the result of the mapping in the destination address field. Generally, an ITR receives IP packets from the end-system on one side and sends LISP-encapsulated IP packets towards the Internet on the other site. Hence, destination RLOC might be an intermediate router which has a better knowledge of the final destination.

- TE-ITR: This is an ITR that is deployed in the network for traffic engineering purposes. It appends an additional LISP header to achieve its aim.
- Egress Tunnel Router (ETR): ETR is a router that accepts IP packets where the destination address in the outer header is one of its own RLOCs. The router then strips the outer header and forwards the packet based on the information in the inner header. Generally, ETR receive LISP encapsulated IP packets on one side and forward them to the end systems on the other side.
- TE-ETR: Just like a TE-ITR, a TE-ETR is deployed in the network for traffic engineering purposes. An ETR strips the additional LISP headers appended by any TE-ITR.
- xTR: xTR is a reference to an ITR and an ETR when the direction of flow is not in the context of the discussion.
- EID-to-RLOC Cache: It is a short-lived cache that acts as an on demand table in an ITR that stores, tracks and is responsible for timing-out and otherwise validating EID-to-RLOC mappings. This cache is dynamic and local to the corresponding ITR and relatively small when compared to the database.

- EID-to-RLOC Database: This is a globally distributed database that contains all the known EID-prefix to RLOC mappings. Every potential ETR contains a piece of this database for the EID-prefixes behind itself. These prefixes, map to one of the router's own globally visible IP address.
- Recursive Tunneling: When a packet has more than one LISP header, it is because of traffic engineering or whatever re-routing reasons. Recursive tunneling implies such situations and is always dynamically configured.
- Re-encapsulating Tunnels: When a packet has more than one LISP header and it needs to be a diverted to a completely new RLOC location, the ETR can strip the outer header and prepends a new header so that the packet reaches its new destination.
- LISP Header: This is the term used to describe the outer IPv4 or IPv6 header that is appended to the IP packet. A LISP header is appended by an ITR and stripped by an ETR.
- Address Family Indicator (AFI): AFI is a term used to describe the address encoding inside the packet. It can be either IPv4 or IPv6.
- Negative Mapping Entry: This mapping entry is one where the EID-to-RLOC mapping advertises no EID-prefixes. This type of entry can be used to describe a prefix from a non-LISP site, which is not present explicitly in the mapping database.
- Data Probe: This is a probe message that is sent to request a Map-reply message in order to discover the destination and the routes.

CHAPTER 4

OVERVIEW AND TUNNELING DETAILS

4.1 Introduction

One of the key points in LISP is that the end-systems keep operating the same way as they do today. The IP addresses that the host (end-systems) use for tracking sockets, connections and for sending and receiving packets does not change. In LISP terminologies as we saw in the earlier section, these are known as EIDs.

The routers will continue to forward packets based on IP destination addresses. When a LISP encapsulated packet is being routed, these addresses are referred as RLOCs. Most routers will continue to operate as they do today and keep forwarding the packets. For routers that are connected to the end-systems or xTR, the destination addresses are the EIDs. For the routers between the ITR and the ETR, the addresses are RLOCs. This design of LISP introduces “tunnel-routers” which append LISP headers at the sending side and strip them at the receiving side. The IP addresses in these outer LISP headers are RLOCs. The ITR performs EID-to-RLOC lookups in order to determine the routing path to the ETR, which has the RLOC as one of its IP addresses.

4.2 Basic rules governing LISP

Some of the basic rules governing the functioning of LISP are [2]:

- End-systems only send to addresses which are also EIDs. They do not know that EIDs are mapped with RLOCs. They just assume that the packets reach the LISP routers which deliver the packet to the appropriate destination.
- EIDs are always IP addresses assigned to the hosts.
- RLOCs are always IP addresses assigned to the routers, preferably topologically oriented addresses from provider CIDR blocks.
- When the router is the source of the packet, it may use as a source address either an EID or a RLOC. When it is acting as a host (Telnet, SSH connections) it can use an EID which is explicitly assigned for this purpose. An EID which identifies a router should never be used as an RLOC because the EID is routable only locally within the scope of that site. A good example of such hybrid behavior can be a BGP configuration where the router uses its local EID to terminate iBGP sessions and its RLOC to handle eBGP sessions.
- EIDs are not supposed to be used globally for end-to-end communication. If the EID-to-RLOC mapping is not present, then they are supposed to be used for intra-site communications only.
- EID-prefixes are assigned hierarchically in a manner which is optimized for administrative convenience and to facilitate scaling of the EID-to-RLOC mapping database.
- When recursive tunneling is employed, the specification mandates that no more than two LISP headers be prepended to the packet. This way, excessive packet overhead is avoided and possible encapsulation loops are avoided too. It is

assumed that two headers are sufficient where the first header is appended by the ITR and the second by a TE-ITR.

4.3 Packet flow sequence

Let us see how the packet flows when LISP is being employed. Figure 3 shows how packets flow:

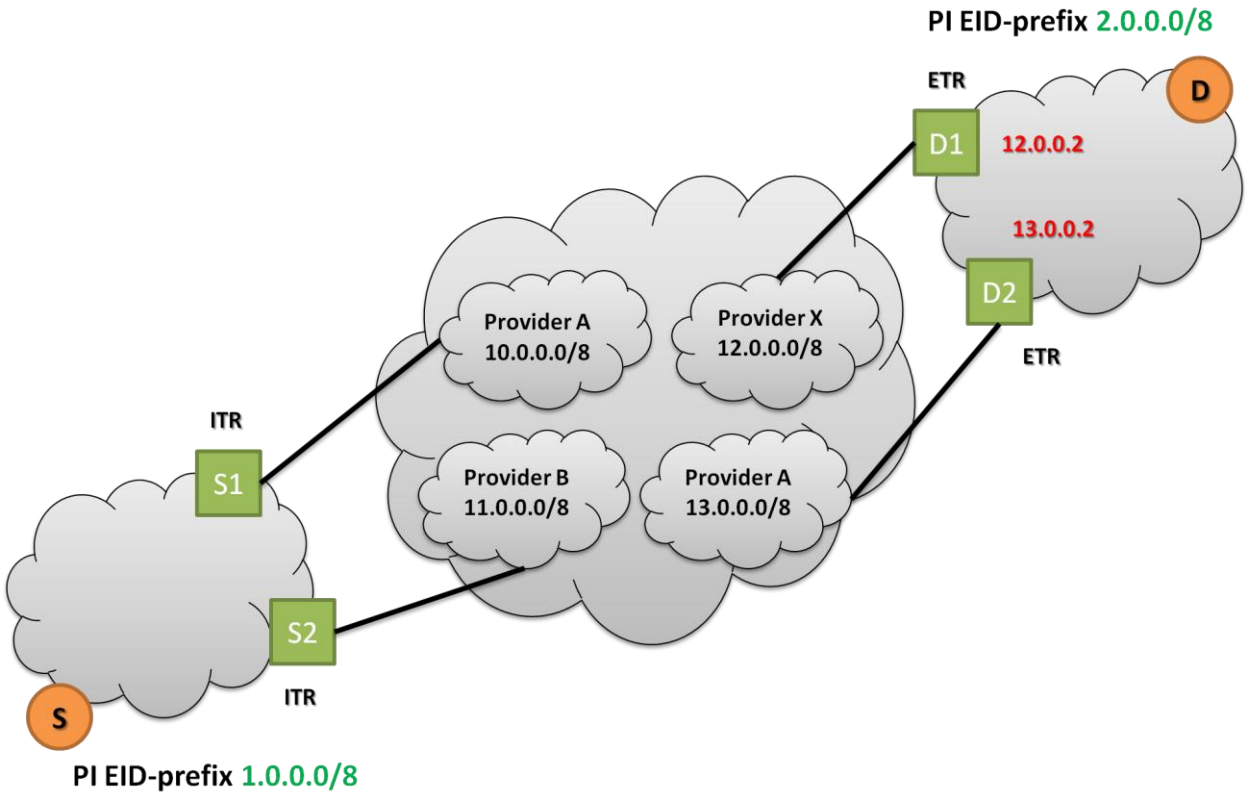

Figure 3. Transmission of LIPS packet

- Source S wants to transmit a packet to destination D. Source S will open a TCP connection to destination D. It will do a DNS lookup on D. After receiving the address from DNS, it will use this address as the destination EID and the locally assigned IP address as the source EID. An IPv4 or IPv6 packet is built using the EIDs in the IPv4 or IPv6 header and sent to the default router.
- The default router S1 is configured as an ITR. The ITR must be able to map the destination EID to an RLOC of the ETR at the destination site. The ITR prepends

a LISP header to the packet with one of its RLOCs as the source in the IPv4 or IPv6 address. The destination address EID from the original packet header is used as the destination IPv4 or IPv6 in the prepended LISP header. The packets will be sent with the outer destination address set to EID till the EID-to-RLOC mapping is obtained.

- In the deprecated LISP 1, the packet is routed through the Internet as it is today. In LISP 1.5, the packet is routed through a different topology which may have many EID prefixes distributed and advertised in an aggregatable fashion. For LISP 2 and 3, the behavior is not fully defined. In either case, the packet arrives at the ETR.
- At the ETR, the LISP header is stripped and the packet is forwarded by the router. The router will look up for the destination EID in its EID-to-RLOC database. The ETR then forwards the packet appropriately.
- A Map-Reply message is configured and sent on the opposite direction to the RLOC on the source side. On receiving the Map-Reply message, the source ITR will after checking the message for format validity, cache the information.
- Subsequent packets from the source to the destination will have the ETR's RLOC as the destination address in the prepended LISP header.
- The ETR will receive these packets directly, strip the LISP header and forward the packet to the concerned end-system.
- In order to eliminate the mapping lookup in the reverse direction, the ETR may make a cache entry for the source ITR's RLOC.

4.4 Tunneling Details

As additional tunnel headers are appended to the IP packets, the packets might become larger than the MTU of any link traversed from ITR to ETR. If the IPv4 packet is bigger than the MTU, then it is not fragmented as it is encapsulated by the ITR. Instead of this, they are dropped. Generally, the MTU for the large ISPs is at least 4770 bytes. The LISP deployment will include collecting the data related to MTU during its pilot messages to either verify or refute the general assumption made about the MTU size [2].

4.5 LISP IPv4 in IPv4 Header Format

LISP IPv4 in IPv4 Header Format is shown in Figure 4.

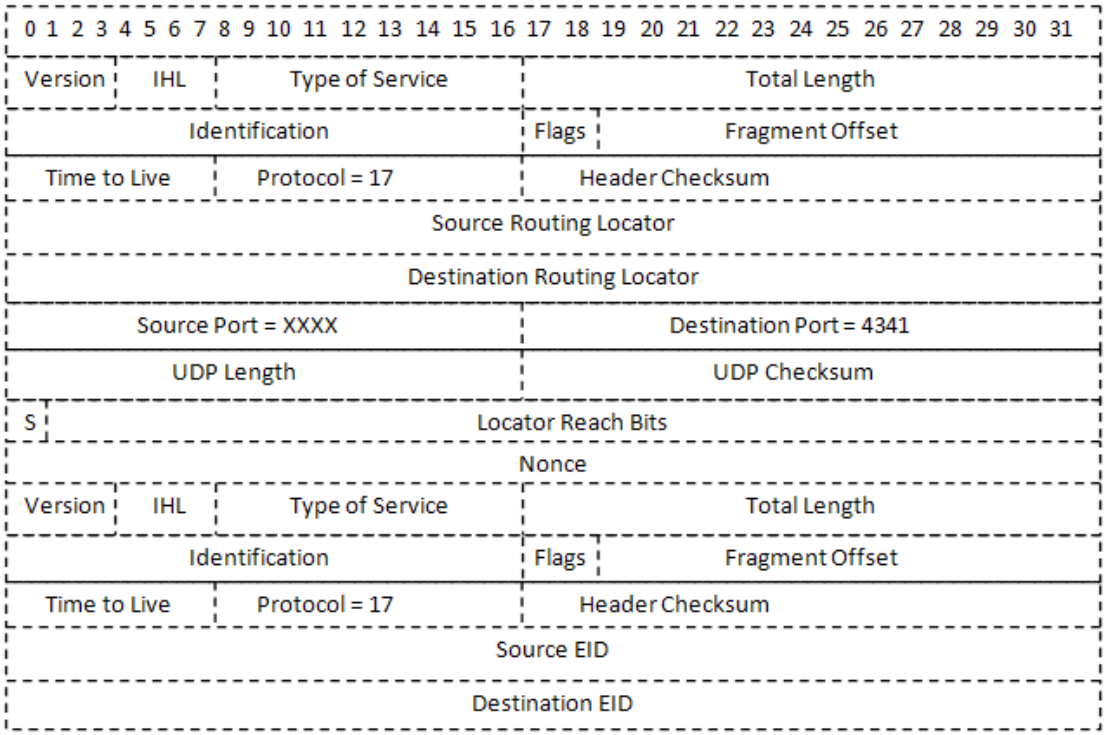

Figure 4. LISP IPv4 in IPv4 Header Format[4]

[4] http://tools.ietf.org/html/draft-ietf-lisp-04

4.6 LISP IPv6 in IPv6 Header Format

LISP IPv6 in IPv6 Header Format is shown in Figure 5.

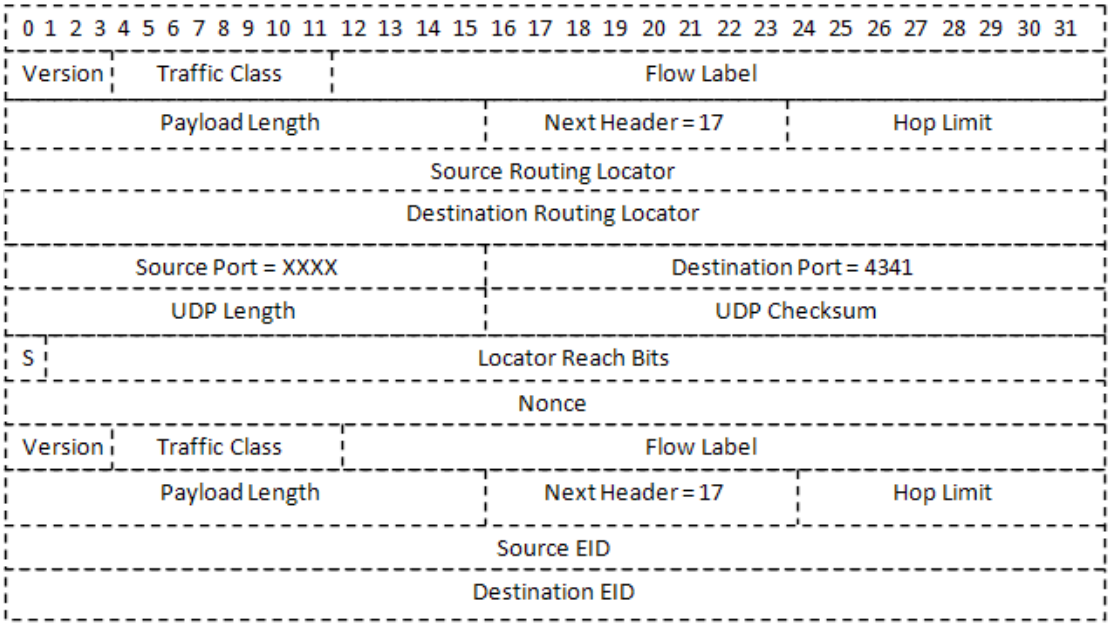

Figure 5. LISP IPv6 in PIv6 Header Format[5]

4.7 Tunnel Header Field Descriptions

- IH Header: This is the inner header which is obtained from the original IP packet received from the originating host. The source and destination addresses in this header are the EIDs.
- OH Header: This is the outer header which is prepended by an ITR. The addresses fields contain the RLOCs which are obtained from the ingress router's EID-to-RLOC cache.
- UDP Header: This field contains the ITR selected source port when encapsulating a packet. The destination must be set to a well-known IANA assigned port value 4341.

[5] http://tools.ietf.org/html/draft-ietf-lisp-04

- UDP Checksum: This field must be transmitted as 0 and ignored on receipt by the ETR. Even though we are transmitting a 0 as the checksum, an intermediate NAT device can recalculate it and assign a different value to it. For this purpose, it needs to be ignored on receipt by the ETR.
- UDP Length: For an IPv4 encapsulated packet, the inner header total length, the UDP and LISP header lengths combined is used. For an IPv6 encapsulated packet, the inner header payload, the size of the IPv6 header, UDP and LISP headers combined is used.
- S: This is the Solicit-Map-Request bit.
- LISP Locator Reach Bits: These bits are set if the ITR wants to indicate to an ETR the reachability of the locators in the source site. Each RLOC in a Map-Reply is assigned an ordinal value from 0 to (n-1). The locator reach bits are numbered from 0 to (n-1) from the right significant bit of the 31-bit field. When a bit is set to 1, the ITR is indicating to the ETR the RLOC associated with the bit ordinal is reachable.
- LISP Nonce: It is a 32-bit value which is randomly generated by an ITR. It is used to test the route-returnability when xTRs exchange encapsulated data packets with the SMR bit set, Data-Probe, Map-Request or Map-Reply messages.
- When doing recursive tunneling, the TTL field (or the Hop limit field for IPv6) of the OH Header should be copied from the IH Header TTL field. The OH Header TOS field (or the Traffic Class field for IPv6) should be copied from the OH header TOS field.
- When doing re-encapsulating tunneling, the new OH Header TTL should be copied from the stripped OH Header. The new OH Header TOS field should be

copied from the stripped OH Header. Copying the TTL helps preserve the distance the host wanted the packet to travel. More importantly, it provides for suppression of looping packets.

4.8 Dealing with large encapsulated packets

If the MTU issue mentioned turns out to be very grave, then there have been two suggestions to tackle it. One is stateless while the other is stateful. The stateless method uses IP fragmentation while the stateful one uses Path MTU discovery [2].

- Stateless solution to MTU Handling

  Define and architectural constant S for the maximum size of a packet, in bytes that an ITR will receive from the source. Define L to be the maximum size, in bytes. A packet of size S and the LISP header will make the maximum size. i.e. L = S + H.

  When an ITR receives a packet which is bigger than the maximum size after adding the header H, it splits the packet into two equal sized fragments. A LISP header is then prepended to both the fragments. This ensures that the two fragments are of size (S/2 + H) which will always be less than L.

  When the ETR receives the packet, it will treat them as two separate packets. It strips the LISP headers then forwards each fragment to the destination host. At the destination host, the two packets are reassembled together to obtain the single IP datagram that originated at the source host.

  If the source packet is an IPv6 packet or the DF field of the IP Header is set, then the ITR will not fragment the packet. Instead it will drop the packet and respond with an ICMP too big message to the source with the value of S.

- Stateful solution to MTU Handling.

  The ITR will keep state of the effective MTU for each locator per mapping cache entry. This effective MTU will be the size that the core network can provide along the path from the ITR to the ETR.

  When an encapsulated packet exceeds what the core network can support, the router along the path which is not capable of handling the message will send an ICMP too big message. The ITR then will make an entry in its cache with this new value for MTU.

  The next time the ITR receives a packet from the source which exceeds the MTU; it will send an ICMP too big message back to the source along with the MTU. Although, this mechanism is stateful, it has an advantage over the stateless mechanism as it does not involve the receiver with re-assembling the packets.

CHAPTER 5

MESSAGES

5.1 LISP IPv4 and IPv6 Control Plane Packet Formats

The LISP UDP based control plane messages are Map-Request and Map-Reply messages. When the map request message is used; the UDP source port is chosen by the sender and the destination port is set to 4342. When the map reply message is used; the source port number is set to 4342 and the destination port is copied from the source port of the map request or the invoking data packet [2]. The packet formats for IPv4 and IPv6 control plane are shown in Figures 6 and 7, respectively.

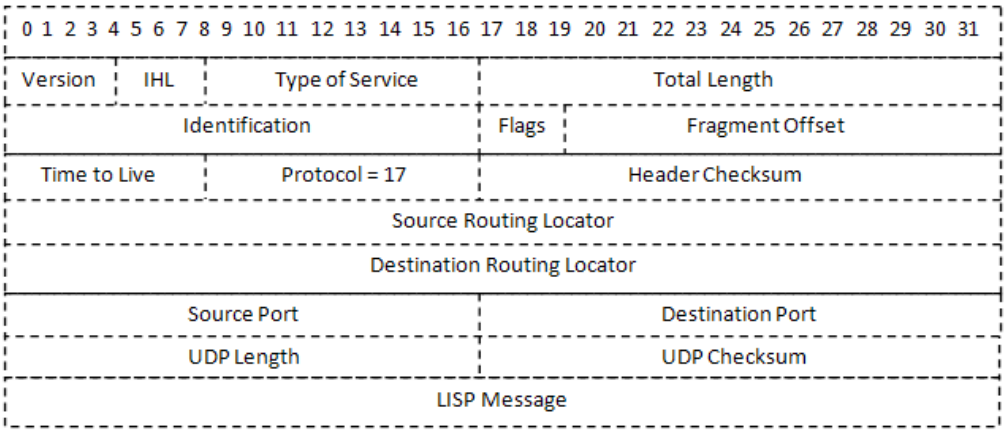

Figure 6. LISP IPv4 Control PlanePacket[6]

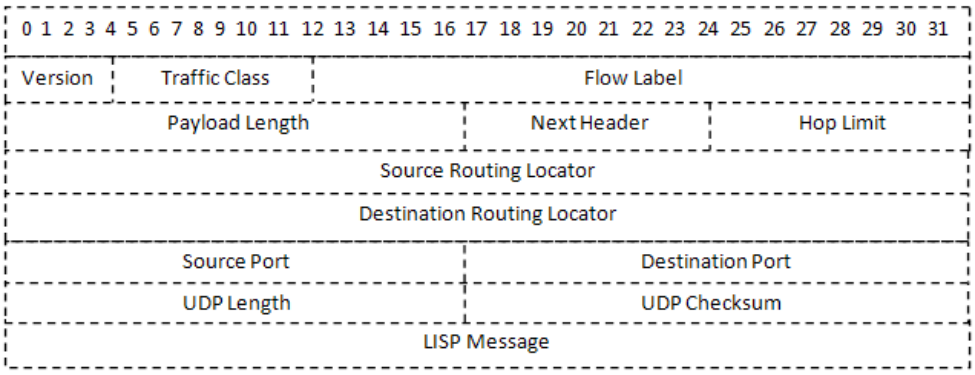

Figure 7. LISP IPv6 Control Plane Packet[6]

5.2 Map-Request Message Format

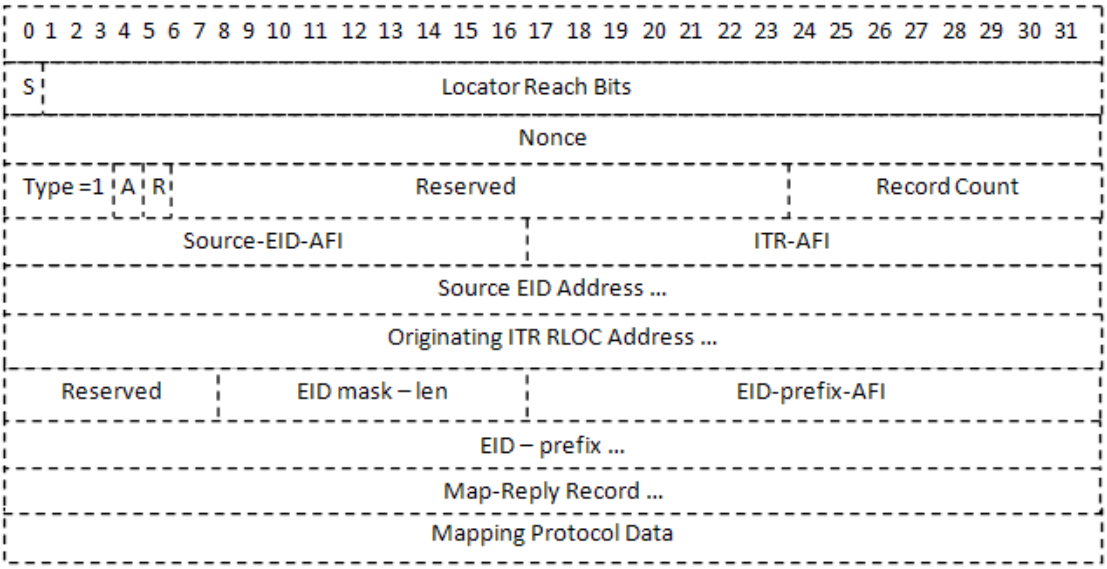

Figure 8. Map-Request Message Format[6]

Map-Request Message Format is shown in Figure 8. It has the following key fields:

- S: This is the SMR (Solicit Map Request) bit.
- Locator Reach Bits: These bits must be set to 0 on transmission and ignored on receipt. They are not used because the map-request does not have the EID-prefix. As a result of this, the receiver of the message does not know what to map it with. However, the mapping data is provided in the map reply message in the map-reply record. The receiver of the map-request is configured to accept the mapping data, the R-bit per locator entry in the EID-prefix record is used to denote reachability.
- Nonce: It is a 4-byte random value created by the sender of the Map-request.
- A: This is an authoritative bit.
- R: When set, it indicates a Map-reply record segment is included in the map-request.
- Reserved: Set to 0 on transmission and ignored on receipt.

[6] http://tools.ietf.org/html/draft-ietf-lisp-04

- Record Count: The number of records in this request message. A record is composed of portion of the packet is labeled ‘Rec’ above and occurs the number of times equal to Record count.
- Source-EID-AFI: Address family of the “Source EID Address” field.
- ITR-AFI: Address family of the “Originating ITR RLOC Address” field.
- Source EID Address: This is the EID of the source host which originated the packet which is invoking this Map-Request.
- Originating ITR RLOC Address: This is used to give the ETR the option of returning with a Map-Reply in the address family of this locator.
- EID mask-len: Mask length for EID prefix.
- EID-AFI: Address family of EID-prefix.
- EID-prefix: 4 bytes if it is an IPv4 address-family and 16 bytes if it is an IPv6 address-family. When a map-request is sent by an ITR because a data packet is received for a destination where there is no mapping entry, the EID-prefix is set to the destination IP address of the data packet. The EID mask-len is set to 64 and 128 respectively for IPv4 and IPv6. If a xTR is querying a site about the status of a mapping it has already cached, the EID-prefix used in the Map-request has the same mask-length as the EID-prefix returned from the site when it sent a Map-Reply message.
- Map-reply record: When the R bit is set; this field is the size of the “Record” field in the Map-Reply format. This record consists of the EID-to-RLOC mapping entry associated with the source EID. This allows the ETR which will receive this Map-Request to cache the data if it chooses to do so.

- Mapping protocol data: This field is optional and present when the UDP length indicates there is enough space in the packet to include it.
- A Map-Request is sent from an ITR when it needs a mapping for an EID, wants to test reachability, or wants to refresh a mapping before TTL expires. In all cases, the UDP source port number for the Map-Request message is a randomly allocated 16-bit value and the UDP destination port number is set to the well-known destination port number 4342. A successful Map-reply updates the cached RLOC associated with the EID prefix range. Map-requests can also be LISP encapsulated using UDP destination port 4341 when sent from an ITR to a Map-Resolver [2].

5.3 Map-Reply Message Format

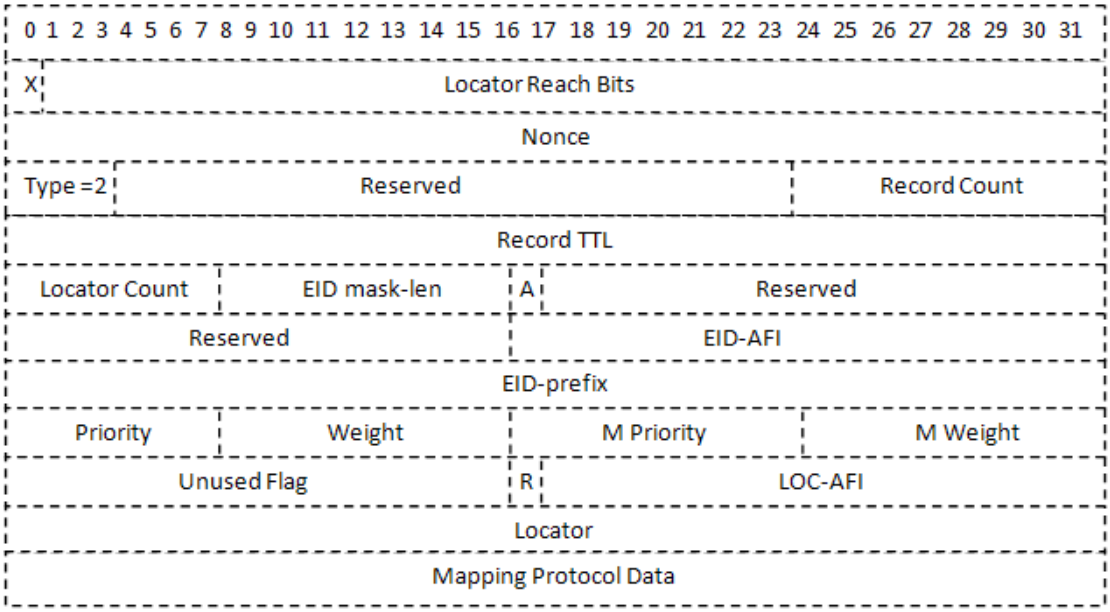

Figure 9. Map-Reply Message Format[7]

[7] http://tools.ietf.org/html/draft-ietf-lisp-04

Map-Reply message format is shown in Figure 9. It has the following fields:

- X: Set to 0 on transmission and ignored on receipt.
- Locator Reach Bits: This bit must be set to 0 on transmission and ignored on receipt. The locator reachability is encoded as the R-bit in each locator entry of each EID-prefix record.
- Nonce: It is a 4-byte value set in a Data-Probe packet or Map-Request that is echoed again here in the Map-Reply.
- Type: Set to 2 (Map-Reply)
- Reserved: This field is set to 0 on transmission and ignored on receipt.
- Record Count: This field represents the number of records in this reply message. A record is the part of the message labeled "Record". It occurs the number of times equal to the number of record count.
- Record TTL: The time in minutes, the recipient of the Map-Reply will store the mapping.
- Locator Count: The number of Locator entries. A locator entry comprises what is labeled above as "Loc". The locator count can be 0 indicating there are no locators for the EID-prefix.
- EID mask len: Mask length of EID prefix.
- A: The authoritative bit is always set by the ETR.
- EID-AFI: Address family of EID-prefix.
- EID-prefix: 4 bytes if an IPv4 address-family, 16 bytes if an IPv6 address-family.
- Priority: Each RLOC is assigned a unicast priority. When multiple RLOCs are present with the same priority, then they may be used in a load-split fashion. A

lower, value is more desirable. If the value is set to 255, then the ROLC must not be used for unicast.

- Weight: When priorities are the same for multiple RLOCs, the weight indicates how to balance unicast traffic between them. It is encoded as a percentage of total unicast packets that match the mapping entry. If a non-zero weight value is used for any RLOC, then all RLOCs must use a non-zero weight value and then the sum of all weights should be equal to 100. If any of them has a zero value, then all of them should have the zero value. And then, the receiver will decide on how to split the load.
- M Priority: Each RLOC is assigned a multicast priority used by an ETR in a receiver multicast site to select an ITR in a source multicast site for building multicast distribution trees. A value of 255 means that the RLOC must not be used for joining a multicast distribution tree.
- M Weight: When priorities are the same for multiple RLOCs, the weight indicates how to balance building multicast distribution trees across multiple ITRs. The weight is encoded as a percentage of total number of trees build to the source site identified by the EID-prefix.
- Unused Flags: This field is set to 0 when sending and ignored on receipt.
- R: When this bit is set, the locator is known to be reachable from the Map-Reply sender's perspective. When there is a single mapping record, the R-bit must match the Loc Reach Bits. If there are multiple records, this is set to zero.
- Locator: An IPv4 or IPv6 address assigned to an ETR or router acting as a proxy for the EID-prefix. The RLOC must not be a multicast address or a link local

multicast address. The destination RLOC can be a multicast address if it is being mapped from a multicast destination EID.

- Mapping Protocol Data: This field is optional and present when the UDP length indicates there is enough space in the packet to include it.

When a data probe packet or Map-Request triggers a Map-Reply to be sent, RLOCs which are associated with the EID-prefix matched by the EID in the original packet IP address field will be returned. The RLOCs in the Map-Reply are globally-routable IP addresses of ETR but are not necessarily reachable. Separate testing of reachability is required.

A Map-Reply may contain different EID-prefix granularity than the Map-Request which triggers it. This may happen if the Map-Request was for a prefix that had been returned by an earlier Map-Reply. In this case, the requester will update its cache with the new prefix information and granularity.

Replies should be sent for an EID-prefix no more than once per second to the same requesting router. For scalability, it is expected that aggregation of EID addresses into EID-prefixes will allow one Map-Reply to satisfy a mapping for EID-prefixes will allow one Map-Reply to satisfy a mapping for the EID addresses in the prefix range thereby reducing the number of Map-Request messages [2].

5.4 Routing Locator Selection

Both client side and the server-side may need control over the selection of RLOCs for the impending conversation between them. This is achieved by manipulating the priority and weight fields in EID-to-RLOC Map-Reply messages. Alternatively, RLOC information may be gleaned from received tunneled packets or EID-to-RLOC Map-Request messages. The following enumerates different scenarios for choosing RLOCs and the controls that will be available.

- Server-side will return one RLOC. Client-side can use only one RLOC. The server-side ahs the complete control over the selection.
- Server-side will return a list of RLOCs where a subset of the list has the same best priority. Client can only use the subset list according to the weighting assigned by the server-side. In this case, the server will be controlling both the subset list and the weighting assigned. The client side can use the RLOCs outside of the subset list if it determines that the subset list is unreachable. The client side has the option of using alternatives to the server-side's subset list if it is unreachable.
- Server-side will set weight to 0 for the RLOC subset list. In this, the client side can choose how the traffic load is spread across the subset list. Control is shared by the server-side determining the list and the client determining load distribution.
- Either side decides not to send Map-Request. If the server-side does not send Map-Request, it gives the client side the responsibility for bidirectional RLOC reachability and preferability.
- RLOCs that appear in EID-to-RLOC Map-Reply messages are considered reachable. The Map-Reply and the database mapping service do not provide any reachability status to locators.

5.5 Routing Locator Reachability

There are numerous methods to determine whether the router is reachable at that given time or not. They are:

- Reachability is determined by an ETR by examining the Loc-Reach-bits from a LISP header of an encapsulated data packet which is provided by an ITR when an ITR encapsulates data.
- Unreachability is determined by an ITR on receiving an ICMP Network message.
- It can also be determined by a BGP enabled ITR when there is no prefix matching a locator address from the BGP RIB.
- Unreachability can be determined when a host sends an ICMP port unreachable message.
- Locator reachability is determined by receiving a Map-Reply message from a ETR's locator address in response to a previously sent Map-Request.
- Locator reachability can also be determined by receiving packets encapsulated by the ITR assigned to the locator address.

When determining Locator reachability, an ETR will receive up to date information by examining the Loc-Reach-Bits of a LISP encapsulated data packet it received from the ITR closest to the locators at the source site. The ITRs in turn determine the reachability when running IGP at the site. The ITR is deployed on the CE routers; typically a default router will be injected into the site's IGP from each of the ITRs. If an ITR goes down, the default router will be withdrawn by the CE router. This will allow the other ITRs at the site to determine that one of the locators has become unreachable [2].

The locators listed in the Map-Reply message are numbered from 0 to (n-1). The Loc-Reach-Bits in a LISP Data message are numbered 0 to (n-1) starting with the least significant bit numbered as 0. So, for example if the ITR with locator listed as the 4th locator position in the Map-Reply message goes down, then all the other ITRs at the site will have the 4th bit from right cleared.

When the ETR is decapsulating a packet, it will look for changes in the Loc-Reach-Bits value. If a bit has gone from 1 to 0, then the ETR will not encapsulate packets to the Locator that has just gone down. It will start using that Locator again only when the bits changes back to 1 from 0. When a locator becomes unreachable, the loc-reach-bit that corresponds to that locator's position in the list returned by the Map-Reply will be set to zero for that particular EID-prefix.

If an ITR receives an ICMP Network or host unreachable message, it can send out its own ICMP unreachable message to the host that originated the data packet the ITR encapsulated.

5.6 Routing Locator Hashing

When an ETR provides an EID-to-RLOC mapping in a Map-Reply message to a requesting ITR, the locator-set for the EID-prefix may contain different priority values for each locator address. It is left up to the ITR to decide how to load share the traffic is more than one best priority locator exists. The following hash algorithm may be used by an ITR to select a locator for a packet destined to an EID for the EID-to-RLOC mapping:

- Either a source and destination address hash can be used or the traditional 5-tuple hash which includes the source and destination TCP, UDP or SCTP port numbers and the IP protocol number field or IPv6 next protocol field of a packet a host originates from within a LISP site.
- Take the hash value and divide it by the number of locators stored in the locator-set for the EID-to-RLOC mapping.
- The remainder will be yield a value of 0 to "number of locators minus 1". Use the remainder to select the locator in the locator-set.

5.7 Changing contents of EID-to-RLOC mappings

The LISP architecture uses a caching scheme to retrieve the stored EID-to-RLOC mappings. The only way an ITR can get a more up to date mapping is by requesting one. The ETRs don't keep a track of the changes to the mapping and there is no way for the ITRs to know in advance when the mappings will change. Due to scalability reasons, this is the desired mode of operation, but there needs to be a way to make sure the ETR can advertise to the current connections it has of the changes to the mappings.

When a locator record is added to the end of a locator-set, it is easy to update the mappings. The ITRs with the new mappings will have the entry at the last and the ITRs using the old mappings will have the smaller cache till it dies. If the ITR has old mapping and it receives bits set in Loc-reach-bits, which are beyond its caches list, then it simply ignores it.

When a record is removed, ITRs that have the mapping cached will not use the removed locator as the loc-reach-bit will be set to 0 by the corresponding ETRs. For new mapping requests, the locator address and the loc-reach-bits are set to 0, so the new ITRs will not try to connect to them.

If there are many changes taking place in the mapping over a long period of time, then, there will be many empty slots in the middle of the locator-set and many new mappings added at the back of the set. It will be useful, to squeeze out these empty slots and compact the locator set so that it becomes more efficient. There are two approaches to achieve this. The first one is the clock sweep method and the second is solicit-map-request.

5.8 Clock Sweep

This approach uses planning in advance and the use of countdown TTLs to time out the mappings that have already been cached. The default setting for EID-to-RLOC mapping TTL is set to 24 hours. So, there is a 24 hour window to time out the old mapping. The following procedure is used:

- 24 hours before a mapping change is going to take effect, a network administrator configures the ETRs at a site to start the clock sweep window.
- During the clock sweep window, ETRs continue to send Map-Reply messages with the current mapping records. The TTL is set to 1 hour for these mappings.
- At the end of 24 hours, all previous cache entries will have timed out, and any active ones will time out in 1 hour. During this one hour period, ETRs continue to send Map-Reply messages with the current mapping records with the TTL set to 1 minute.
- At the end of the hour mark, the ETRs will send Map-Reply messages with the new mapping records [2].

5.9 Solicit-Map-Request (SMR)

Soliciting a Map-Request is a selective way for xTRs, at the site where the mappings change; to control the rate they receive requests for Map-Reply messages. They are also used to tell remote ITRs to update the mappings they have cached.

Since the xTRs do not keep a track of the remote ITRs that are using their mappings, they cannot tell which ITRs need updates. So, a xTR will solicit Map-Requests from sites it is currently sending encapsulated data to. The xTRs can locally decide the algorithm as to how often and how many sites will send these SMR messages [2].

An SMR message is a bit set in an encapsulated data packet. When an ETR at a remote site decapsulates a data packet that has the SMR bit set, it can tell that a new Map-Request message is being solicited. Both the xTR that sends the SMR message and the site that acts on it must be rate-limited. The following procedure shows how a SMR exchange occurs when a site is doing locator set compaction for an EID-to-RLOC mapping:

- When the database mappings in an ETR change, the ITRs at the site begin to set the SMR bit in packets they encapsulate to the sites they communicate with.
- A remote xTR which decapsulates a packet with the SMR bit set will schedule sending Map-Request message to the source locator address of the encapsulated packet. The NONCE in the Map-Request is copied from the NONCE in the encapsulated data packet that has the SMR bit set.
- The remote xTR transmits the Map-Request slowly until it gets a Map-Reply while continuing to use the cached mapping.

- The ETRs at the site with the changed mapping will reply to the Map-Request with a Map-Reply message provided the Map-Request NONCE matches the NONCE from the SMR. The Map-Reply messages should be rate limited.
- The ETRs at the site with the changed mapping, records the fact that the site that sent the Map-Request has received the new mapping data in the mapping cache entry for the remote site so the loc-reach-bits are reflective of the new mapping for packets going to the remote site. The ETR hen stops sending packets with the SMR bit set.
- For security reasons, ITRs must not process unsolicited Map-Replies.

CHAPTER 6

INTERWORKING LISP WITH IPv4 and IPv6

6.1 Introduction

A key point of separating the locators and the end-point-ids is that EID prefixes are never advertised in what we know as the Default Free Zone (DFZ). Only, RLOCs are carried in the DFZ. Because of this, existing Internet sites which do not support LISP should still be able to reach sites which are numbered from this non routed EID space.

There are namely two mechanisms recognized as feasible to provide reachability between sites that are LISP-enabled and those which are not LISP enabled. The first uses a Proxy Tunnel Router (PTR) which acts as a LISP intermediate ITR for the non-LISP sites. While the other one adds a notion of Network Address Translation (NAT) functionality to the xTRs to replace the routable IP addresses for the non-routable EIDs.

6.2 Interworking Models

There can be 4 different combinations of unicast connectivity which describe how the LISP enabled and the LISP non enabled sites can communicate with each other.

1. Non-LISP site to Non-LISP site.

   The first case is similar to our conventional Internet.

2. LISP site to LISP site.

   The second case has been explained above for LISP to LISP communication.

3. LISP site to Non-LISP site.

In the third case of a LISP site sending packets to a Non-LISP site, a LISP site can send packets to a Non-LISP site since the non-LISP prefixes are routable. The only hurdle in this communication is to know when NOT to LISP-encapsulate a packet. This can be achieved via two mechanisms [3].

- If the ITR at the source site cannot find the destination address in the EID-to-RLOC mapping database, then it can safely assume that the site is not LISP-enabled and can send the packet without the LISP encapsulation.
- Also, at the source site, if the destination address for an IP packet is found to match one of the prefix in the BGP routing table, then that means the site is directly reachable by the BGP core that exists and operates today.

4. Non-LISP site to LISP site.

This is the most demanding case at hand. For a packet to make it out of the non-LISP site and make it to a LISP-speaking router is very challenging. The packet will keep getting routing within the site till it gets dropped. In order to avoid it from being dropped, two mechanisms are introduced as mentioned before. They are the PTR and LISP-NAT. It is interesting to note that case three here becomes a subset of case four as it includes packets returning from the LISP site to the non-LISP site.

6.3 Definition of new terms

Some new terms have been defined to better explain the new mechanisms. These are [3]:

- EID-Prefix Aggregate: A set of EID-prefixes which are said to be aggregatable. It is defined as a single contiguous power-of-two EID-prefix block.
- EID Prefix reachability: An EID prefix is said to be reachable if one or more of its locators are reachable. That is, the EID prefix is reachable if the ETR is reachable.
- Default Mapping: A default mapping is a mapping entry for EID-prefix 0.0.0.0/0. It maps to a locator-set used for all EIDs in the Internet. If a more specific EID-prefix mapping entry is available, then the default mapping is overridden.
- LISP Routable Site (LISP-R): This is a LISP site whose address is globally routable. It can be used as both a globally routable IP address and as an EID.
- LISP Nonroutable Site (LISP-NR): This is a LISP site whose address is not globally routable and can be used only as an EID.
- LISP Proxy Tunnel: (PTR): PTRs are used to provide connection between sites that use LISP and the ones that do not use LISP. They act sort of as a gateway between the legacy Internet and the LISP network. A PTR will advertise one or more aggregated EID prefixes into the Internet domain and will then act as the ITR for traffic received from the Internet.
- LISP Network Address Translation (LISP-NAT): It is like the traditional NAT. Network Address Translation between the EID assigned to a site and the RLOC assigned of the same site.

6.4 Routable EIDs

A very obvious way to achieve interworking between LISP and non-LISP hosts is to simply announce EID prefixes into the DFZ. As a result of this, the EIDs will behave as PI prefixes. The impact will be similar to the case in which LISP is not being deployed at all. This hampers the primary objective of LISP which is to reduce the global routing system state and defeats the purpose of having LISP in the first place. Non-LISP sites use BGP to enable ingress traffic engineering. Relaxing this requirement is another primary design goal of LISP [3].

6.5 Proxy Tunnel Routers

PTRs allow for non-LISP sites to communicate with LISP-NR sites. A PTR shares many of its characteristics with an ITR. PTRs facilitate non-LISP sites to send packets to LISP-NR sites without any changes in the protocol or any new hardware on the sender's side.

PTRs advertise highly aggregated EID-prefix space on behalf of LISP sites so that non-LISP sites can reach them. They also encapsulate non-LISP Internet traffic into LISP packets and route them towards their intended destination RLOCs. Aggressive aggregation is performed to ensure a minimal number of new announced routes. Also, placement of PTRs will affect the efficiency of the PTRs greatly. Hence, it is logical to deploy the PTRs closer to the non-LISP site rather than the LISP sites. Such a deployment will limit the scope of EID-prefix route advertisements. It also allows for better load balancing among the PTRs [3].

Packets can reach from a non-LISP site to a LISP-NR site via the PTR. The PTR will advertise for the particular EID-prefix in the global routing system. All the traffic designated for the EID prefix will be routed to the PTR which performs LISP

encapsulation. The following is an example of the process. Let us assume that the LIS-NR site has the EID-prefix 240.0.0.0/24.

- The source will do a DNS lookup for the EID destination and will get 240.1.1.1 in return.
- The source has the default route to the Customer edge router and will forward the packet till there.
- The customer edge router in turn has the route to the provider edge router and the packet gets routed till there.
- The provider edge router has the route to 240.0.0.0/24 and its next destination is the PTR.
- Depending upon the prevailing conditions, the PTR either already has the mapping or acquires it for 240.1.1.1. It then "LISP encapsulates" the packet. The packet will have the destination address of the RLOC and the PTR's RLOC as the source.
- The PTR will look up the RLOC and forward the packet to the next hop.
- The packet will reach ETR which in turn will decapsulated the packet and deliver it to the destination 240.1.1.1.
- Packets that originate from the LISP site for the non-LISP site will not go through the PTR. The ITR will recognize the destination address as globally routable and route it directly.

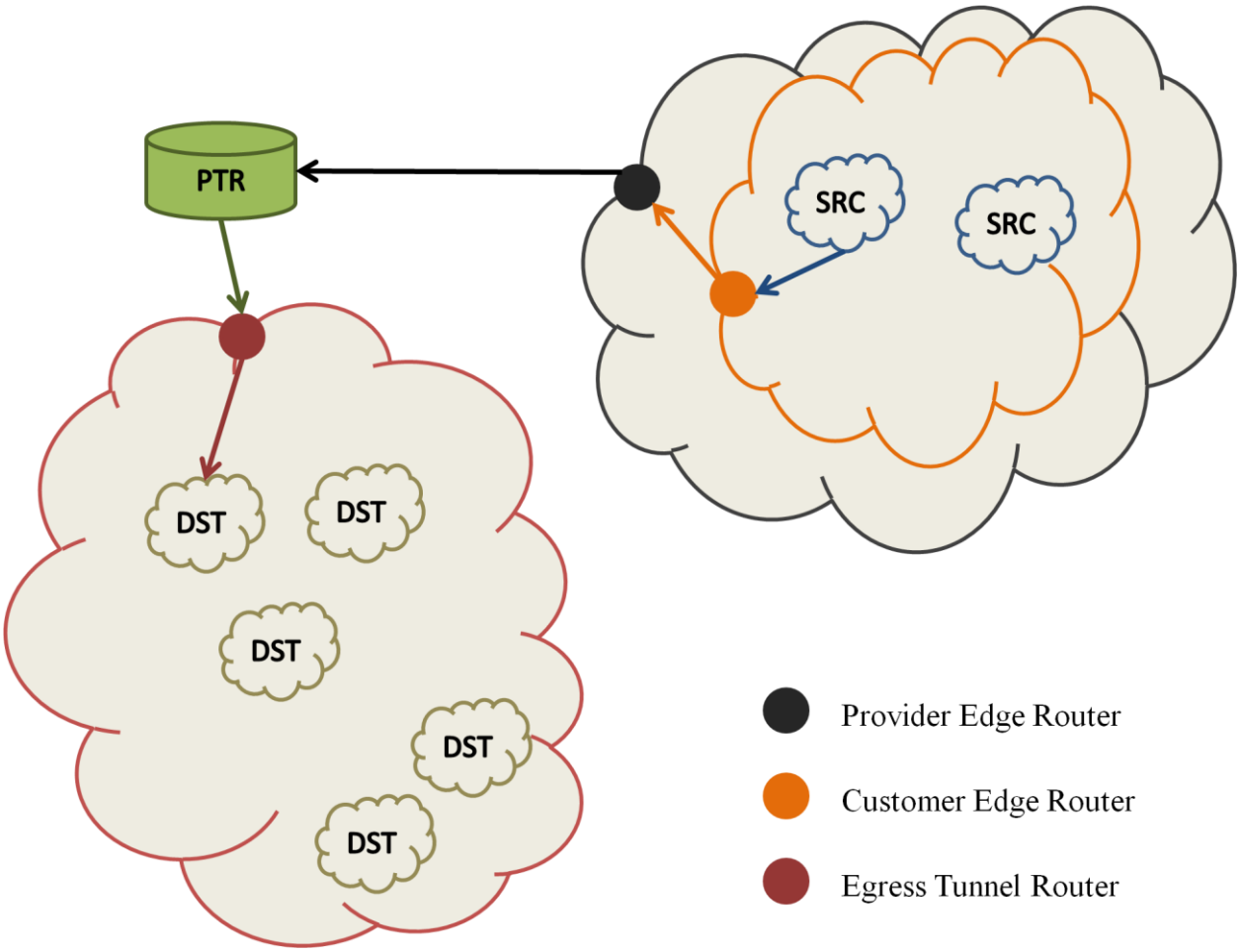

Figure 10. LISP Interworking

Figure 10 shows the flow for LISP Interworking. PTRs announce the LISP EID namespace in the non-LISP global routing system to grab traffic. There are many ways by which the network can control how traffic reaches the PTR to prevent it from receiving more traffic then it can handle. The PTR's aggregate routes can be selectively announced so that the quantity of traffic directed towards the PTR can be controlled. Also, many PTR's can announce the same address at the same time, to take off the load.

Placing the PTR close to the ingress of traffic allows for better communication between the non-LISP and the LISP site as the number of forward hops to reach the focal points is less. When traffic is destined for LISP-NR site arrives and is encapsulated by a PTR, a new LISP header is added to it. The new destination of the packet is the destination RLOC. As a result, the chance of the packet following the network's engineering policies is better. For large transit providers, deploying PTR's may attract more traffic which can result in more revenue.

6.6 LISP-NAT

LISP Network Address Translation is a type of NAT implementation. LISP-NAT is designed to enable the interworking between a non-LISP site and a LISP-NR site. This is achieved by ensuring that the LISP-NR's site address is always routable.

In LISP-NAT, the address of the source is translated into another address which is routable. A table is maintained for mapping the "inner" (original) address to the "outer" (translated) address.

The basic concept is that, when transmitting a packet, the ITR will replace a non-routable EID source address with a routable source address. LISP-NAT is not needed in the case of LISP-R as the source address is globally routable [3].

LISP-NAT allows a host that has a LISP-NR EID to communicate with non-LISP hosts by translating the LISP-NR EID to a globally unique address. Following is an example of the translation:

- Let us assume that the LISP-NR EID has been assigned the address 220.1.1.0/24. In order to employ LISP-NAT, the site is also provided with a PA-EID 128.200.1.0/24. Let us assume that it uses the first address 128.200.1.1 as the site's RLOC. The rest of the PA space is used as the translation pool. The translation table is shown in Table 1:

Table 1. Translation Table[8]

Site NR-EID	Site R-EID	Site's RLOC	Translation Pool
220.1.1.0/24	128.200.1.0/24	128.200.1.1	128.200.1.2 – 128.200.1.254

- The host sends a packet which is destined for a non-LISP site to its default ITR.
- The ITR receives the packet and determines that the destination is a non-LISP site. The method used to determine this is left up to the ITR.
- The ITR then changes the source address from 220.1.1.2 to 128.200.1.2 which is an available address in the LISP-R EID space available to it.
- The ITR keeps this translation in a table so that it can reverse the process when receiving the packets.
- When the ITR forwards this packet without encapsulating it, it uses the entry in its LISP-NAT table to translate the returning packets' destination IP to the proper host.

When a site has two addresses that a host can use for global reachability, care must be taken as to which EID is discovered by DNS. Should it pick up the LISP-NR EID or the LISP-R EID? Using PTR, we can mitigate this problem since LISP-NR can be reached in all cases.

6.7 LISP NAT & PTR together

With the LISP-NAT, there are two EIDs possible for a given host, the LISP-NR EID and the LISP-R EID. When a site has more than one address that can be used for global reachability, the name-to-address directories may get modified. This is a general problem in NAT. Using PTRs can mitigate this problem as the LISP-NR EID can be reached in all cases.

[8] http://tools.ietf.org/html/draft-ietf-lisp-interworking-00

CHAPTER 7

NERD

7.1 Introduction

NERD is the Not-so-novel EID to RLOC database. It is made up of the following components:

- A network database format.
- A change distribution format
- A database retrieval/bootstrapping method
- A change distribution method.

NERD is a way to avoid dropping packets. The database is specified in such a way that the methods used to distribute or retrieve the packets may vary over time. NERD will be potentially using several transport methods. Most notably it will be using HTTP as it is widely deployed and has restart and compression capabilities.

7.2 Assumptions for the database

In order to specify a mapping, it is important to realize what the contents of the mapping are and how the mapping will be used. The following assumptions can be made for LISP:

- The data contained by the mapping will not change when a link goes down or a broken link comes up. It will change only on provisioning or configuration operations. NERD can be used as a verification method to ensure that whatever operational mapping changes an ITR receives are authorized.
- Weights and priority are not defined on hop-by-hop basis. As a result of this, the information in the mapping will not change with respect to its position in the topology.

- Just like LISP, NERD is also designed to ease the inter domain routing; its use is intended to be within the Interdomain environment [4].

7.3 Theory of Operation

A NERD is generated by the same authority that generates the PI addresses which are used by sites as EIDs. As a part of generating them, the authority generates a digest for the database and signs it with a private key whose public key is a part of the X.509 certificate. That signature and the copy of the authority's public key are included in NERD.

NERD is distributed to a group of well-known servers. ITRs will retrieve an initial copy of the NERD via HTTP when they come into service. The ITRs are preconfigured with a group of certificates whose private keys are used by database authorities to sign the NERD.

ITRs then verify the validity of both the public key and the signed NERD digest. If either of the validation fails, then the ITR tries to retrieve the NERD from a different source. This process continues until either a valid database is found or the list of sources is exhausted.

Once a valid NERD is retrieved, the ITR installs it into both non-volatile and local memory. At some point the authority updates the NERD and increments the database version counter. At the same time it generates a list of changes, which it also signs, as it does with the original database.

Periodically, the ITR will poll from their list of servers to determine if a new version of the database exists. When a new version is found, an ITR will attempt to retrieve a change file, using its list of preconfigured servers. The ITR validates a change file just as it does the original database. Assuming the change file passes validation; the ITR installs new entries, overwrites existing ones and removes the empty ones, based on the content of the change file [4].

7.4 NERD Format

The NERD header consists of the database version and a signature that is generated by ignoring the signature field and setting the authentication block length to null. The authentication block itself consists of a signature and a certificate whose private key counterpart was used to generate the signature.

Records are kept sorted in numeric order with AFI plus EID as a primary key and mask length as secondary. In this way, after a database update it should be possible to reconstruct that database to verify the digest signature, which may be retrieved separately from the database for verification purposes.

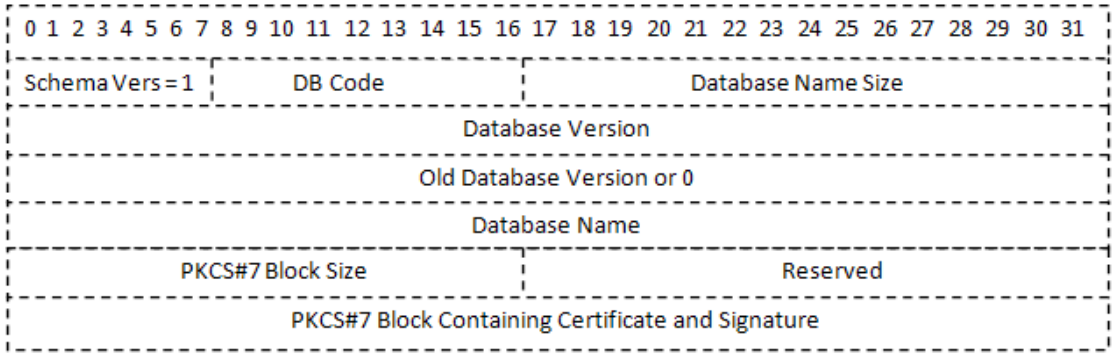

Figure 11. NERD Format[9]

[9] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

NERD format is shown in Figure 11. The DB code indicates 0 if what follows is an entire database or 1 if what follows is an update. The database file version is incremented each time the complete database is generated by the authority. In the case of an update, the database file version indicates the new database file version, and the old database file version is indicated in the "old DB version" field. The database file version is used to determine if they are the latest and most current database or not. The database domain name is a domain name. The purpose of the database name is to allow more than one database. These databases will be merged by the router. It is of prime importance that an EID/RLOC mapping be listed in no more than one database, otherwise there will be inconsistencies. It is quite possible to transition a mapping from one database to another. During the transition period, the mapping must be identical otherwise, the resultant behavior is undefined.

7.5 NERD Record format

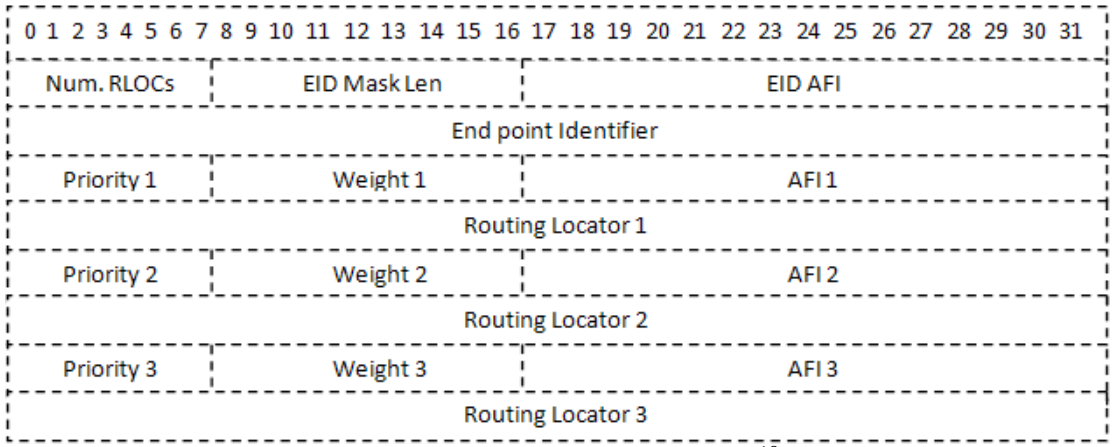

Figure 12. NERD Record Format[10]

NERD Record format is shown in Figure 12. Priority, weight and AFI are dependent on routing locator. There will always be at least one routing locator. The minimum record size for IPv4 is 16 bytes, while each additional record will increase the size by 8 bytes. The main purpose of this format is to keep the database compact,

[10] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

but still easily readable. In order to reduce the storage and transmission amount for IPv6, only the necessary number of bytes as specified by the prefix length is kept in the record, rounded to the nearest four byte boundary. IPv6 RLOCs are represented as normal 128-bit IPv6 addresses.

A database update contains a set of changes to an existing database. Each AFI or EID has an RLOC associated with it. In the case that there are no RLOCs, the EID entry is removed from the database. Records that do contain EIDs and mask lengths that were not previously listed are simply added. Otherwise, the old record for the EID and mask length is replaced by the more current information.

7.6 Initial bootstrap

Bootstrap occurs when a router needs to retrieve the entire database. It knows that it needs to retrieve the entire database because it either has none of it or an update will be too substantial to process completely.

To bootstrap the ITR appends the database name plus “current/entiredb” to the Base distribution URI and retrieves the file via HTTP. The Base Distribution URI is an absolute. It is used to construct a URI to an EID/RLOC mapping database [4].

The routers must check the signature on the database prior to installing it, and must check that the database schema matches the schema that they understand. Once, the routers have the valid database; they must store that database in a non-volatile memory.

7.7 Retrieving changes

In order to retrieve a set of database changes an ITR will have previously retrieved the entire database. Hence, it knows the current version of the database it has. The first step is to retrieve the current version of the database. It achieves this by appending the current/version to the base URI when retrieving the file. Once, the ITR has the current version, it compares version of its local copy. If there are no differences, then the router is up to date.

If the versions are different however, the router will then send a request for the appropriate change file. There is a good chance that the server might not have the change file that the router has requested, either because there have been too many revisions to the file and this particular version is not available or that this file version was not generated at all. If the server has any new file that can update the router, then the server should redirect the HTTP to that particular location and the router must get the new file [4].

7.8 Database Size

The information in the database, by its design is static in nature and topologically independent. While, some processing power will be required to setup the table but, it will be far less when compared to the amount required for the routing information database because the level of entropy will be lower.

For the purpose of the following analysis, let us assume that IPv6 has been successfully deployed since this will increase the size of the database. And also assume that the prefix length is limited to only 64 bits. Based on these assumptions and the fact that the mapping information for each EID/Prefix will include a group of RLOCs each with an associated priority and weight; also that the minimum record

size with IPv6 EIDs which hold at least one RLOC is 30 bytes when uncompressed. Each additional IPv6 RLOC will cost 20 bytes.

Table 2. Database Size[11]

10^n EIDs	2 RLOCs	4 RLOCs	8 RLOCs
4	500 KB	900 KB	1.7 MB
5	5 MB	9 MB	17 MB
6	50 MB	90 MB	170 MB
7	500 MB	900 MB	1.7 GB
8	5 GB	9 GB	17 GB

A simple formula can be used for the database size: E * (30 + 20 * (R - 1))

E = number of EIDs (10^n) and R = number of RLOCs in EID. Some representative values are shown in Table 2.

The scaling target is set to accommodate 10^8 multihomed systems. With these number of entries, a device will be expected to use anywhere between 5 to 17 GB of RAM. Any router that sits in the core of the network will require a similar amount of memory in order to perform the ITR functions. The large enterprise ETRs will also be similarly strained because of the diversity of the sites that communicate with each other. This is not the prevalent behavior. This is what being targeted as the scaling target. Under prevalent conditions, the number of EIDs is around 10^4 to 10^5, thus resulting in memory requirements of 500 KB to 17 MB.

[11] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

7.9 Router throughput versus time

Table 3. Router throughput versus time[12]

Table Size (10^N)	1 MB/s	10 MB/s	100 MB/s	1 GB/s
6	8	0.8	0.08	0.008
7	80	8	0.8	0.08
8	800	80	8	0.8
9	8000	800	80	8
10	80000	8000	800	80
11	800000	80000	8000	800

The length of the time it takes to process the database is significant in models where the device acquires the entire table because, during this time, either the router will be unable to route packets using LISP or the router needs to have some sort of query mechanism for specific EIDs while the rest of it gets populated through the transfer. Table 3 shows us the time it would take for the router to get the database at our scaling targets. With a download rate of 1 MB/s, it would take around 80 seconds. The fastest processing time we get for 1 GB/s for 10^9 is 8 seconds and for 10^10 bytes is 80 seconds.

[12] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

7.10 Number of servers required

The entries in the following table are generated using the following method:

- For 10^8 entries with four RLOCs per EID, the table size is 9 GB according to the previous table. Let us assume that we have a 1GB/s line and that full utilization is done. After ignoring the protocol overhead, a single transfer will take around 48 seconds and can get no faster. Hence, each entry is as follows:

$$Max(1X, N*X/S)$$

where, N = number of transfers, X = 72 seconds and S = number of servers.

Table 4. Number of Servers required (simultaneous requests) [13]

#simultaneous requests	10 Servers	100 Servers	1000 Servers	10000 Servers
100	720	72	72	72
1,000	7,200	720	72	72
10,000	72,000	7,200	720	72
100,000	720,000	72,000	7,200	720
1,000,000	7,200,000	720,000	72,000	7,200
10,000,000	72,000,000	7,200,000	720,000	72,000

If we have a distribution model that every device must retrieve the mapping information upon start, the table shows the length of time in seconds it will take for a given number of servers to complete a transfer to a given number of devices. From Table 4, it can be seen that it would take roughly 20 hours for 1 million ITRs to simultaneously retrieve the database from one thousand servers. If we consider a cold start scenario, this number is very high. Hence, it is of prime importance to avoid such a scenario or ease the load should it occur. The first line of defense against this case would be for the ITRs to retrieve the database from their peers or upstream providers.

[13] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

Secondary defense could be that a norm be agreed as to how much the database should change in any given update over any given period of time.

Table 5. Number of Servers required (% Daily Change)[14]

% Daily Change	100 Servers	1,000 Servers	10,000 Servers
0.1 %	300	30	3
0.5 %	1,500	150	15
1 %	3,000	300	30
5 %	15,000	1,500	150
10 %	30,000	3,000	300

Table 5 shows that with 10000 servers the average transfer time with 1 GB/s links for 10,000,000 routers will be 300 seconds with 10% daily change spread over daily updates. The amounts of changes depend upon the purpose of the implementation of LISP. If LISP is implemented for providing effective multi-homing support to end customers, then we might anticipate few changes. If on the other hand, service providers use LISP to achieve some sort of traffic engineering, then the data handled by LISP will have relatively more updates.

[14] http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B

CHAPTER 8

LISP AND MOBILITY

8.1 Introduction:

Mobile IP helps mobile nodes to communicate when they are not in their home network. Figure 13 describes how the basic functionality of mobile IP works.

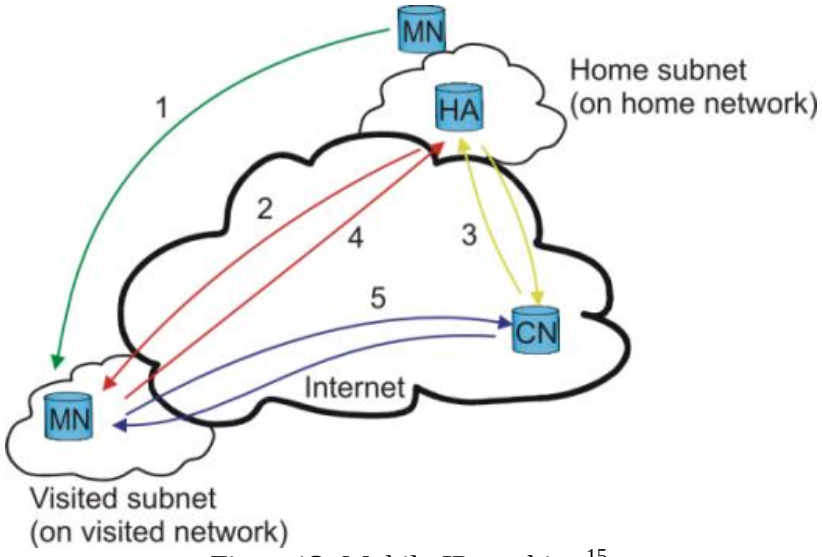

Figure13. Mobile IP working[15]

When a mobile node (MN) moves into the foreign network, it receives a Care of Address (CoA). It then contacts its home agent (HA) and informs it about the new location and provides the HA the CoA. When a correspondent node (CN) tries to communicate with the MN, the HA tunnel's the packets to the MN using the CoA. Once the MN replies, the communication either directly between the CN and the MN or via the tunnel created by the HA.

[15] http://tldp.org/HOWTO/Mobile-IPv6-HOWTO/intro.html#mobileIP

8.2 Mobility using LISP

With help of LISP, it is possible to achieve a mobile IP like behavior for standard IP Addresses without the need for the CoA or the HA to do any tunneling.

In LISP, the packets are transmitted to the host through the ITR sitting in the middle on the core network. In LISP, the IP address is divided into two parts, the RLOC and the EID. The IPv4 address will look something like this:

72.14.253.147.10.0.0.1

Where, the first four bits are for RLOC and the last four bits are for EID's. As an IPv6 address is large, the IP address is just split into two parts.

2001:0102:0304:0506:1111:2222:3333:4444

When the transmitter sends a packet, it goes to RLOC in the core network first. The RLOC has the mapping as to which ITR the packet should be going to. The packet then goes to the ITR from where it is sent to the host.

Now, when the host moves from behind one ITR to another ITR, a map request is sent which updates the host's location in the mapping tables. Now, when another packet is being sent, it gets directed to the new ITR from where the packet is forwarded to the host.

The difference between LISP and Mobile IP is that LISP achieves this without having a care of address since the combination of the RLOC and EID part of the IP Addresses will be unique. This will save overhead. Figure 14 explains the scenario.

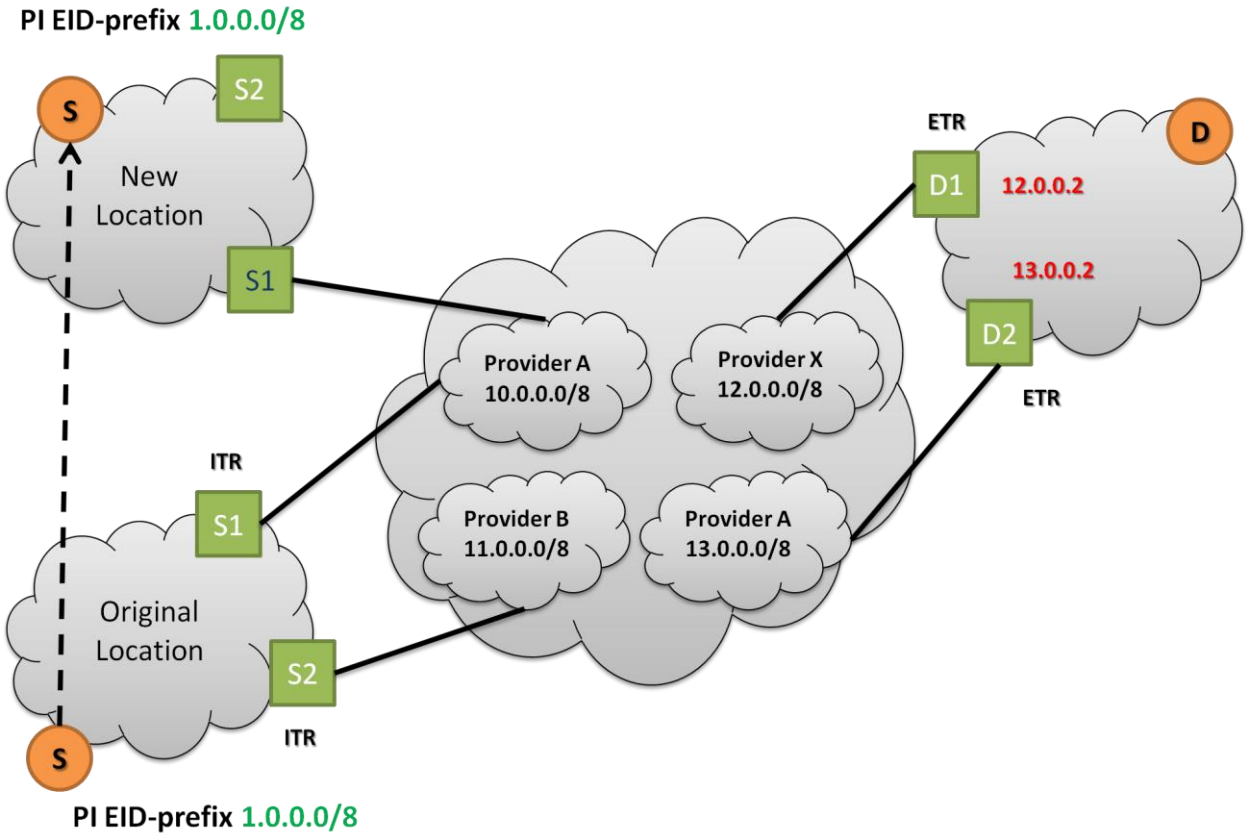

Figure14. Mobility with LISP

- Let us assume that S is the source which sent packets to destination D.
- After sending the packet, the source moved to its new location as shown in the figure.
- After moving, a set of map-request and map-reply messages are sent updating the location of the source in the mapping for the RLOC.
- When the destination sends something to the source, the packet comes to the RLOC, where the new location is already updated and the packet gets transmitted to that location.
- The PI EID prefix of the machine remains same even after moving from one location to another.

- Also, since the RLOC is the same, the IP Address essentially does not change and re routing is achieved without any redirections.
- This reduces the number of packets travelling in the network due to redirection and reduces the chances of having dropped/lost packets.

  The map request messages are sent periodically and hence, the locations will remain up to date at most times.

8.3 Merits of this Approach

- This approach reduces the number of packets travelling in the network due to redirection.
- This also reduces the chances of having dropped/lost packets.
- This also reduces the number of extra handshaking signals that need to be transmitted (signals between MN and HA in case of normal mobile IP)
- These are counted as extra since the map-request and map-reply messages in LISP are already considered as a part of LISP overhead.

8.4 Demerits of this Approach

- The map-request and map-reply message should be transferred as soon as the node enters the new network.
- We will still need a separate CoA in the new network if the new network is not supported by the host's provider.
- If the node has just completed a map-request/map-reply session and then moves into the new network, then the communication will be broken unless one more request/reply session is enforced or till the next session takes place.

- The amount of time taken for the map request and map reply messages will vary with respect to the traffic. So, it might take time for the map-request and map-reply messages to come in, which might result in loss of data and connection.

CHAPTER 9

IMPACT ON ROUTING TABLE AND EDGE NETWORK ROUTERS

9.1 Introduction

Since LISP will aggregate EIDs behind RLOCs, it will help in containing the growth of the routing table. Figure 15illustrates a simple example explaining how LISP will be impacting the routing table size.

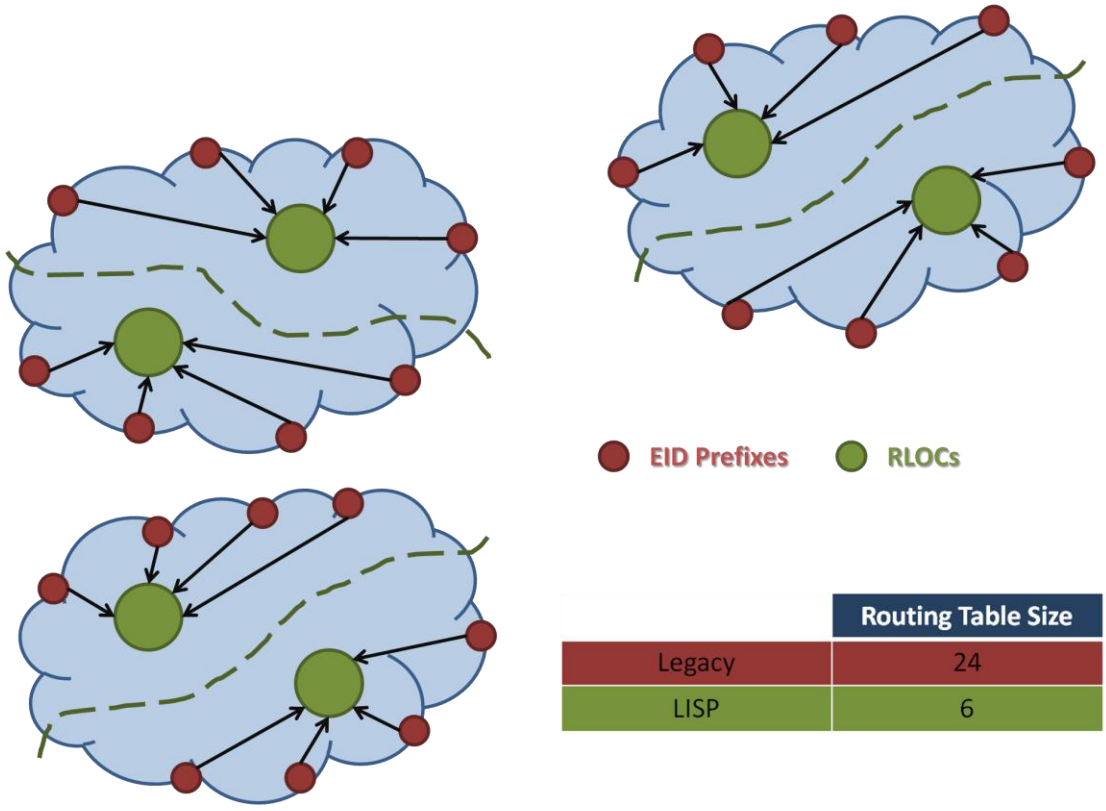

Figure 15. Impact on Routing Table Size

From Figure 15, we can see that when LISP is not being deployed each network will have to advertise 8 prefixes. This means the total size of the routing table will be 24. But when LISP is deployed, for every 4 prefixes, 1 RLOC is being defined. Hence, the total number of entries in the table will go down to 6.

9.2 A Simple Quantitative Analysis.

We can carry out a simple mathematical analysis to see how much LISP will help in reducing the routing table size. Let us assume that there are 100,000 prefixes present and that there are five networks.

Now, in the current behavior, let us also assume that the load is equally distributed among the five networks. So each network will have to advertise:

100,000/5 = 20,000 prefixes

Now, assume that LISP is being deployed and that for 10 EIDs we are advertising 1 RLOC. Therefore, every network will have to then advertise:

20,000/10 = 2,000 RLOCs

As we can see, the number of entries being advertised has gone down from 20,000 to 2000. From the above calculation, a simple formula can be derived to determine the number of entries that will be advertised when LISP is deployed as:

$$\frac{\text{(\# of prefixes)/ (\# of networks)}}{\text{(\# of prefixes per RLOC)}}$$

Table 6 provides a comparison for the routing table size in the current scenario versus the routing table size when LISP is deployed.

Table 6. Table for the impact on the routing table size when LISP is deployed

# of prefixes	# of N/W's	Legacy Table Size	LISP Deployed Table Size(prefixes/RLOC)		
			10 prefixes	20 prefixes	50 prefixes
100,000	2	50,000	5,000	2,500	1,000
	5	20,000	2,000	1,000	400
	10	10,000	1,000	500	200
200,000	2	100,000	10,000	5,000	2,000
	5	40,000	4,000	2,000	800
	10	20,000	2,000	1,000	400
500,000	2	250,000	25,000	12,500	5,000
	5	100,000	10,000	5,000	2,000
	10	50,000	5,000	2,500	1,000
600,000	2	300,000	30,000	15,000	6,000
	5	120,000	12,000	6,000	2,400
	10	60,000	6,000	3,000	1,200

9.3 Impact on ITR and ETR

In LISP, a major part of the processing has moved from the core networks to edge networks. The ITRs and the ETRs will now have to do extra processing. The extra processing is namely:

- Appending/Stripping off the additional LISP header.
- Traversing the database to map the EID to the RLOC.

Since, ITR and ETR will be handling this now, there will be some hit taken at the edge network.

In the current scenario, BGP Update messages are used to transmit IP Prefixes to the routers. IPv4 has been considered for all of the following calculations. Let us assume that,

Each BGP update message holds 1 IP prefix.

N: No. of IP Prefixes = 100,000

L: Link Speed = 1 Gbps

S_{BGP}: Size of BGP Packet = 4096 bytes = 32768 bits

$S_{BGPUpdate}$: Size of BGP update message = 23 bytes = 184 bits

S_{IP}: Size of the IP Prefix = 23 bytes = 184 bits

N_{BGP}: Number of BGP packets required

M: Number of BGP update messages in one BGP packet.

S_T: Total size in bytes of the packets required to transmit all the IP Prefixes.

S_{LISP}: Total size in bytes of the LISP database.

T_{cur}: Total time taken to read all the prefixes in current architecture.

T_{LISP}: Total time taken to read all the prefixes in the LISP architecture.

Therefore, we can calculate M as:

$$M = 32768/184 = 179 \text{ update messages}$$

In order to carry 100,000 prefixes we need:

$$N = 100,000/179 = 559 \text{ packets}$$

Hence, the total size can be computed as:

$$S_T = 562 \text{ X } 32768 = 18,317,312 \text{ bits} = 18.31 \text{ Mb}$$

Finally, the total time taken can be computed as:

$$\mathbf{T_{cur} = 18.31 \text{ Mb}/ 1\text{Gbps} = 0.017 \text{ seconds}}$$

Now, let us assume that LISP has been deployed with 10 EIDs = 1 RLOC. Then each network as calculated earlier will have to advertise 2000 entries. Hence, each ITR/ETR will have to read 10,000 entries. The size of a single entry in the NERD database is 16 bytes.

The total size of the LISP database can be computed as:

$$S_{LISP} = 10,000 \text{ x } 16 \text{ x } 8 = 1,280,000 \text{ bits} = 1.28 \text{ Mb}$$

The total amount of time taken to read all the entries is:

$$\mathbf{T_{LISP} = 1.28Mb/1Gbps = 0.0012 \text{ seconds}}$$

As it can be seen, under current routing architecture, it will take 0.017 seconds to read all the prefixes while in the LISP architecture it will take 0.0012 seconds to read the entire database.

Let us further assume,

T_{router}: Time taken by router to transmit a packet = 0.01 second

$T_{cur\text{-}total}$: Total time taken by current architecture to read prefixes plus transmit packet.

$T_{LISP\text{-}total}$: Total time taken by LISP architecture to traverse database and transmit packet.

$$\mathbf{T_{cur\text{-}total} = 0.017 + 0.01 = 0.027 \text{ seconds}}$$

In LISP, there will be additional time added when the ITR or ETR traverses through the database and for appending or removing the header.

Let us assume that,

α = the amount of time taken to traverse through the database. This will depend on the size of the database.

β = the amount of time taken to append or remove a LISP Header. Let us assume it is equal to 0.04.

Now, S_{LISP} = 1.28 Mb.

Let R: Rate at which the router looks up the database. Let us assume R = 1Gbps.

Then,

$$\alpha = 1.28\text{Mb}/1\text{Gbps} = 0.0012 \text{ seconds.}$$

So, $T_{LISP\text{-}total}$ can be calculated as,

$$\mathbf{T_{LISP\text{-}total} = 0.01 + 0.0012 + 0.04 + 0.0012 = 0.0164 \text{ seconds}}$$

A formula can be derived based on the preceding calculations:

$$\mathbf{T_{LISP\text{-}total} = T_{router} + \alpha + \beta + T_{LISP}}$$

Table 7 gives different values of α when we use different values for number of entries in the database and the rate at which the router traverses the database.

Table 7. Value of α for different values of table size and R

Table size # of entries	**Rate at which the router traverses the label (α) (seconds) 100,000 prefix range**		
	1 Gbps	**2 Gbps**	**5 Gbps**
10^4	0.0012	0.0006	0.00024
10^5	0.012	0.006	0.0024
10^6	0.12	0.06	0.024
10^7	1.2	0.6	0.24
10^8	12	6	2.4
10^9	120	60	24

Looking at the table and comparing the values with the current routing architecture, we can see that despite the additional processing time at the ITR and the ETR, we still get a gain of around 0.01 seconds.

Figure 16graphs plots the value for the LISP architecture (R = 1 Gbps):

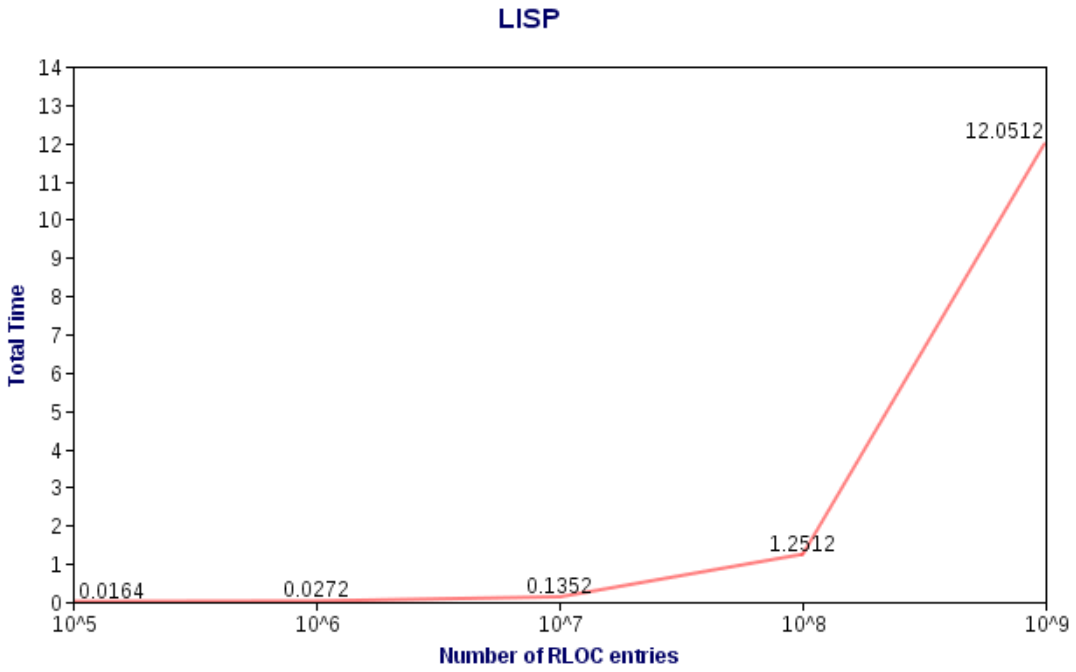

Figure 16. Plot for Time taken to transmit packet (LISP)

Figure 17 graphs plots the values for current routing architecture:

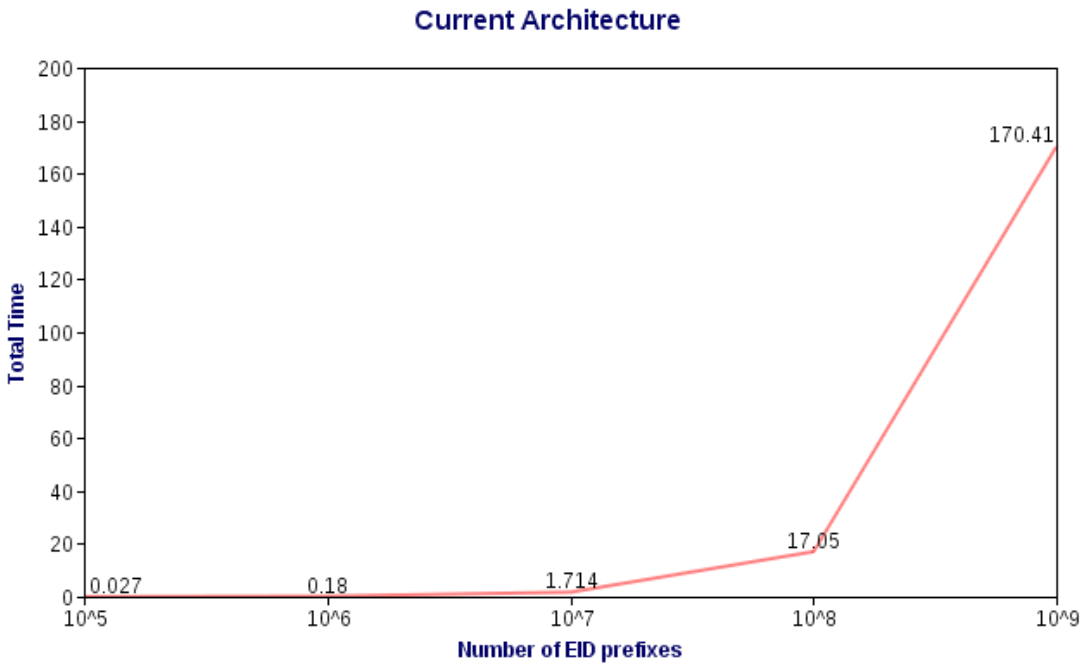

Figure 17. Plot for Time taken to transmit packet (Current Architecture)

CHAPTER 10

COMPETITIVE COMPARISION OF LISP

10.1 Six-One Router

10.1.1 Introduction

Six-One router is one the other proposed solution to alleviate the problem of routing scalability. Six-one technology requires new hardware (routers). It forwards the packet from the origin to the local six-one router where it gets forwarded to the remote six-one router and from there it gets sent to the remote host. It adds two more routers on its way which are more like gateways, so any packets that are sent out by machines behind these gateways are routed through these six-one routers [5].

10.1.2 Advantages of LISP over Six-One Router

- Six-One router needs new hardware. It needs new routers.
- Since, LISP encapsulates the existing messages; it does not require new hardware. The old routers can be configured to adapt to LISP.
- These new routers cannot distinguish whether the packet is destined for a legacy router or a new Six-One router. The use of the Six-One header option in packets of the initiating host is hence opportunistic during the first round trip. As a result of this the routing quality will suffer.
- Since no new routers are required, the chances of the routing quality to suffer are very less.
- Six-One router relies on an outside mapping resolution system. This might lead to circular dependencies between the address indirection and the mapping resolution. In order to avoid this, the transit address must be provisioned for nay edge address that may be required for mapping resolution. This has been left as a

responsibility for the mapping resolution protocol. LISP takes care of its own mapping resolution and hence this will not be a problem.

- Since, all the routers will be behind one ETR/ITR, the number of routing addresses in the routing table goes down.
- Also, the numbers of messages travelling goes down as well since the number of routing addresses are less.

10.2 Name Based Sockets.

10.2.1 Introduction

Name based sockets changes the existing address based socket system into a name based sockets system. Right now, machines and applications on machines communicate with each other using the IP Addresses. According to name based sockets, the machines will be able to communicate using the domain names instead of the IP addresses. [6]

Name based sockets will move the discovery for the impending communication from the IP Layer to the applications. This would require all connections to be Provider Independent[16].

10.2.2 Cons of Name Based Sockets

- It will be really difficult to overhaul the entire system to become name based from IP Address based.
- LISP doesn't require any changes in the system to change the way machines communicate.
- Since the communication will be bound to domain name instead of IP address, the security concerns will be further increased as the domain names can be easily faked as we see even nowadays.
- Backwards compatibility will be difficult. When a new application that uses name base sockets communicates with a legacy application, a self-generated domain

[16] http://christianvogt.mailup.net/pub/2009/vogt-2009-name-based-sockets-summary.txt

name will represent the legacy application's IP Address. This adds further overhead as there will be need for translation from the domain name to the IP Address.

- If this is implemented, since it will be based on domain names, it will be difficult to change the domain names, especially in the case where a company owing the domain names gets split.
- Deployment of LISP is much easier and smooth as compared to implementation of Name based sockets.

CHAPTER 11

ADVANTAGES AND DISADVANTAGES OF LISP

11.1 Advantages:

- All the machines will be behind the ITRs and the ETRs As a result, the routing table will be reduced considerably.
- Number of messages exchanged will go down.
- More flexibility to both user and service provider.
- No new hardware is required.
- LISP is adaptable for both IPv4 and IPv6.
- Is being developed by CISCO which is highly reputed in this field and has many other protocols deployed in the field.
- Since the host's sit behind ETR's and ITR's they can be easily monitored. If, the user moves from one location to another location, they can be easily tracked by the service provider since they sit behind their own ITR's and hence, there is less re-routing of the packages and as a result reduces the chances of the packet getting lost in transition.

11.2 Disadvantages:

- Since additional tunnel headers are prepended, the packet becomes larger and in theory can exceed the MTU of any link traversed from the ITR to the ETR. It is recommended, in Ipv4 that packets do not get fragmented as they are encapsulated by the ITR. Instead, the packet is dropped and an ICMP Too Big message is returned to the source.

- Controversy remains, however, as to whether the encapsulation overhead of map-n-encap schemes is problematic or not; opinions exist on both sides of this issue.

CHAPTER 12

CONCLUSION

After investigating on LISP and considering the current conditions of routing scalability, LISP looks like one of the best solutions to alleviate the current routing scalability problem in the core of the Internet. Also, in comparison with other solutions that are being proposed, LISP seems to be the one which is best suited. It does not require extra hardware and hence, the cost of implementation will be very less. Also, since no new hardware is required, the transition can be very smooth.

There is also more flexibility in the network. Since, the routers sit behind ETR's and ITR's, the number of messages being sent and received goes down. Also, the number of addresses that needs to be advertised is low too.

We can see from the qualitative analysis in Chapter 9, LISP will provide us with gains on many fronts. It will reduce the size of the routing table, and even though the ITRs and ETRs take a hit on processing time, LISP will do better than the current architecture.

The down side is that the overhead will increase with new headers getting appended to the messages that will impact packet processing. Also, more management will be required at the source side as the ITR's and ETR's will need to map the incoming and outgoing packages to the host's that are sitting behind them.

REFERENCES

[1] Bonaventure, O. Reconsidering the Internet Routing Architecture. (2007) Available from: http://tools.ietf.org/html/draft-bonaventure-irtf-rrg-rira-00; accessed December 2009.

[2] Farinacci, D., et al. Locator/ID Separation Protocol (LISP). (2009) Available from http://tools.ietf.org/html/draft-ietf-lisp-04; accessed January 2010.

[3] Lewis, D., et al. Interworking LISP with IPv4 and IPv6. (2009) Available from http://tools.ietf.org/html/draft-ietf-lisp-interworking-00; accessed January 2010.

[4] Lear, E. NERD: A Not-so-novel EID to RLOC Database. (2010) Available from http://tools.ietf.org/html/draft-lear-lisp-nerd-08#appendix-B; accessed March 2010.

[5] Vogt, C Six/One Router – Design and Motivation. (2008) Available from http://users.piuha.net/chvogt/pub/2008/vogt-2008-six-one-router-design.pdf; accessed December 2009.

[6] Vogt, C Name Based Sockets. (2009) Available from http://christianvogt.mailup.net/pub/2009/vogt-2009-name-based-sockets-summary.txt; accessed May 2010.

[7] Farinacci, D., et al. LISP Alternative Topology (LISP + ALT). (2009) Available from http://tools.ietf.org/html/draft-fuller-lisp-alt-05; accessed January 2010.

[8] Narten, T., et al., Routing and Addressing Problem Statement. (2007), Available from draft-narten-radir-problem-statement-01.txt; accessed December 2009.

[9] Farinacci, D., et al. LISP for Multicast Environments. (2008), Available from http://tools.ietf.org/html/draft-farinacci-lisp-multicast-01; accessed January 2010

[10] Meyer, D. The Locator ID Separation Protocol. (2008), Available from "The Internet Protocol Journal", Volume 11, No.1; accessed December 2009

[11] Brim, S., et al. LISP-CONS: A content distribution Overlay Network Service for LISP. (2008), Available from http://tools.ietf.org/html/draft-meyer-lisp-cons-04; accessed February 2010

VITA

Varun Sudhir Jain was born in Ambala Cantt, India on the 5th of September 1984. He did his schooling at Umedbhai Patel English School and completed his 10th class in the year 2000. He completed his twelfth at Patkar College of Science and Arts. He then pursued his engineering at Ramrao Adik Institute of Technology to become a Bachelor of Engineering in Electronics Engineering. He joined the Masters program in the Department of Electrical Engineering at UMKC in the Fall semester of 2007 and graduated in May 2010. He is currently working with Symantec Corporation.