IMPLEMENTATION OF LOCATION IDENTIFIER SEPARATION PROTOCOL (LISP) ROUTING PROTOCOL IN NETWORK SIMULATOR 2

A Thesis by

Prithvi Manduva

B.Tech, Progressive Engineering College, JNTU 2008

Submitted to the Department of Electrical Engineering and Computer Science
and the faculty of the Graduate School of
Wichita State University
in partial fulfillment of
the requirements for the degree of
Master of Science

May 2014

IMPLEMENTATION OF LOCATION ID SEPARATION PROTOCOL (LISP) ROUTING PROTOCOL IN NETWORK SIMULATOR 2

The following faculty members have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirement for the degree of Master of Science with a major in Computer Networking.

_____
John Watkins, Committee Chair

_____
Ravi Pendse, Committee Member

_____
Atul Rai, Committee Member

DEDICATION

To my family and friends.

ACKNOWLEDGEMENTS

I wholeheartedly thank my advisor and professor Dr. Ravi Pendse for his invaluable insight, guidance, and constant encouragement throughout not only my thesis work but also my studies at Wichita State University. I also extend my appreciation to Mr. Amarnath Jasti for his guidance, advice, and time. I thank my committee members Dr. John Watkins and Dr. Atul Rai for their precious time and suggestions. Thank you to my leads Amarnath Jasti and Vijay Ragothaman for all their support during my tenure at work. Finally, I thank my family and friends for their constant encouragement and support during my research.

ABSTRACT

The Internet, which has had an impact on almost every facet of our lives. It has grown at a rapid pace and devices connecting to in direct proportion, in turn increasing the routing tables. The Location Identifier Separation Protocol (LISP) helps to reduce the burden in both enterprise routers and Internet routers without changing the hardware, which could cost more than implementing a new routing protocol. The LISP helps in reducing Border Gateway Protocol (BGP) routes as well as keeping devices connected to the Internet or network with less downtime. Implementation of the LISP in a simulator helps network engineers learn the upcoming protocol and design, and test the network.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVATIONS

LISP	Location Identifier Separation Protocol
BGP	Border Gateway Protocol
IP	Internet Protocol
CIDR	Classless Interdomain Routing
VLSM	Variable Length Subnet Masking
EIGRP	Enhanced Interior Gateway Routing Protocol
IGRP	Interior Gateway Protocol
RIP	Routing Information Base
EID	Endpoint Identifier
RLOC	Routing Locator
ETR	Egress Tunnel Router
PA	Provider Assigned
PI	Provider Independent
ITR	Ingress Tunnel Router
AFI	Address Family Identifier
LSB	Locator-Status-Bits
ICMP	Internet Control Message Protocol
TCL	Tool Command Language
OTCL	Object oriented Tool Command Language
NAM	Network Animator
NAT	Network Address Translation

CHAPTER 1

INTRODUCTION

The Internet occupies a huge presence in today's world. With the vast amount of information on the internet, routing protocols play an important role in everyday life. An Internet protocol (IP) address is a numerical label assigned to a device that participates in a network. Each and every device must be assigned with an IP address, which consists of two parts: network address and host address. The network address helps in identifying the whole network or subnet while the host address helps in identifying the machine's connection or interface of that network. IP addresses are divided into various classes: Class A, Class B, Class C, Class D and Class E, Class D addresses are reserved for multicast work, and Class E are reserved for experimental and research work. In Class A, the first octet is the network address, in Class B the first two octets are the network address, and in Class C the first three octets are the network address. As the Internet began to grow in the late 1980's, Class C allocation became an issue; because class C addresses could increase the routing table size in the Internet core due to having three octets in its network address. To mitigate this issue the Internet Engineering Task Force introduced the concept of Classless Interdomain Routing (CIDR). CIDR helps to reduce the routing table with Variable Length Subnet Masking (VLSM), which aggregates addresses and advertises them as a single route.

Internet architecture is growing rapidly. However, today's Internet uses only the routing protocol Border Gateway Protocol (BGP), which is very slow protocol unlike other distance vector routing protocols, such as Enhanced Interior Gateway Routing Protocol (EIGRP), Interior Gateway Routing Protocol (IGRP), and Routing Information Protocol (RIP). The BGP routing table holds approximately 450,000 [1] entries, and the Forwarding Information Base (FIB) is not

getting any smaller. Massive growth of BGP routes in the routing table increases the core memory usage. The time taken up by the core routers to go through the entire table impacts the latency. Considering all these factors and the expanding Internet architecture, Cisco has proposed and developed an Internet routing protocol called the Location Identifier Separation Protocol (LISP). LISP, an open-source protocol, helps in reducing the routing table on core routers and mitigates the problem of increasing routing information base (RIB) and forwarding information base (FIB) and the BGP messages processed, with minimal changes to the hardware and software. Figure 1 shows the BGP routing table growth rate across the years.

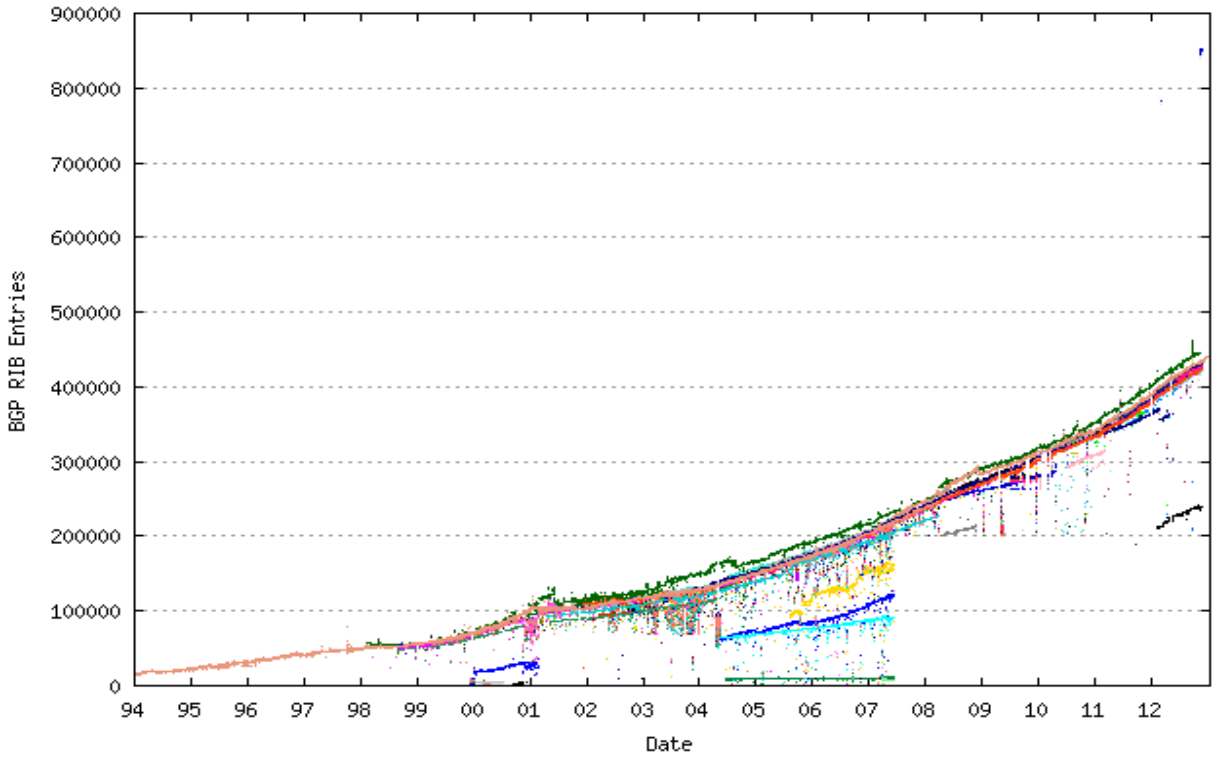

Figure 1. BGP Routing Table Growth Rate [1]

This thesis work involves study of LISP and implementation. Chapter 2 explains the LISP, new terms introduced in the protocol, how it works, the routing architecture, and the flow sequence. Presented here is the LISP protocol implementation in a Network Simulator 2 (NS2)

and how it helps in the field of networking. BGP and LISP Implementation are explained in Chapter 3 and the Simulation and Results are presented in Chapter 4.

CHAPTER 2

LOCATION IDENTIFIER SEPARATION PROTOCOL

2.1 Introduction

Currently, each and every electronic device is connected to the network or the Internet. The LISP was proposed and developed to reduce the scaling and the BGP routes on the core routers with minimal hardware and software changes. The basic proposal was based on the common a concept: locator and identifier, termed Loc/ID.

Keeping this as a constraint developers came up with a concept of Location Identifier Separation Protocol. The LISP can be used by using the IPv4 address and encapsulating the IPv6 address which would be similar to the DNS server. IPv4 addresses are diminished, and migration to IPv6 addresses is also supported by LISP. The LISP has become an open standard, using a new semantic for IP addressing. It is a network-based map and encapsulation protocol separates Internet addresses into two namespaces Endpoint Identifier (EID) and Routing Locator (RLOC). EIDs are the IP addresses assigned to the end-hosts, and RLOCs are assigned IP addresses to the devices that comprise the global routing system. End systems operate as current scenarios, and no changes are required for the host stack; the Network layer is simply "jacked up" and a new network layer is inserted below it.

2.2 Design Goals

Design goals for developing the LISP are as follows:

- No hardware and software changes to the end systems.
- Minimal changes to the current Internet architecture.
- No major router hardware changes and minimal amount of software change.

- Minimal packet loss during the endpoint identifier to routing Locator mapping.
- Optimized IP routing.
- Reduction in operational complexity.

Figure 2 shows how the network layer is jacked up in the transmission control protocol (TCP/IP) stack.

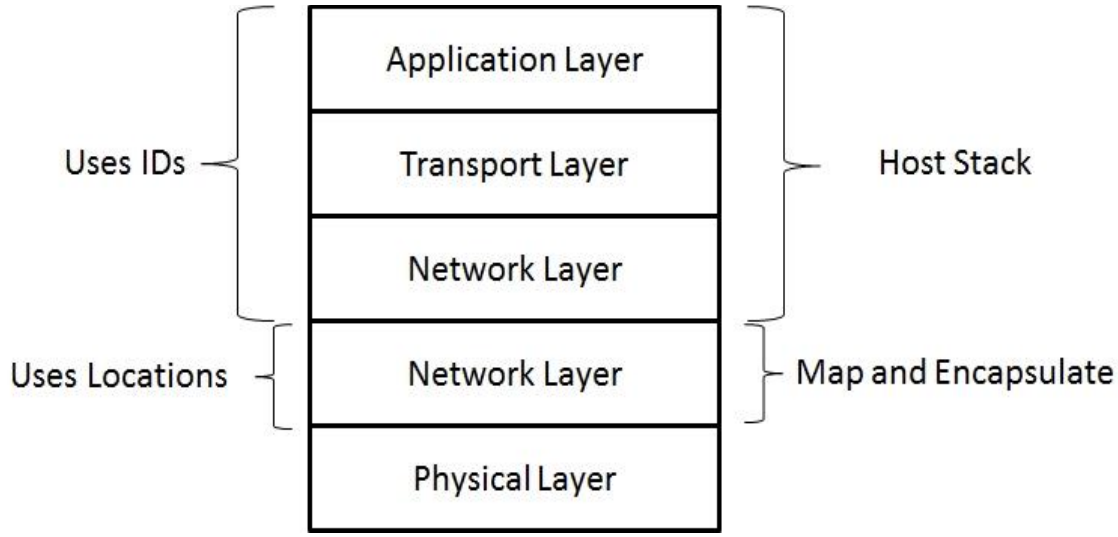

Figure 2. LISP Jack Up at Network Layer [2]

2.3 Definitions

A brief description of various terms affiliated with the LISP [3] follow:

- **Endpoint Identifier (EID)**: EID is an IP address that is used in the source and destination address fields of the LISP header. The mechanism used to obtain the destination address of the host is the same as the current processes. Source EID is assigned or obtained with a set of local host IP addresses. The EID can be used to point to another host. EID blocks are assigned in a hierarchical manner to scale the database mapping.
- **Routing Locator (RLOC)**: RLOC is the IP address of the Egress Tunnel Router (ETR). EID to RLOC mapping lookup is used to derive the RLOC address. A single EID can be

mapped to more than one RLOC. Usually RLOCs are numbered from topologically-aggregatable blocks assigned to a site at each point to which it attaches to the global Internet and the topology is defined by the provider network. RLOCs can be considered as a provider assigned addresses.

- **Provider Assigned (PA) Address**: The service provider assigns a block of addresses. Every assigned PA is a sub-block of CIDR block of the service provider and cumulates to a large block before advertising to the Internet.
- **Provider Independent (PI) Address** : PI addresses are a block of addresses assigned from a pool and the addresses are not allocated to any network. They are not aggregatable in the routing table.
- **EID Prefix**: The EID is an IPv4 or IPv6 address associated with RLOC addresses, which serves the purpose of database mapping. An address allocation authority assigns or allocates a power of two block EID's to a site. The EID prefix is divided into smaller blocks when it is required to be associated with a smaller EID prefix. The EID prefix is not a globally routable address, and vice versa, a globally routable address is not an EID prefix.
- **End System**: The end system is a device that generates the packet with a IPv4 or IPv6 address. The EID value for the destination address is supplied by the end system. An end system can be a host computer, a switch or router, or any network device.
- **Ingress Tunnel Router (ITR)**: Ingress Tunnel Routers accepts a packet with a non-LISP header attached to it. The ITR treats the inner header as an EID, and performs EID-to-RLOC mapping, and then the router appends the RLOC address, which is routable over the Internet in the source field, resulting in mapping to the destination address field.

- **TE-ITR**: The aim of the TE-ITR deployed in a service provider network, is to append an additional LISP header.
- **xTR**: The term xTR refers to an ITR or ETR, where data flow is not in discussion.
- **EID-to-RLOC Cache**: The EID-to-RLOC cache is responsible for storing, tracking, validating the, EID-to-RLOC mapping, and timing out the mapping. It is a dynamic, short term cache and is local to its respective router (ITR) and relatively small compared to the database.
- **EID-to-RLOC Database**: The globally distributed database has all known EID prefixes to RLOC mappings. Every ETR has a piece of database for EID prefixes. These prefixes are mapped to the routable global IP address.
- **Recursive Tunneling**: Recursive tunneling occurs when a packet has more than one LISP header due to traffic engineering or when rerouting occurs.
- **LISP Header**: This term is used to describe the IP headers appended to the IP packet at the network layer. The LISP header is encapsulated at the ITR and de-capsulated at the ETR.
- **Re-encapsulating Tunnel**: When a packet is received with more than one LISP header it needs to be redirected or directed to a completely new RLOC location. The ETR de-capsulates the outer header and encapsulates a new header to divert the packet to a new destination.
- **Address Family Identifier (AFI)**: AFI helps indicate the type of address in the encapsulated packet either IPv4 or IPv6.

- **Negative Mapping Entry**: This is an EID-to-RLOC entry where an EID prefix is saved in the database without a RLOC. It helps differentiate a prefix from a non-LISP site, this is not explicitly in the mapping database.
- **Data Probe**: A probe message or request (Map-Request) is sent to a Map-Reply message from an ETR
- **LISP Site**: LISP sites are routers with a single administrator in an edge network.
- **Locator-Status-Bits (LSBs)**: These Bits are present in a LISP header. These bits indicate the router status of ITRs and ETRs.

2.4 Overview And Tunneling Details Of LISP

The IP address of the end system (i.e EID's) is used to send a packet from the source to a destination. The LISP does not change the working of end systems, and there are no hardware changes required for the existing system. As in the current system, the LISP uses the destination address (EID) to route the packet. LISP encapsulated packets are RLOC's and are routable IP addresses.

The LISP design introduces tunnel-routers, which encapsulate or append LISP headers when the packet is sent and decapsulate LISP headers at the receiving end router. LISP headers have RLOCs. The ITR performs the EID-to-RLOC lookups to determine the routing path of the ETR, which has the RLOC as an IP address.

2.4.1 Rules Governing LISP

Rules governing the LISP [3] including the following:

- End hosts only send IP addresses that are EIDs, they do not map EIDs to RLOCs but rather assume that LISP routers deliver the packets to the appropriate destinations.
- EID is assigned to host.

- The RLOC IP address is assigned to the routers, preferably topologically oriented addresses from providers CIDR blocks.
- When the router generates a source packet, it may use a source address either to an EID or an RLOC address. An EID prefix which helps in identifying a router should never be used as an RLOC because an EID is routable only locally within the scope of that site. A good example of such hybrid behavior would be a BGP configuration where the router uses its local EID to terminate iBGP sessions and its RLOC to handle eBGP sessions.
- In the absence of EID-to-RLOC mapping, EIDs are not used for host-to-host communication but are used locally for intra-site communication.
- EID prefixes are designated hierarchically in an optimized manner for administrative convenience and to scale EID-to-RLOC mapping.
- EIDs are structured in such a way that they are suitable for local routing in an autonomous system.
- No more than two LISP headers can be attached to a packet.

2.4.2 Working with Example and Packet Flow

Figure 3 shows how and where the LISP protocol can be implemented and how the packet is traversed. Packet flow is as follows:

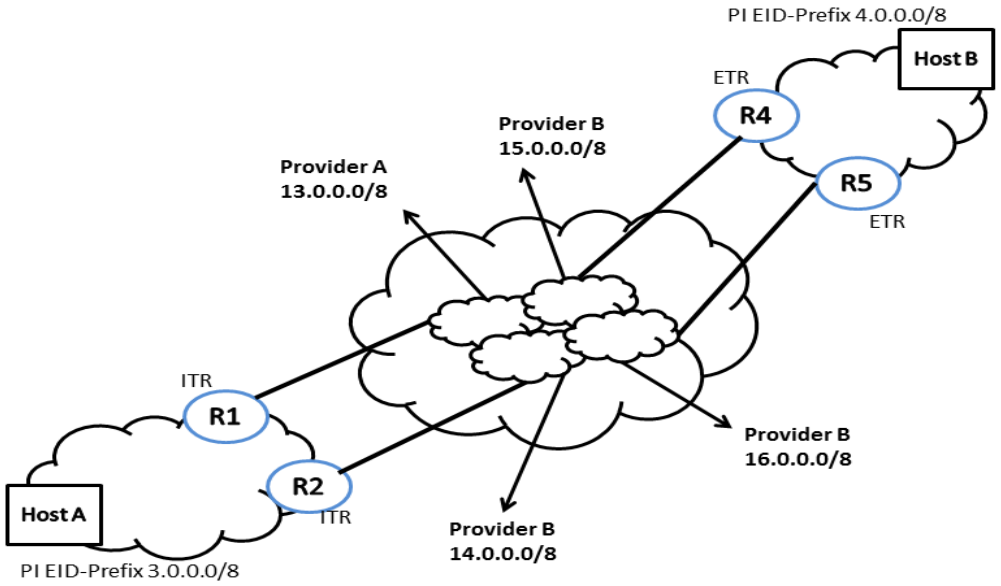

Figure 3. LISP Packet Transmission

- Host A sends a packet to Host B. When Host A wants to do a TCP connection with Host B, it will look in the domain name servers (DNS) for Host B. After receiving the DNS result , it will use the address as the destination EID and the local address as the source EID.

- Router R1 and R2 are configured as ITR. The ITR maps the EID-to-RLOC of the ETR, which is R4 or R5 and the ETR at the destination site. Whenever the next packet from the source to destination ETR's and RLOC as the destination address in the LISP header.

- The ETR will receive the packet and forward the packet to the end system. To eliminate the map lookup in the reverse direction, the ETR has the cache entry for future reference.

2.4.3 Tunneling Details

When tunneling occurs the IP packets are encapsulated or prepended with the tunnel headers and thus the packet becomes larger than the maximum transmission unit (MTU). In LISP packets are not fragmented because they are encapsulated by the ITR. They would be dropped at the router itself. The maximum MTU that an ISP router can allow would be 4770 bytes [3]. Figures 4 and 5 shows the LISP IPv4 header format and LISP IPv6 header format followed by an explanation of the fields.

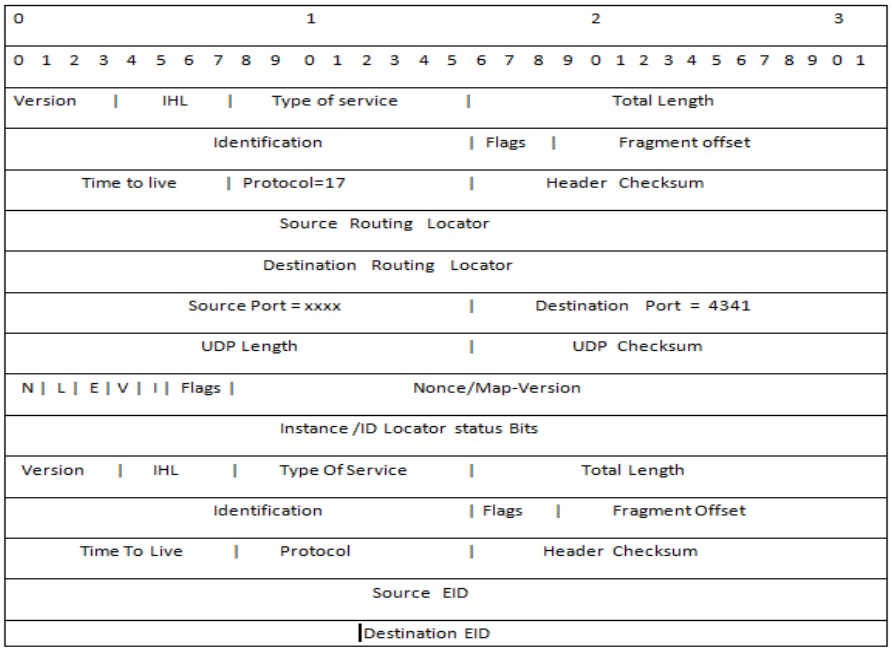

Figure 4. LISP IPv4 Header Format [3]

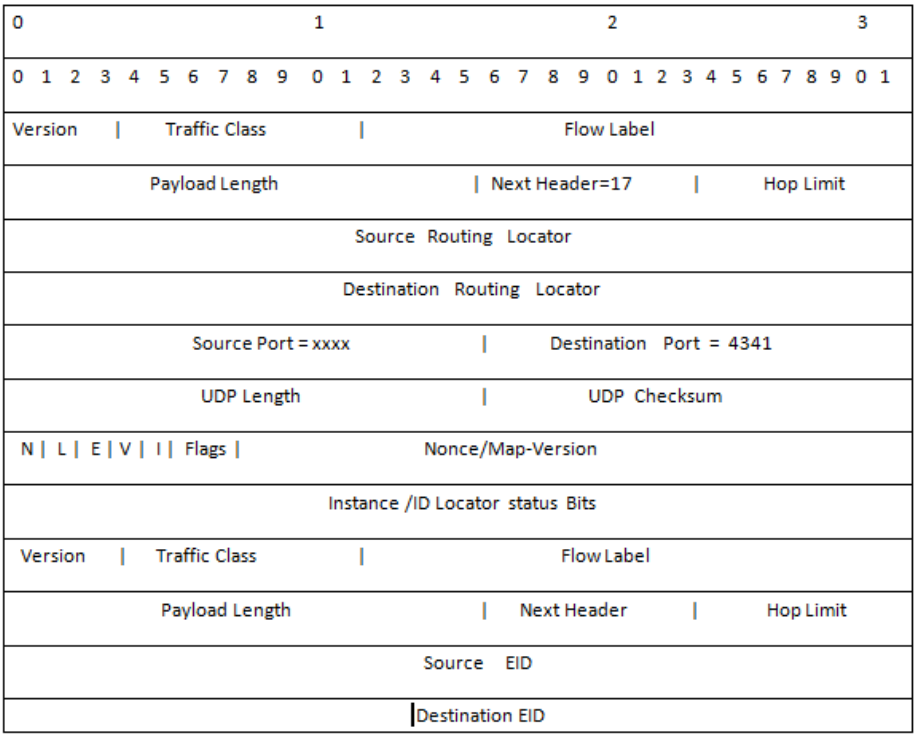

Figure 5. LISP IPv6 Header Format [3]

2.4.4 Tunnel Header Field Description

The tunnel header field description is as follows:

- **IH Header**: The IH header is the IP packet header from source host. The source address and destination address in this header are the EIDs.

- **OH Header**: This header has the information encapsulated by the ITR. The RLOCs obtained from the EID-to-RLOC database mapping cache would be in the OH header.

- **User Datagram Protocol (UDP) packet**: A source port is assigned while encapsulating the packet selected by the ITR. The destination port should be port 4341.

- **UDP Checksum**: ETR must ignore the checksum and must not recalculate the checksum computation. This field should be transmitted as 0 and ignored at the ETR. A network

address translation (NAT) device can recalculate the checksum and assign a new UDP checksum which should be a non-zero value.

- **UDP Length**: The encapsulated packet has an inner header, UDP and LISP header. The UDP header is 8 bytes and the LISP header is 8 bytes when the loc-reach-bit header extension is not being used.
- **S**: S refers to the Solicit-Map-Request bit.
- **Locator Reach Bits**: These bits are set to indicate ETR reachability for the locator in the source network. An ordinal value from 0 to n-1 is given to each RLOC in the map reply. A bit is set to 1, indicates that the RLOC associated with ETR is reachable. When a site has multiple EID prefixes there should be multiple mapping.
- **LISP Nonce**: A 32-bit random value generated by the ITR, is used for route returnability when xTRs transmits and receives the encapsulated packet with an SMR bit set, Data-Probe, Map-Request or Map-Reply.
- If a recursive tunnel occurs, then the time to live (TTL) field of the OH header is copied from the IH header, and type of service (TOS) field for the IPv6 packet should be taken from the IH header.
- When re-encapsulation occurs, the new OH header TTL should be copied from the OH header TTL field, which would be stripping the OH header. The new TOS field is copied from the stripped OH header. The TTL field helps to prevent looping when there is misconfiguration. It also preserves hops that packet needs to reach the destination.

2.4.5 LISP Control Packet Format

The LISP uses UDP based messages for control packets such as Map-Reply and Map-Request. If a Map-Request message occurs, then the UDP port is allocated by the source, and the destination is set to 4342. A Map-Reply message is sent with the source as 4342, and the destination port is copied from the received Map-Request message [3] or the authoritative data packet. The control packet format for IPv4 and IPv6 address are shown Figure 6 and 7, respectively. These figures also show the LISP IPv4 and IPv6 control packet formats.

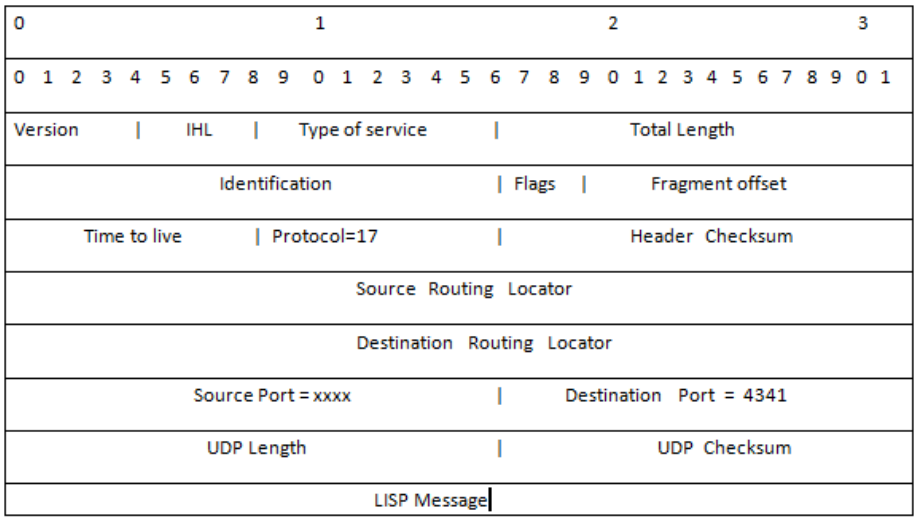

Figure 6. IPv4 Control Packet Format

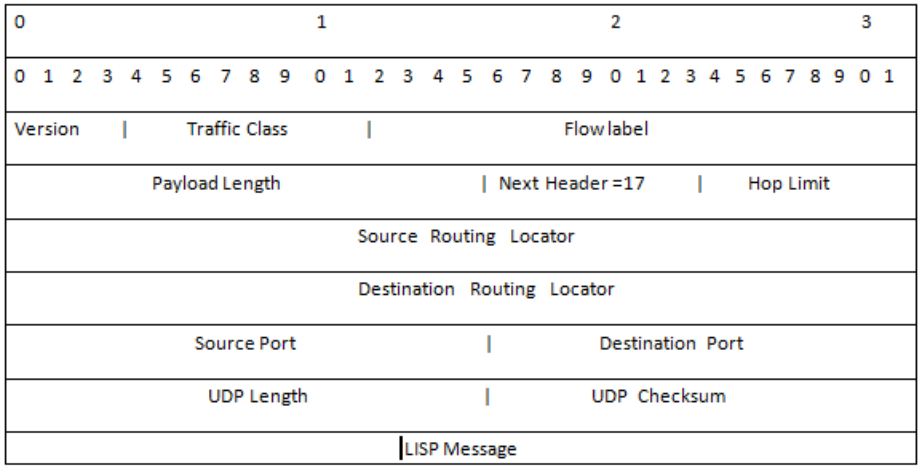

Figure 7. IPv6 Control Packet Format

2.4.6 Map Request Message Format

Figure 8 shows the Map-Request message format followed by an explanation of the fields.

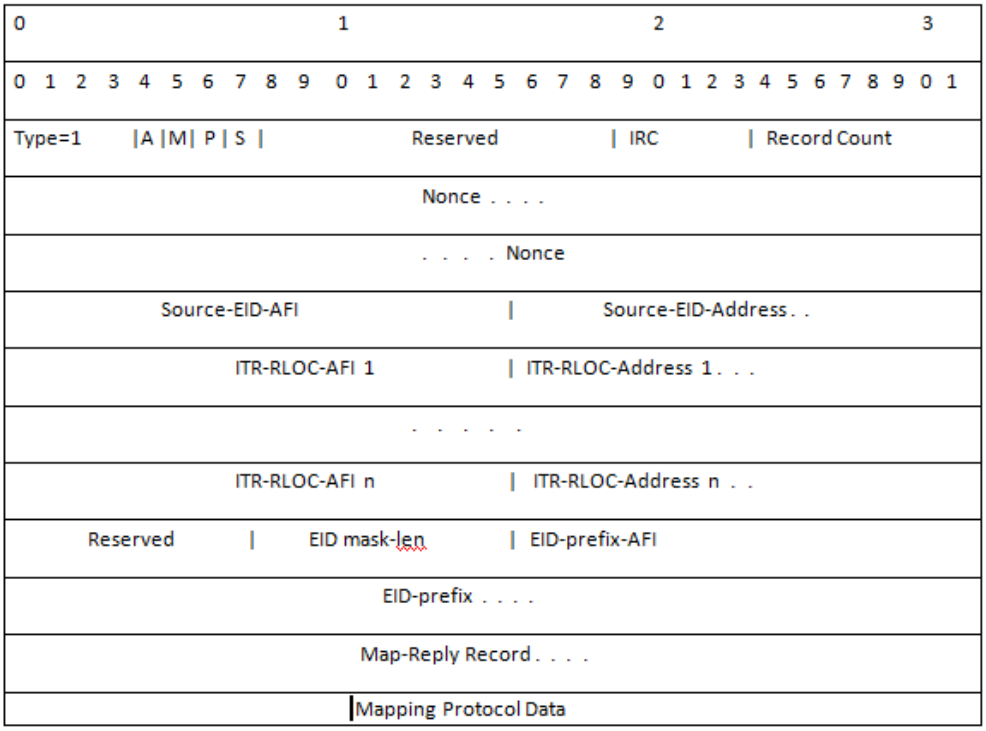

Figure 8. Map Request Message Format

- **S**: Solicit Map Request (SMR) bit.
- **Locator Reach Bit**: This bit is set to 0 before transmitting and is ignored at the receivers end. Locator Reach Bit cannot be used to indicate reachability because Map-Request message does not have the EID prefix. The receiver would not know what to map it with. Mapping data is informed in the Map-Reply record. The R bit per locator entry in the EID prefix is recorded to denote reachability.
- **Nonce**: This four byte random value is generated by the sender.
- **A**: This is an authoritative bit.

- **R**: R is set when a Map-Reply record segment. The record segment is then sent along with map request.
- **Reserved**: Here the value is set to 0 before sending and ignored at the end host.
- **Record Count**: This is a count of records in the request message. A record is composed of a portion of the packet, labeled as 'Rec' above and occurs the number of times equal to the record count.
- **Source-EID-AFI**: This is the address family of the "Source EID Address" field.
- **ITR-AFI**: This is the address family of the "Originating EID Address" field.
- **Source EID Address**: The EID of the source host initiates this Map-Request message.
- **EID Mask-len**: This is the EID prefix mask length.
- **EID-AFI**: This is the address family of the EID prefix.
- **EID-Prefix**: This is a four-bytes value if an IPv4 address is used and a sixteen-byte value if an IPv6 address is used. When a data packet reaches the destination, the EID prefix is set to the destination address if there is no map entry when a Map-Request message is sent by the ITR. The EID mask-len is set to 32 for IPv4 addresses and 128 for IPv6 addresses. When a cached map entry is being queued by an xTR EID, the prefix mask length of the map request would be the same as the EID prefix returned from the site, which would be in Map-Reply message.
- **Map-Reply Record**: This field is the size of the record field in the Map-Reply message format. It has the record of EID-to-RLOC mapping entries with respect to the source EID, which helps the ETR to cache the Map-Request message if it chooses to do so.

2.4.7 Map-Reply Message Packet Format

Figure 9 shows the LISP Map-Reply message format followed by an explanation of the fields.

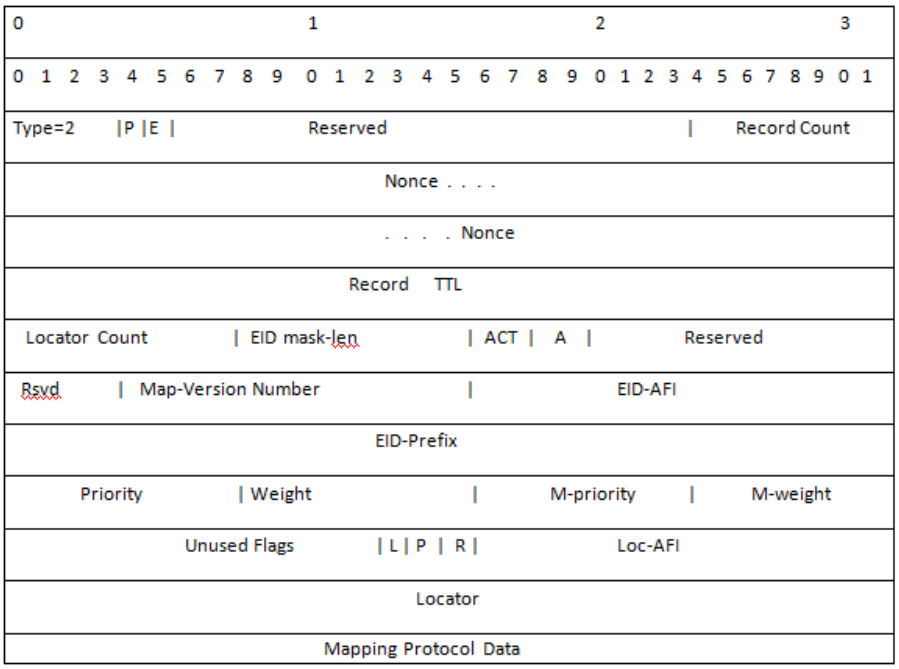

Figure 9. Map Reply Message Format

- **X**: This is set to 0 while sending and is ignored on the receipt.
- **Locator Reach Bits**: This filed is 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**: This field is a 4 byte value set in a data probe packet or it is copied from the Map request message received. Type is set to 2 for Map-Reply messages.
- **Reserved**: This field is set to 0 on transmission and ignored on receipt.

- **Reserved Count**: This field represents the number of records in the reply message. Record is a part of message labeled ‘Record’. It occurs the number of times equal to the number of record count.
- **Record TTL**: This field tells the end host how many minutes to store the Map-Reply in the cache.
- **Locator Count**: The number of locator entries. Zero indicates that there is no EID prefix in the cache.
- **EID mask len**: The mask length for the EID prefix.
- **A**: The authoritative bit set by ETR.
- **EID-AFI**: The address family of the EID prefix.
- **EID-Prefix**: 4 bytes and 16 bytes of an IPv4 and IPv6 address family respectively.
- **Priority**: A unicast priority is set to each RLOC, lower priority values are preferred. If the priority is set to 255, then that RLOC must not be used for unicast.
- **Weight**: If two or more RLOCs have same priority, the weight field helps to balance the unicast traffic between them. A non-zero weight value is assigned to any RLOC. All RLOCs should use non-zero values to which the sum of them equal to 100. If an RLOC has zero as the value, then the receiver will decide how to split the load.
- **M Priority**: A multicast priority is assigned to each RLOC by an ETR to select an ITR for building a multicast distribution tree. RLOC must not be used if the value is 255.

- **M Weight**: If priorities are similar, M weight helps to decide how to balance building multicast distribution tree over the multiple ITRs. The weight is encoded as a percentage of the total number of trees built to the source site identified by the EID-prefix.
- **Unused Flags**: This flag is set to zero when sending and ignored at the receiver end.
- **R**: With respect to the senders perspective this bit is set to indicate that the locator is reachable. The R bit must match the Locator Reach bit if there is single mapping in the record. If there are multiple, then it is set to zero.
- **Locator**: This field is an address assigned to an ETR, which acts like a proxy for the EID prefix. An RLOC address should not be a link local multicast address or multicast address. A destination RLOC can be a multicast address if it is mapped to a multicast destination EID.
- **Mapping Protocol Data**: This field is used if a UDP packet has space. It is an optional field.

When a Map-Rrequest is received, a Map-Reply message must be sent with respect to the requested RLOC with an EID prefix. RLOCs are the routable IP addresses which means they are not reachable. Replies for an EID prefix must be sent not more than once per second to the requesting router. Aggregating EID addresses into EID prefixes will allow one Map-Reply message to satisfy a mapping for the EID address in the prefix range, which reduce the Map-Request messages and improves scalability [3].

2.4.8 Routing Locator Selection

Both the server and the client need to have control over the selection of RLOCs in order to communicate with each other. This is attained by manipulating priority and weights in the

EID-to-RLOC Map-Reply messages. RLOC information may be extracted from the received tunneled packets or EID-to-RLOC Map-Request messages.

Different scenarios for choosing the RLOCs and the controls that are available are as follows:

- The client uses one RLOC received from the server end that sends the RLOCs. RLOC selection is completely controlled by the server.
- The server returns the list of RLOCs with the same best priority. The client uses the subset received from the server. The server will be controlling the subset list and load splitting. If the subset list is unreachable, then the client uses RLOC outside the subset list. If the RLOC is unreachable, then the client has the option of selecting alternatives to the server side subset list.
- The client chooses how the traffic load is to be spread, if the server sets the weight to zero for the RLOC across the subset list. Control is shared between the client who determines the load distribution and the server determining who determines the list.
- If a Map-Request is not sent from either side, and the server does not send the request, then client has responsibility for bidirectional RLOC reachability and preferablility.
- Reachability status is not providedto the locator by the Map-Reply message and database mapping service.

2.4.9 Routing Locator Reachability

A locator is determined to be reachable, unreachable, or has become unreachable in the following instances:

- Where there is a Loc Reach Bit in the LISP header, which is provided by an ITR.
- When an Internet Control Message Protocol (ICMP) network or host has unreachable messages.

- When there is no prefix matching a locator address from the BGP routing information base of a BGP enabled ITR.
- When an ICMP port unreachable message sent by a host.
- When a Map-Reply message is sent from an ETR in response to a sent Map-Request message.
- When receiving packets are encapsulated by the ITR assigned to the locator address.

CHAPTER 3

BGP AND LISP IMPLEMENTATION

3.1 Introduction

In the present scenario, the BGP is the only exterior routing protocol on which every one depends. The BGP has attributes that decide how to choose the best route to install in the routing table. There are thirteen different attributes.

- Highest weight
- Path with highest local preference
- Path that is locally originated
- Path with the shortest autonomous system path
- Path with lowest origin type
- Path with lowest multi exit discriminator
- Path with eBGP over iBGP
- Multiple path installation required in the routing table
- Paths received first are preferred, if two paths are received.
- Install the route that comes from the BGP router with the lowest router ID
- Path with minimum cluster list length is preferred when the originator or router ID are the same for multiple paths.
- Path which comes from the lowest neighbor address.

These thirteen attributes decide which route is to be installed in the BGP route table. In the current scenario the BGP router's routing information base (RIB) is increasing exponentially. The exponential increase of RIB information impacts the router processor utilization, processor memory, latency etc. Whenever a packet is received by the BGP router, it needs to check the

entire routing table for the information. This, in turn, causes a delay in the network and because the memory utilization goes too high due to the fact that the packet needs to be saved for an interval of time.

Assume that there are N route in a BGP router and a packet is received by the router. The BGP router takes T milliseconds to check whether the packet needs to go to the first route in the route table. If the packet needs to go to the last route of the routing table, then the total time taken to check it would be (N x T). So if there are 10,000 routes in the routing table and the packet needs to go to the last route the total time taken would be 10,000 x T milliseconds. If the LISP is implemented, assume that for this case the router has 10,000 routes and the packet needs to be sent to the end host and the end host is running LISP. The router checks its map cache for the EID-to-RLOC mapping, and if the information is not found, then the router sends an EID to RLOC mapping request to the map server (MS) or map resolver (MR) to receive the mapping and saves it to the cache. This process takes less time when compared to checking all routes in the route table.

Assume the time taken for EID-to-RLOC mapping is T_R milliseconds and the time taken to check the mapping in the map cache is T_M milliseconds. Therefore, the total time taken for the process when the map cache does not have the mapping is (T_M + T_R) milliseconds. If another packet needs to be sent to the same end host, then the total time taken to send the packet would be T_M milliseconds, whereas in the router with the BGP running, the packet takes T milliseconds for each packet. Figure 10 shows a WAN network with the BGP and the LISP running.

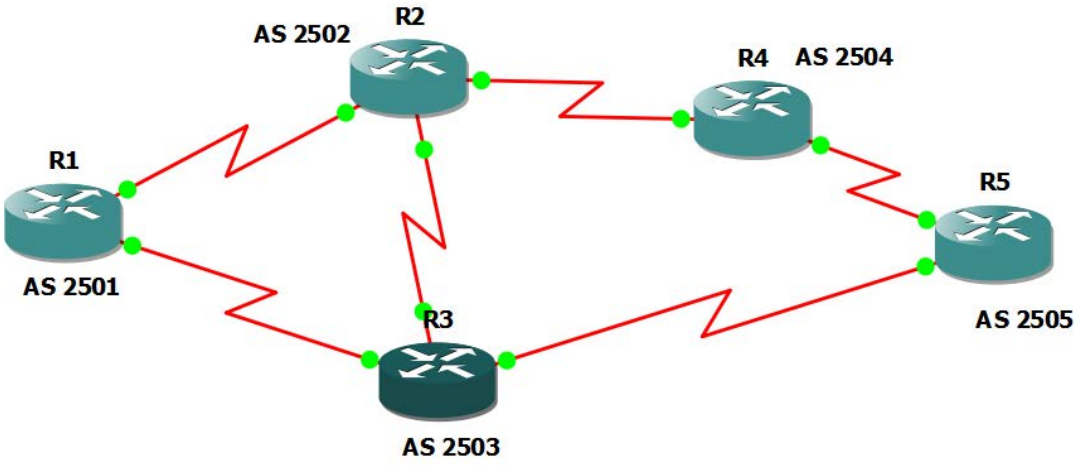

Figure 10. WAN Network with the BGP and the LISP

From the above assumptions if R1 has to send a packet to R5,then the BGP has two paths R1-R2-R4-R5 and R1-R3-R5, but the second path is installed in the R1 route because it has low autonomous system (AS) paths to traverse. If R1 receives a request to send some data to R5, then time taken for R1 to send the data would be 1 milli second, if R1 has only one route in the route table, but if it has 10,000 routes in the route table, then it takes more than 1ms to send the packet. If the LISP is configured for R1 and R5, then when a packet is received for R5 at R1, the router checks for EID-to-RLOC mapping in the map cache. If the mapping is there, then it forwards the packet or sends a Map-Request message to the MS or MR, and saves the EID to RLOC mapping in the map cache and forwards the packet. The time taken for this process is less, compared to the router running the BGP. Practical implementation is discussed in the chapter 4. Simulation and Results.

3.2 LISP Implementation in NS2 Simulator

A flow chart can simplify the working and understanding of the LISP implementation. Figure 11 shows how a packet must be forwarded or encapsulated, depending on the request received by the router.

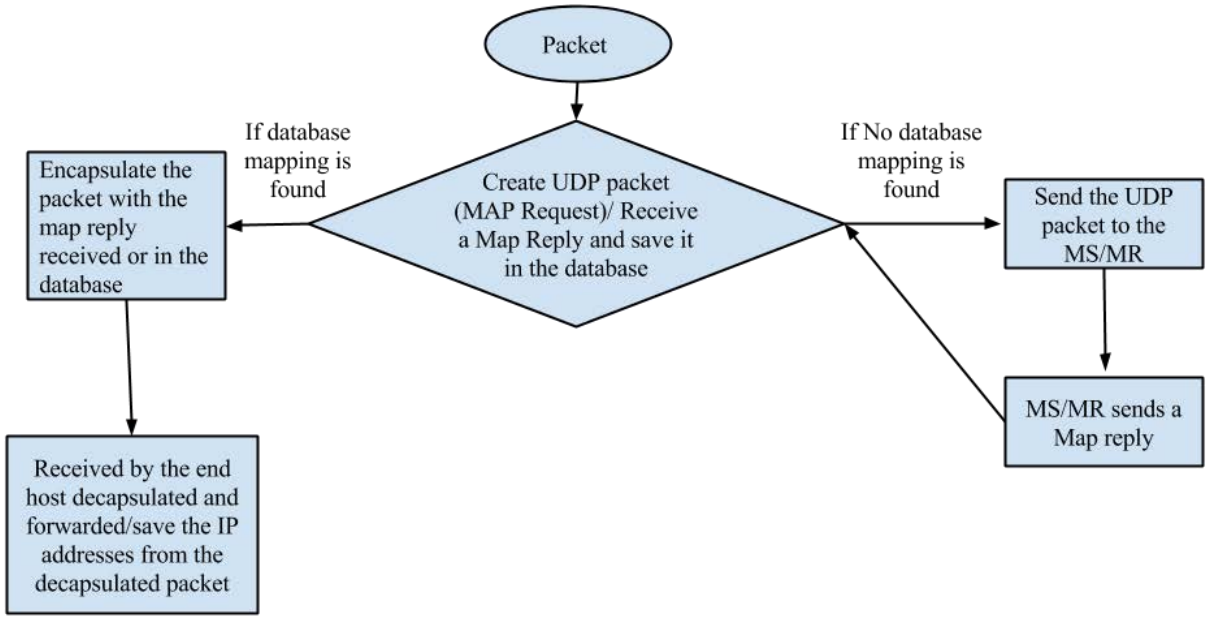

Figure 11. LISP Flow chart

Assuming all packets received by the router are LISP addresses, then when a packet is received by the router, it checks for database mapping and a map-cache. If we have map-cache entries, then the router encapsulates the packet with a new header and forwards it to the end host where the packet is de-capsulated and saves the IP addresses from the de-capsulated packet in a table and forwards the de-capsulated packet to the host. If there is no entry in the map-cache, then the router sends a UDP Map-request message to the map server or map resolver, and the entry to the map-cache then encapsulates the packet and forward it to the address in the map-cache entry.

The Lisp.h file defines the declaration of the header format of LISP. Lisp.cc file will have the constructor and will process the packet. The Map-cache.h and map-cache.cc files handle the

map-cache entry, checking the address for availability, and if the address is not available , the value is entered.

CHAPTER 4

SIMULATION AND RESULTS

4.1 Introduction

Many electronic technologies have been invented to aid the process of exchanging information in an efficient and creative way. Complex systems like computer networks are layered to ease implementation and design flexibility [5]. These layers have virtual links between the homologous layers in the two communicating nodes. To create a complex network a straightforward mathematical formulation is not possible. In these cases, the simulation approach is preferred over the analytical approach. Simulation is a process of a flow of network entities.

Network Simulator 2 is an object oriented, discrete event simulator developed at the University of California, Berkley. It is built with the a C++ core and the simulation objects are linked to shadow objects with Object Oriented Tool Command Language (OTcl). Scripts that run the simulation are written in OTcl an open source tool which can run on almost all operating systems platforms, such as Windows, Linux, etc. To increase the efficiency of network simulator, the data path implementation is separate from the control path implementation. To reduce event and packet processing time, network components and event schedulers objects are written and compiled using C++ [6]. Compiled outputs are made accessible to the OTcl with OTcl linkage which creates an OTcl object for each C++ object compiled i.e., controls of C++ objects are given to the OTcl object.

Two kinds of classes can be derived: one class that can be accessed by the OTcl domain only, for example, the Nsobject, Packet queue, Routing module etc. and the other class which is a stand-alone, like the handler module. A handler specifies an action with an event. Therefore, derived classes are responsible for providing the implementation of function handles like

NSobject receiving the incoming packet and the class QueueHandler invoking the function resume associated with the queue object. Figure 12 illustrates the directory hierarchy of the NS2 simulator.

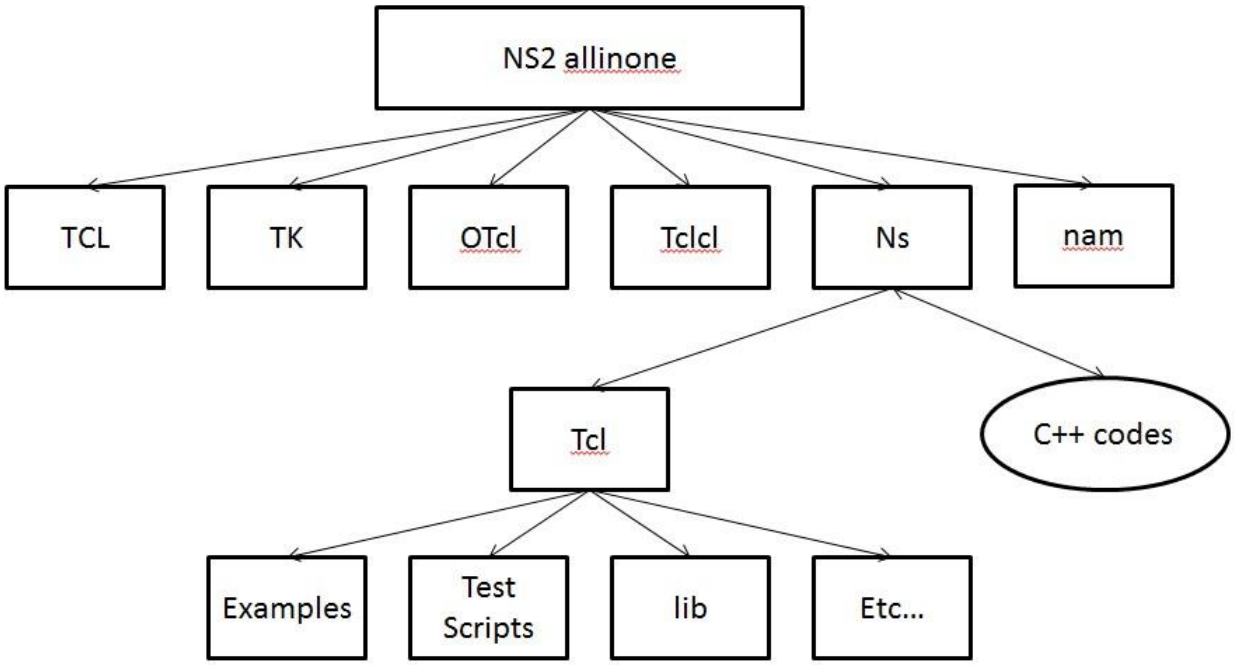

Figure 12. NS2 Directory Hierarchy

The NS2 helps run simulations with C++ codes, which are the base to the simulator. The NS2 directory is where the packet header and routing protocols are defined and how they must behave on packet arrival. The objects created here are handled by the Tool Command Language (tcl), which is a sub directory of NS. The Tcl directory has libraries to run the script and handle the objects created, and example directories have scripts test the working simulator. Test directories check the integrity and working of the simulator. The NS2 Directory Hierarchy helps a developer understand and implement new protocols that are needed. Network animator (NAM) directories help the simulator to create the network animator for visualization of the simulator.

4.2 Simulation Scenario

Figure 13 shows the simulation scenario in the NS2 simulator.

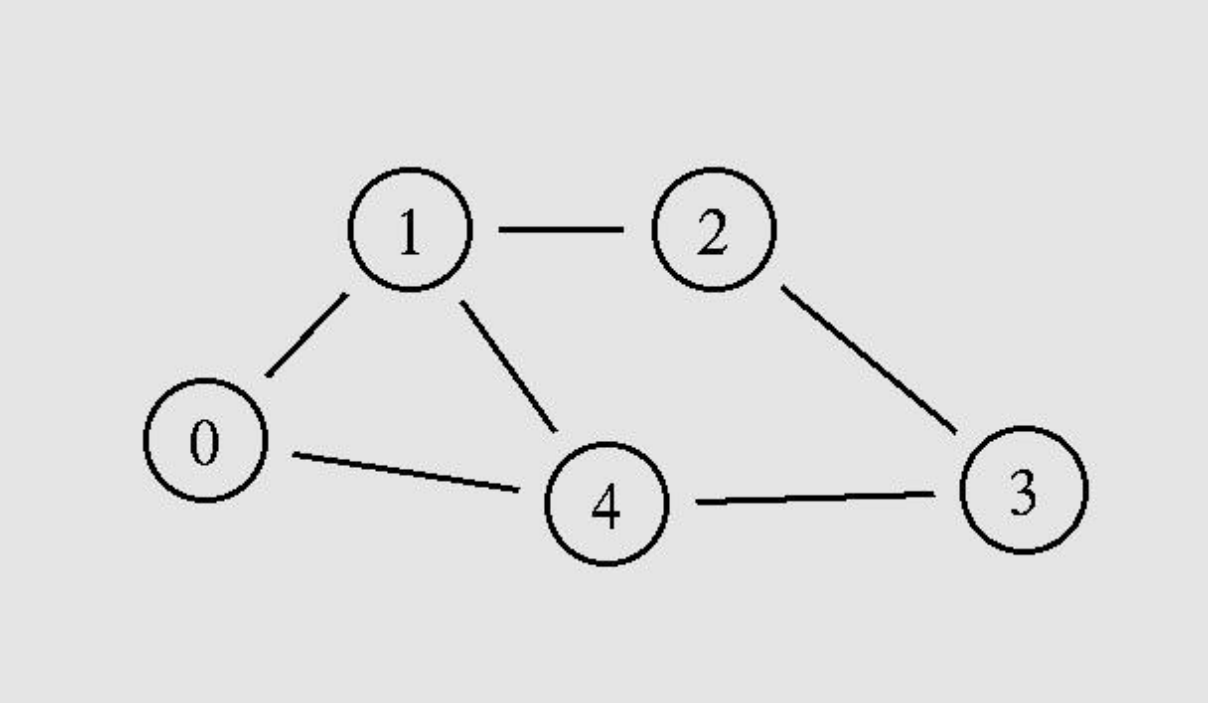

Figure 13. Simulation Scenario

4.3 Goals of Simulation

The following are the goals of the simulation:

- Simulate a real-world scenario network diagram with the BGP.
- Implement the LISP using the same network scenario.
- Use different number of routing tables to test the processor memory utilization, the average time taken to ping in both scenarios.
- Compare results to determine whether the LISP is advantageous to use or not.

In the Figure 13, all routers are running the BGP with different autonomous systems to replicate the real-time topology. They have formed neighbors with each other and exchanged routing tables. The routing table was increased from a scale of 250 to 2,000 routes. Figure 14 shows that with the increase in the routing table the processor memory utilization on each router increases as BGP routes increase. The standard deviation observed for BGP for processor memory utilization for BGP is 500 bytes and for the LISP is 50 bytes .

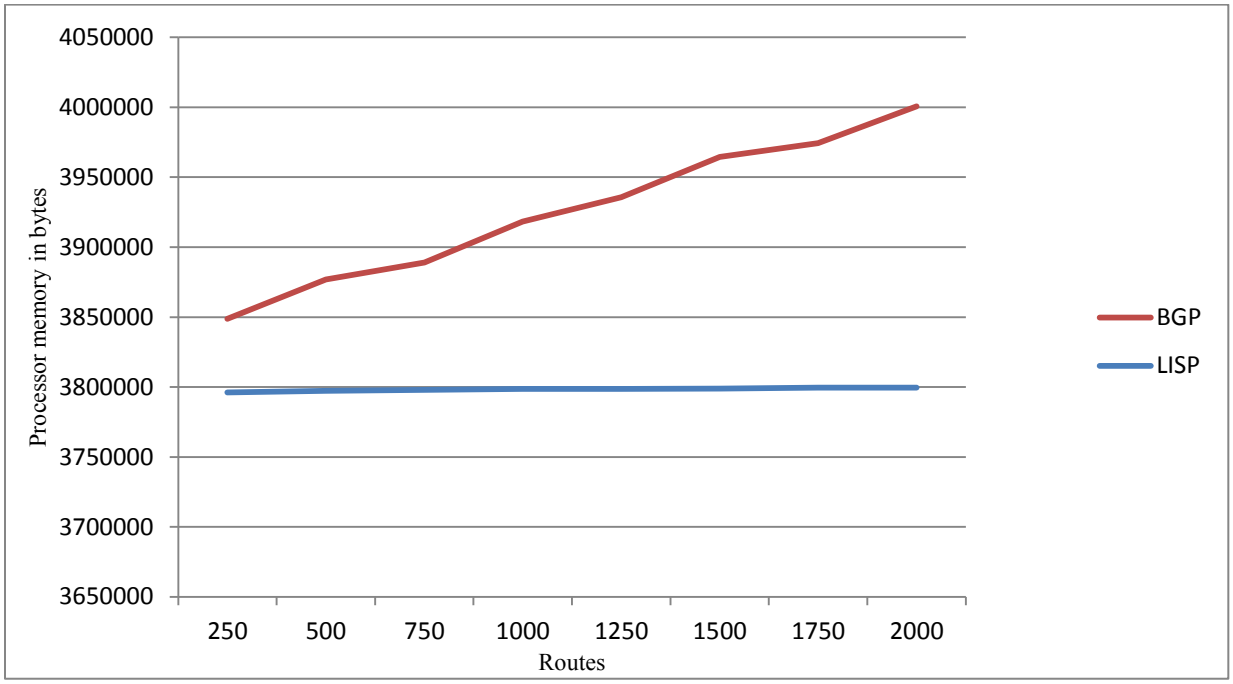

Figure 14. Processor Memory Utilization

When the LISP is implemented in the same scenario the processor memory utilization is low compared to the BGP. This happens because, unlike the BGP, the LISP does not require information about all the routes, rather, it has a consolidated or classful boundary address. When a destination address with LISP enters the router running LISP, the router sends a Map-Request message, which is a UDP packet, to the map server. The received Map-Reply message is received it is used to update information in the map cache, and encapsulate the packet and the network address with EID and RLOC, and forward the packet. Figure 15 shows when all routers with the BGP have a routing table with 2,000 routes and when a ping is sent in increments of every 100 packets.

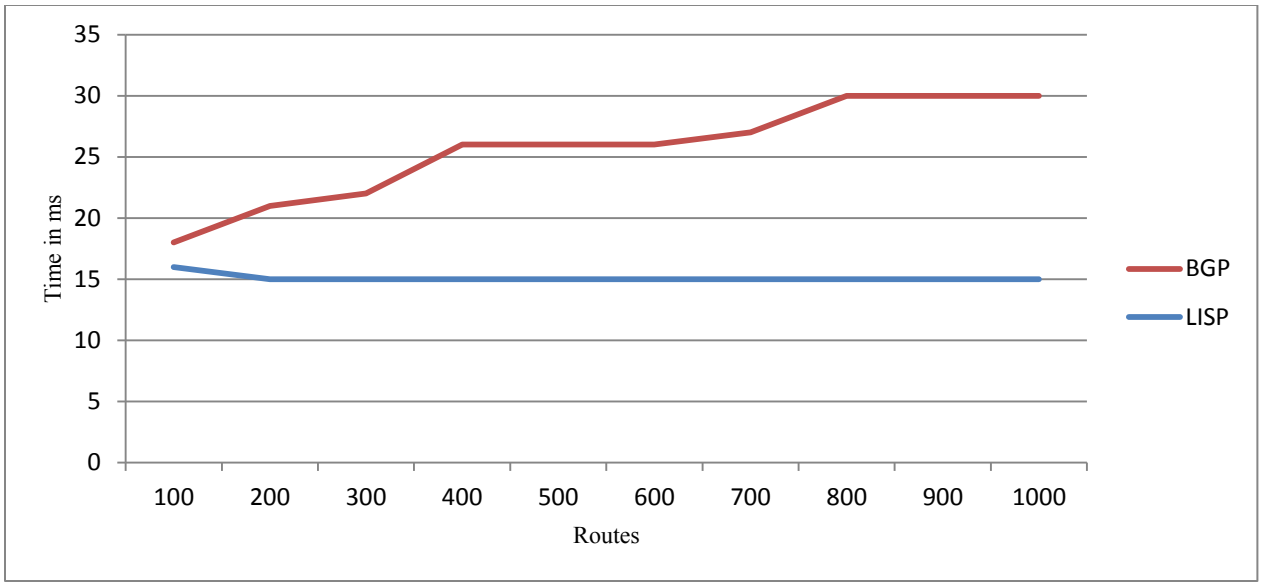

Figure 15. Ping Test

The average time taken by the ping increases with the ping packet, because each processor need to check its routing table to verify the destination address and forward the packet. Whereas, when the LISP is run in the router, the packet initially needs to check whether the IP address has a database mapping. If it does, then it sends the UDP packet to the map server and receives the Map-Reply message with the EID and RLOC. The routers then encapsulates the packet with the new header and forwards it. The graph in Figure 15 shows that initially the time taken by the LISP packet is high and remains constant even when the ping packet increases. The standard deviation observed for BGP for processor. The standard deviation observed for BGP for processor memory utilization for BGP is 0.25ms and for the LISP is 0.01ms.

4.4 Conclusion and Future work

Results from the simulation prove that implementation of the LISP improves the performance of BGP routers without any hardware changes, which could incur more cost. LISP implementation in the NS2 can be further improved for implementing the LISP with NAT and for mobile nodes in wireless networks.

REFERENCES

REFERENCES

[1] BGP Routing Table Analysis Reports, http://bgp.potaroo.net/; [accessed March 2013].

[2] Meyer, D., The LISP Identifier Seperation Protocol (LISP) http://www.cisco.com/; [accessed March 2013].

[3] Farinacci, D., et al. Locator/ID Separation Protocol (LISP), (2009), available from http://tools.ietf.org/html/draft-ietf-lisp-04 ; [accessed March 2013].

[4] Farinacci, D., et al. Locator/ID Separation Protocol (LISP). (2009) available from http://tools.ietf.org/html/draft-ietf-lisp-23; [accessed March 2013].

[5] Issariyakul, T., and Ekram , H., 2008, *Introduction to Network Simulator NS2*, Springer Science and Business Media, LLC, Chapter 1.

[6] Issariyakul, T., and Ekram , H., *Introduction to Network Simulator NS2*, Springer Science and Business Media, LLC, Chapters 2 and 3.

[7] Farinacci, D., and Fuller, V., "LISP Map Server," draft-ietf-lisp-ms-06 (work in progress), October 2010.

[8] Narten, T., et al., Routing and Addressing Problem Statement, (2007), available from draft-narten-radir-problem-statement-01.txt; [accessed March 2013].

[9] Meyer, D., The Locator ID Separation Protocol, (2008), available from The Internet Protocol Journal, Vol. 11, No. 1; available from http://www.cisco.com; [accessed December 2011].

[10] Farinacci, D., Lewis, D., Meyer, D., and Fuller, V., The Locator/ID Separation Protocol (LISP), available from https://tools.ietf.org/html/draft-ietf-lisp-22; [accessed March 2013].

[11] Lewis, D., et al. Interworking LISP with IPv4 and IPv6. (2009), available from http://tools.ietf.org/html/draft-ietf-lisp-internetwroking-00; [accessed March 2013].