Internet Engineering Task Force (IETF)
Request for Comments: 6830
Category: Experimental
ISSN: 2070-1721

D. Farinacci
Cisco Systems
V. Fuller

D. Meyer
D. Lewis
Cisco Systems

January 2013

The Locator/ID Separation Protocol (LISP)

Abstract

This document describes a network-layer-based protocol that enables separation of IP addresses into two new numbering spaces: Endpoint Identifiers (EIDs) and Routing Locators (RLOCs). No changes are required to either host protocol stacks or to the "core" of the Internet infrastructure. The Locator/ID Separation Protocol (LISP) can be incrementally deployed, without a "flag day", and offers Traffic Engineering, multihoming, and mobility benefits to early adopters, even when there are relatively few LISP-capable sites.

Design and development of LISP was largely motivated by the problem statement produced by the October 2006 IAB Routing and Addressing Workshop.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6830.

Table of Contents

1. Introduction

This document describes the Locator/Identifier Separation Protocol (LISP), which provides a set of functions for routers to exchange information used to map from Endpoint Identifiers (EIDs) that are not globally routable to routable Routing Locators (RLOCs). It also defines a mechanism for these LISP routers to encapsulate IP packets addressed with EIDs for transmission across a network infrastructure that uses RLOCs for routing and forwarding.

Creation of LISP was initially motivated by discussions during the IAB-sponsored Routing and Addressing Workshop held in Amsterdam in October 2006 (see [RFC4984]). A key conclusion of the workshop was that the Internet routing and addressing system was not scaling well in the face of the explosive growth of new sites; one reason for this poor scaling is the increasing number of multihomed sites and other sites that cannot be addressed as part of topology-based or provider-based aggregated prefixes. Additional work that more completely describes the problem statement may be found in [RADIR].

A basic observation, made many years ago in early networking research such as that documented in [CHIAPPA] and [RFC4984], is that using a single address field for both identifying a device and for determining where it is topologically located in the network requires optimization along two conflicting axes: for routing to be efficient, the address must be assigned topologically; for collections of devices to be easily and effectively managed, without the need for renumbering in response to topological change (such as that caused by adding or removing attachment points to the network or by mobility events), the address must explicitly not be tied to the topology.

The approach that LISP takes to solving the routing scalability problem is to replace IP addresses with two new types of numbers: Routing Locators (RLOCs), which are topologically assigned to network attachment points (and are therefore amenable to aggregation) and used for routing and forwarding of packets through the network; and Endpoint Identifiers (EIDs), which are assigned independently from the network topology, are used for numbering devices, and are aggregated along administrative boundaries. LISP then defines functions for mapping between the two numbering spaces and for encapsulating traffic originated by devices using non-routable EIDs for transport across a network infrastructure that routes and forwards using RLOCs. Both RLOCs and EIDs are syntactically identical to IP addresses; it is the semantics of how they are used that differs.

This document describes the protocol that implements these functions. The database that stores the mappings between EIDs and RLOCs is explicitly a separate "module" to facilitate experimentation with a variety of approaches. One database design that is being developed for experimentation as part of the LISP working group work is [RFC6836]. Others that have been described include [CONS], [EMACS], and [RFC6837]. Finally, [RFC6833] documents a general-purpose service interface for accessing a mapping database; this interface is intended to make the mapping database modular so that different approaches can be tried without the need to modify installed LISP-capable devices in LISP sites.

This experimental specification has areas that require additional experience and measurement. It is NOT RECOMMENDED for deployment beyond experimental situations. Results of experimentation may lead to modifications and enhancements of protocol mechanisms defined in this document. See Section 15 for specific, known issues that are in need of further work during development, implementation, and experimentation.

An examination of the implications of LISP on Internet traffic, applications, routers, and security is for future study. This analysis will explain what role LISP can play in scalable routing and will also look at scalability and levels of state required for encapsulation, decapsulation, liveness, and so on.

2. Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Definition of Terms

Provider-Independent (PI) Addresses: PI addresses are an address block assigned from a pool where blocks are not associated with any particular location in the network (e.g., from a particular service provider) and are therefore not topologically aggregatable in the routing system.

Provider-Assigned (PA) Addresses: PA addresses are an address block assigned to a site by each service provider to which a site connects. Typically, each block is a sub-block of a service provider Classless Inter-Domain Routing (CIDR) [RFC4632] block and is aggregated into the larger block before being advertised into the global Internet. Traditionally, IP multihoming has been implemented by each multihomed site acquiring its own globally visible prefix. LISP uses only topologically assigned and aggregatable address blocks for RLOCs, eliminating this demonstrably non-scalable practice.

Routing Locator (RLOC): An RLOC is an IPv4 [RFC0791] or IPv6 [RFC2460] address of an Egress Tunnel Router (ETR). An RLOC is the output of an EID-to-RLOC mapping lookup. An EID maps to one or more RLOCs. Typically, RLOCs are 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. Multiple RLOCs can be assigned to the same ETR device or to multiple ETR devices at a site.

Endpoint ID (EID): An EID is a 32-bit (for IPv4) or 128-bit (for IPv6) value used in the source and destination address fields of the first (most inner) LISP header of a packet. The host obtains a destination EID the same way it obtains a destination address today, for example, through a Domain Name System (DNS) [RFC1034] lookup or Session Initiation Protocol (SIP) [RFC3261] exchange. The source EID is obtained via existing mechanisms used to set a host's "local" IP address. An EID used on the public Internet must have the same properties as any other IP address used in that manner; this means, among other things, that it must be globally unique. An EID is allocated to a host from an EID-Prefix block associated with the site where the host is located. An EID can be used by a host to refer to other hosts. EIDs MUST NOT be used as LISP RLOCs. Note that EID blocks MAY be assigned in a hierarchical manner, independent of the network topology, to facilitate scaling of the mapping database. In addition, an EID block assigned to a site may have site-local structure (subnetting) for routing within the site; this structure is not visible to the global routing system. In theory, the bit string that represents an EID for one device can represent an RLOC for a different device. As the architecture is realized, if a given bit string is both an RLOC and an EID, it must refer to the same entity in both cases. When used in discussions with other Locator/ID separation proposals, a LISP EID will be called an "LEID". Throughout this document, any references to "EID" refer to an LEID.

EID-Prefix: An EID-Prefix is a power-of-two block of EIDs that are allocated to a site by an address allocation authority. EID-Prefixes are associated with a set of RLOC addresses that make up a "database mapping". EID-Prefix allocations can be broken up into smaller blocks when an RLOC set is to be associated with the larger EID-Prefix block. A globally routed address block (whether PI or PA) is not inherently an EID-Prefix. A globally routed address block MAY be used by its assignee as an EID block. The converse is not supported. That is, a site that receives an explicitly allocated EID-Prefix may not use that EID-Prefix as a globally routed prefix. This would require coordination and cooperation with the entities managing the mapping infrastructure. Once this has been done, that block could be removed from the globally routed IP system, if other suitable transition and access mechanisms are in place. Discussion of such transition and access mechanisms can be found in [RFC6832] and [LISP-DEPLOY].

End-system: An end-system is an IPv4 or IPv6 device that originates packets with a single IPv4 or IPv6 header. The end-system supplies an EID value for the destination address field of the IP header when communicating globally (i.e., outside of its routing domain). An end-system can be a host computer, a switch or router device, or any network appliance.

Ingress Tunnel Router (ITR): An ITR is a router that resides in a LISP site. Packets sent by sources inside of the LISP site to destinations outside of the site are candidates for encapsulation by the ITR. The ITR treats the IP destination address as an EID and performs an EID-to-RLOC mapping lookup. The router then prepends an "outer" IP header with one of its globally routable RLOCs in the source address field and the result of the mapping lookup in the destination address field. Note that this destination RLOC MAY be an intermediate, proxy device that has better knowledge of the EID-to-RLOC mapping closer to the destination EID. In general, an ITR receives IP packets from site end-systems on one side and sends LISP-encapsulated IP packets toward the Internet on the other side.

Specifically, when a service provider prepends a LISP header for Traffic Engineering purposes, the router that does this is also regarded as an ITR. The outer RLOC the ISP ITR uses can be based on the outer destination address (the originating ITR’s supplied RLOC) or the inner destination address (the originating host’s supplied EID).

TE-ITR: A TE-ITR is an ITR that is deployed in a service provider network that prepends an additional LISP header for Traffic Engineering purposes.

Egress Tunnel Router (ETR): An ETR is a router that accepts an IP packet where the destination address in the "outer" IP header is one of its own RLOCs. The router strips the "outer" header and forwards the packet based on the next IP header found. In general, an ETR receives LISP-encapsulated IP packets from the Internet on one side and sends decapsulated IP packets to site end-systems on the other side. ETR functionality does not have to be limited to a router device. A server host can be the endpoint of a LISP tunnel as well.

TE-ETR: A TE-ETR is an ETR that is deployed in a service provider network that strips an outer LISP header for Traffic Engineering purposes.

xTR: An xTR is a reference to an ITR or ETR when direction of data flow is not part of the context description. "xTR" refers to the router that is the tunnel endpoint and is used synonymously with the term "Tunnel Router". For example, "An xTR can be located at the Customer Edge (CE) router" indicates both ITR and ETR functionality at the CE router.

LISP Router: A LISP router is a router that performs the functions of any or all of the following: ITR, ETR, Proxy-ITR (PITR), or Proxy-ETR (PETR).

EID-to-RLOC Cache: The EID-to-RLOC Cache is a short-lived, 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 distinct from the full "database" of EID-to-RLOC mappings; it is dynamic, local to the ITR(s), and relatively small, while the database is distributed, relatively static, and much more global in scope.

EID-to-RLOC Database: The EID-to-RLOC Database is a global distributed database that contains all known EID-Prefix-to-RLOC mappings. Each potential ETR typically contains a small piece of the database: the EID-to-RLOC mappings for the EID-Prefixes "behind" the router. These map to one of the router’s own globally visible IP addresses. The same database mapping entries MUST be configured on all ETRs for a given site. In a steady state, the EID-Prefixes for the site and the Locator-Set for each EID-Prefix MUST be the same on all ETRs. Procedures to enforce and/or verify this are outside the scope of this document. Note that there MAY be transient conditions when the EID-Prefix for the site and Locator-Set for each EID-Prefix may not be the same on all ETRs. This has no negative implications, since a partial set of Locators can be used.

Recursive Tunneling: Recursive Tunneling occurs when a packet has more than one LISP IP header. Additional layers of tunneling MAY be employed to implement Traffic Engineering or other re-routing as needed. When this is done, an additional "outer" LISP header is added, and the original RLOCs are preserved in the "inner" header. Any references to tunnels in this specification refer to dynamic encapsulating tunnels; they are never statically configured.

Re-encapsulating Tunnels: Re-encapsulating Tunneling occurs when an ETR removes a LISP header, then acts as an ITR to prepend another LISP header. Doing this allows a packet to be re-routed by the re-encapsulating router without adding the overhead of additional tunnel headers. Any references to tunnels in this specification

refer to dynamic encapsulating tunnels; they are never statically configured. When using multiple mapping database systems, care must be taken to not create re-encapsulation loops through misconfiguration.

LISP Header: LISP header is a term used in this document to refer to the outer IPv4 or IPv6 header, a UDP header, and a LISP-specific 8-octet header that follow the UDP header and that an ITR prepends or an ETR strips.

Address Family Identifier (AFI): AFI is a term used to describe an address encoding in a packet. An address family currently pertains to an IPv4 or IPv6 address. See [AFI] and [RFC3232] for details. An AFI value of 0 used in this specification indicates an unspecified encoded address where the length of the address is 0 octets following the 16-bit AFI value of 0.

Negative Mapping Entry: A negative mapping entry, also known as a negative cache entry, is an EID-to-RLOC entry where an EID-Prefix is advertised or stored with no RLOCs. That is, the Locator-Set for the EID-to-RLOC entry is empty or has an encoded Locator count of 0. This type of entry could be used to describe a prefix from a non-LISP site, which is explicitly not in the mapping database. There are a set of well-defined actions that are encoded in a Negative Map-Reply (Section 6.1.5).

Data-Probe: A Data-Probe is a LISP-encapsulated data packet where the inner-header destination address equals the outer-header destination address used to trigger a Map-Reply by a decapsulating ETR. In addition, the original packet is decapsulated and delivered to the destination host if the destination EID is in the EID-Prefix range configured on the ETR. Otherwise, the packet is discarded. A Data-Probe is used in some of the mapping database designs to "probe" or request a Map-Reply from an ETR; in other cases, Map-Requests are used. See each mapping database design for details. When using Data-Probes, by sending Map-Requests on the underlying routing system, EID-Prefixes must be advertised. However, this is discouraged if the core is to scale by having less EID-Prefixes stored in the core router’s routing tables.

Proxy-ITR (PITR): A PITR is defined and described in [RFC6832]. A PITR acts like an ITR but does so on behalf of non-LISP sites that send packets to destinations at LISP sites.

Proxy-ETR (PETR): A PETR is defined and described in [RFC6832]. A PETR acts like an ETR but does so on behalf of LISP sites that send packets to destinations at non-LISP sites.

Route-returnability: Route-returnability is an assumption that the underlying routing system will deliver packets to the destination. When combined with a nonce that is provided by a sender and returned by a receiver, this limits off-path data insertion. A route-returnability check is verified when a message is sent with a nonce, another message is returned with the same nonce, and the destination of the original message appears as the source of the returned message.

LISP site: LISP site is a set of routers in an edge network that are under a single technical administration. LISP routers that reside in the edge network are the demarcation points to separate the edge network from the core network.

Client-side: Client-side is a term used in this document to indicate a connection initiation attempt by an EID. The ITR(s) at the LISP site are the first to get involved in obtaining database Map-Cache entries by sending Map-Request messages.

Server-side: Server-side is a term used in this document to indicate that a connection initiation attempt is being accepted for a destination EID. The ETR(s) at the destination LISP site are the first to send Map-Replies to the source site initiating the connection. The ETR(s) at this destination site can obtain mappings by gleaning information from Map-Requests, Data-Probes, or encapsulated packets.

Locator-Status-Bits (LSBs): Locator-Status-Bits are present in the LISP header. They are used by ITRs to inform ETRs about the up/down status of all ETRs at the local site. These bits are used as a hint to convey up/down router status and not path reachability status. The LSBs can be verified by use of one of the Locator reachability algorithms described in Section 6.3.

Anycast Address: Anycast Address is a term used in this document to refer to the same IPv4 or IPv6 address configured and used on multiple systems at the same time. An EID or RLOC can be an anycast address in each of their own address spaces.

4. Basic Overview

One key concept of LISP is that end-systems (hosts) operate the same way they do today. The IP addresses that hosts use for tracking sockets and connections, and for sending and receiving packets, do not change. In LISP terminology, these IP addresses are called Endpoint Identifiers (EIDs).

Routers continue to forward packets based on IP destination addresses. When a packet is LISP encapsulated, these addresses are referred to as Routing Locators (RLOCs). Most routers along a path between two hosts will not change; they continue to perform routing/forwarding lookups on the destination addresses. For routers between the source host and the ITR as well as routers from the ETR to the destination host, the destination address is an EID. For the routers between the ITR and the ETR, the destination address is an RLOC.

Another key LISP concept is the "Tunnel Router". A Tunnel Router prepends LISP headers on host-originated packets and strips them prior to final delivery to their destination. The IP addresses in this "outer header" are RLOCs. During end-to-end packet exchange between two Internet hosts, an ITR prepends a new LISP header to each packet, and an ETR strips the new header. The ITR performs EID-to-RLOC lookups to determine the routing path to the ETR, which has the RLOC as one of its IP addresses.

Some basic rules governing LISP are:

- End-systems (hosts) only send to addresses that are EIDs. They don’t know that addresses are EIDs versus RLOCs but assume that packets get to their intended destinations. In a system where LISP is deployed, LISP routers intercept EID-addressed packets and assist in delivering them across the network core where EIDs cannot be routed. The procedure a host uses to send IP packets does not change.

- EIDs are always IP addresses assigned to hosts.

- LISP routers mostly deal with Routing Locator addresses. See details in Section 4.1 to clarify what is meant by "mostly".

- RLOCs are always IP addresses assigned to routers, preferably topologically oriented addresses from provider CIDR (Classless Inter-Domain Routing) blocks.

- When a router originates packets, it may use as a source address either an EID or RLOC. When acting as a host (e.g., when terminating a transport session such as Secure SHell (SSH), TELNET, or the Simple Network Management Protocol (SNMP)), it may use an EID that is explicitly assigned for that purpose. An EID that identifies the router as a host MUST NOT be used as an RLOC; an EID is only routable within the scope of a site. A typical BGP configuration might demonstrate this "hybrid" EID/RLOC usage where a router could use its "host-like" EID to terminate iBGP sessions to other routers in a site while at the same time using RLOCs to terminate eBGP sessions to routers outside the site.

- Packets with EIDs in them are not expected to be delivered end-to-end in the absence of an EID-to-RLOC mapping operation. They are expected to be used locally for intra-site communication or to be encapsulated for inter-site communication.

- EID-Prefixes are likely to be hierarchically assigned in a manner that is optimized for administrative convenience and to facilitate scaling of the EID-to-RLOC mapping database. The hierarchy is based on an address allocation hierarchy that is independent of the network topology.

- EIDs may also be structured (subnetted) in a manner suitable for local routing within an Autonomous System (AS).

An additional LISP header MAY be prepended to packets by a TE-ITR when re-routing of the path for a packet is desired. A potential use-case for this would be an ISP router that needs to perform Traffic Engineering for packets flowing through its network. In such a situation, termed "Recursive Tunneling", an ISP transit acts as an additional ITR, and the RLOC it uses for the new prepended header would be either a TE-ETR within the ISP (along an intra-ISP traffic engineered path) or a TE-ETR within another ISP (an inter-ISP traffic engineered path, where an agreement to build such a path exists).

In order to avoid excessive packet overhead as well as possible encapsulation loops, this document mandates that a maximum of two LISP headers can be prepended to a packet. For initial LISP deployments, it is assumed that two headers is sufficient, where the first prepended header is used at a site for Location/Identity separation and the second prepended header is used inside a service provider for Traffic Engineering purposes.

Tunnel Routers can be placed fairly flexibly in a multi-AS topology. For example, the ITR for a particular end-to-end packet exchange might be the first-hop or default router within a site for the source host. Similarly, the ETR might be the last-hop router directly connected to the destination host. Another example, perhaps for a VPN service outsourced to an ISP by a site, the ITR could be the site’s border router at the service provider attachment point. Mixing and matching of site-operated, ISP-operated, and other Tunnel Routers is allowed for maximum flexibility. See Section 8 for more details.

4.1. Packet Flow Sequence

This section provides an example of the unicast packet flow with the following conditions:

- Source host "host1.abc.example.com" is sending a packet to "host2.xyz.example.com", exactly what host1 would do if the site was not using LISP.
- Each site is multihomed, so each Tunnel Router has an address (RLOC) assigned from the service provider address block for each provider to which that particular Tunnel Router is attached.
- The ITR(s) and ETR(s) are directly connected to the source and destination, respectively, but the source and destination can be located anywhere in the LISP site.
- Map-Requests can be sent on the underlying routing system topology, to a mapping database system, or directly over an Alternative Logical Topology [RFC6836]. A Map-Request is sent for an external destination when the destination is not found in the forwarding table or matches a default route.
- Map-Replies are sent on the underlying routing system topology.

Client host1.abc.example.com wants to communicate with server host2.xyz.example.com:

1. host1.abc.example.com wants to open a TCP connection to host2.xyz.example.com. It does a DNS lookup on host2.xyz.example.com. An A/AAAA record is returned. This address is the destination EID. The locally assigned address of host1.abc.example.com is used as the source EID. An IPv4 or IPv6 packet is built and forwarded through the LISP site as a normal IP packet until it reaches a LISP ITR.
2. The LISP ITR must be able to map the destination EID to an RLOC of one of the ETRs at the destination site. The specific method used to do this is not described in this example. See [RFC6836] or [CONS] for possible solutions.
3. The ITR will send a LISP Map-Request. Map-Requests SHOULD be rate-limited.

4. When an alternate mapping system is not in use, the Map-Request packet is routed through the underlying routing system. Otherwise, the Map-Request packet is routed on an alternate logical topology, for example, the [RFC6836] database mapping system. In either case, when the Map-Request arrives at one of the ETRs at the destination site, it will process the packet as a control message.

5. The ETR looks at the destination EID of the Map-Request and matches it against the prefixes in the ETR's configured EID-to-RLOC mapping database. This is the list of EID-Prefixes the ETR is supporting for the site it resides in. If there is no match, the Map-Request is dropped. Otherwise, a LISP Map-Reply is returned to the ITR.

6. The ITR receives the Map-Reply message, parses the message (to check for format validity), and stores the mapping information from the packet. This information is stored in the ITR's EID-to-RLOC mapping cache. Note that the map-cache is an on-demand cache. An ITR will manage its map-cache in such a way that optimizes for its resource constraints.

7. Subsequent packets from host1.abc.example.com to host2.xyz.example.com will have a LISP header prepended by the ITR using the appropriate RLOC as the LISP header destination address learned from the ETR. Note that the packet MAY be sent to a different ETR than the one that returned the Map-Reply due to the source site's hashing policy or the destination site's Locator-Set policy.

8. The ETR receives these packets directly (since the destination address is one of its assigned IP addresses), checks the validity of the addresses, strips the LISP header, and forwards packets to the attached destination host.

In order to defer the need for a mapping lookup in the reverse direction, an ETR MAY create a cache entry that maps the source EID (inner-header source IP address) to the source RLOC (outer-header source IP address) in a received LISP packet. Such a cache entry is termed a "gleaned" mapping and only contains a single RLOC for the EID in question. More complete information about additional RLOCs SHOULD be verified by sending a LISP Map-Request for that EID. Both the ITR and the ETR may also influence the decision the other makes in selecting an RLOC. See Section 6 for more details.

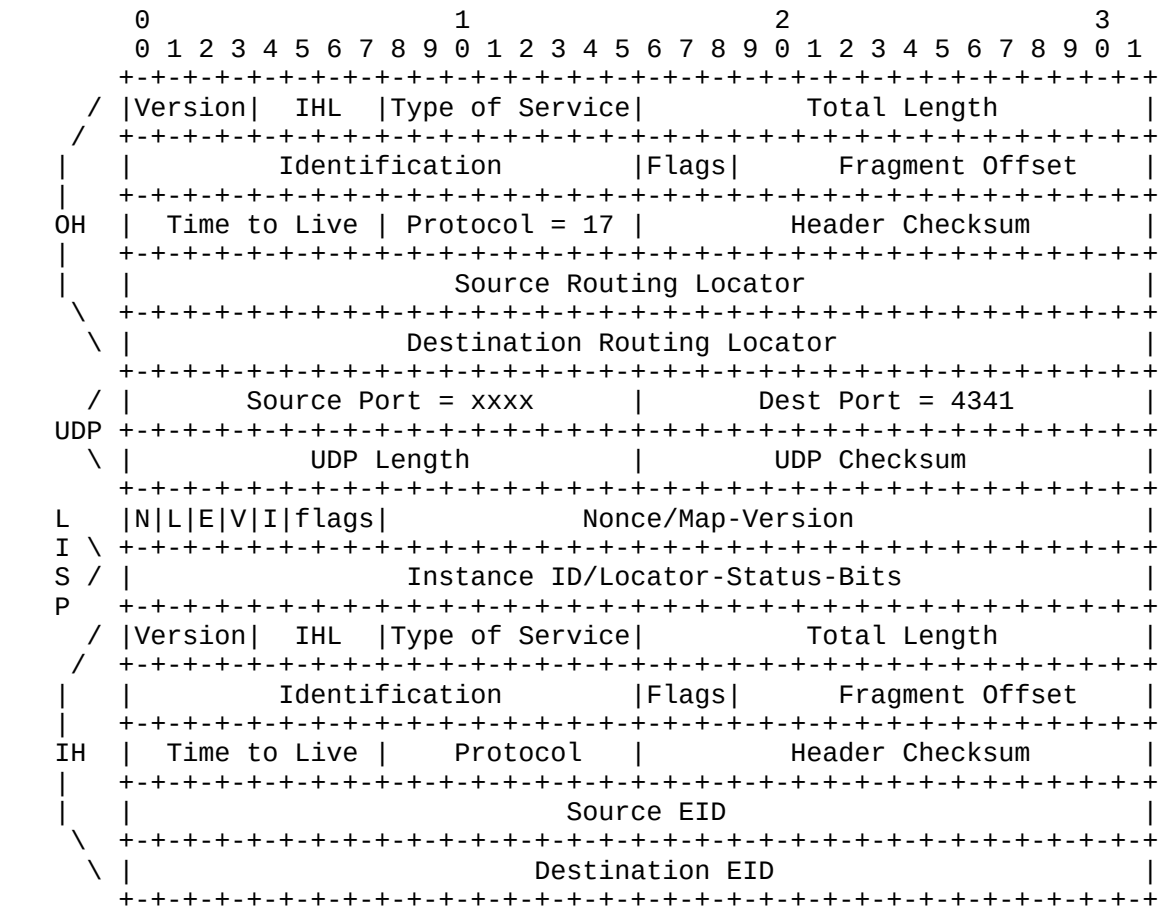
5. LISP Encapsulation Details

Since additional tunnel headers are prepended, the packet becomes larger and 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.

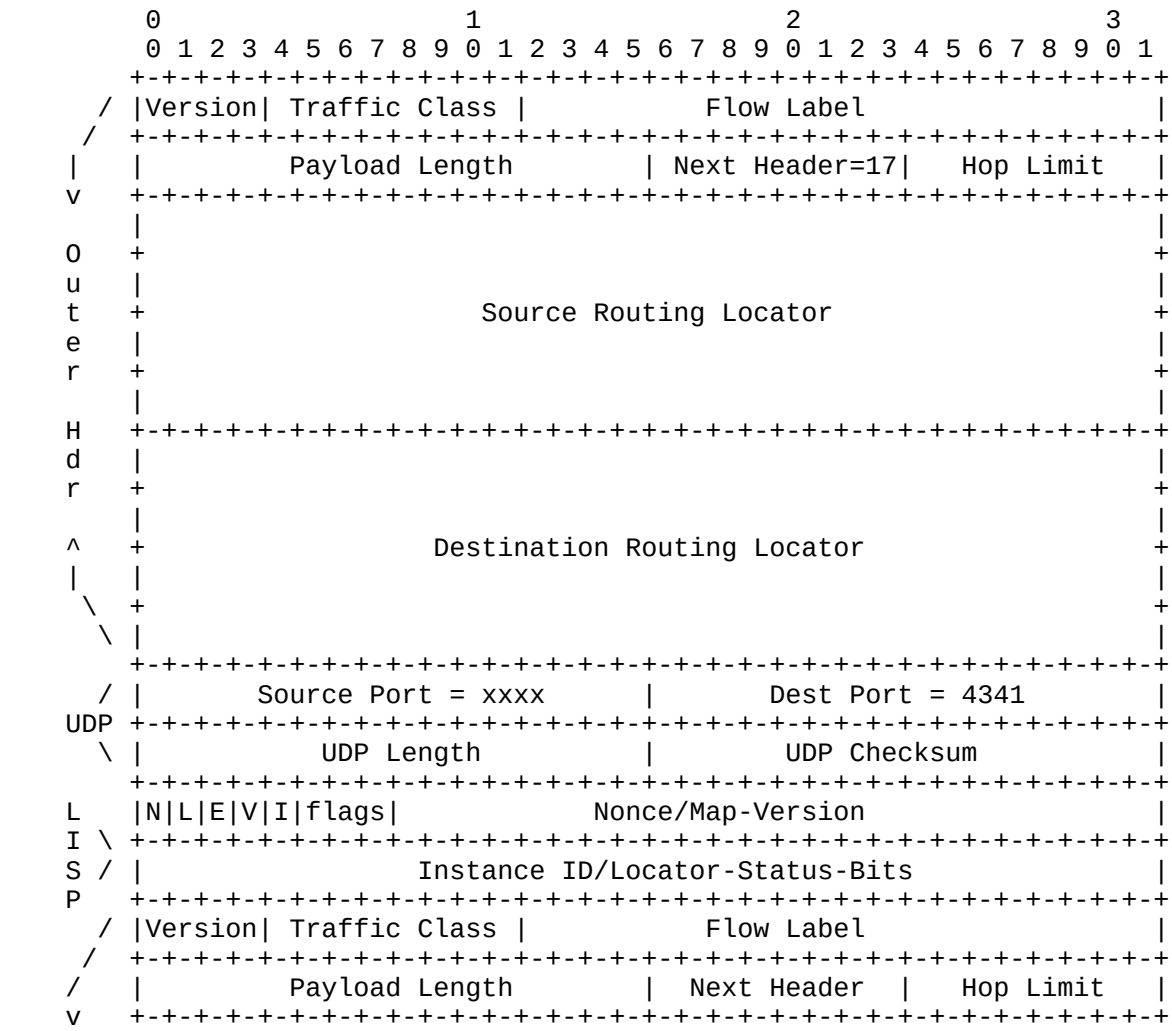
This specification RECOMMENDS that implementations provide support for one of the proposed fragmentation and reassembly schemes. Two existing schemes are detailed in Section 5.4.

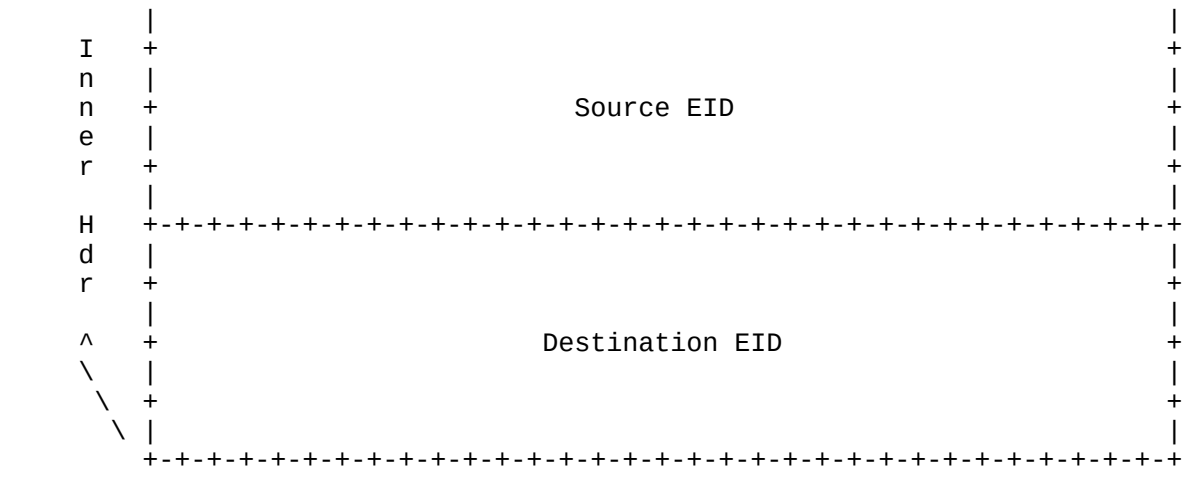
Since IPv4 or IPv6 addresses can be either EIDs or RLOCs, the LISP architecture supports IPv4 EIDs with IPv6 RLOCs (where the inner header is in IPv4 packet format and the outer header is in IPv6 packet format) or IPv6 EIDs with IPv4 RLOCs (where the inner header is in IPv6 packet format and the outer header is in IPv4 packet format). The next sub-sections illustrate packet formats for the homogeneous case (IPv4-in-IPv4 and IPv6-in-IPv6), but all 4 combinations MUST be supported.

5.1. LISP IPv4-in-IPv4 Header Format



5.2. LISP IPv6-in-IPv6 Header Format





5.3. Tunnel Header Field Descriptions

Inner Header (IH): The inner header is the header on the datagram received from the originating host. The source and destination IP addresses are EIDs [RFC0791] [RFC2460].

Outer Header: (OH) The outer header is a new header prepended by an ITR. The address fields contain RLOCs obtained from the ingress router's EID-to-RLOC Cache. The IP protocol number is "UDP (17)" from [RFC0768]. The setting of the Don't Fragment (DF) bit 'Flags' field is according to rules listed in Sections 5.4.1 and 5.4.2.

UDP Header: The UDP header contains an ITR selected source port when encapsulating a packet. See Section 6.5 for details on the hash algorithm used to select a source port based on the 5-tuple of the inner header. The destination port MUST be set to the well-known IANA-assigned port value 4341.

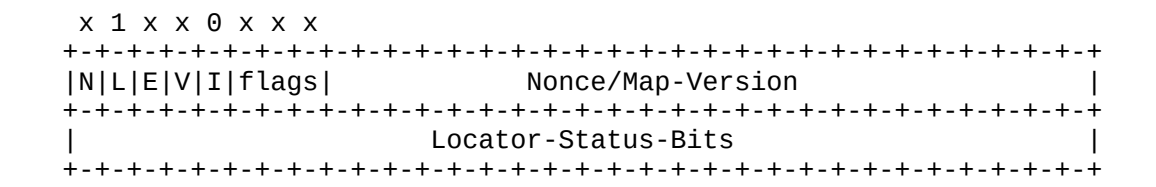
UDP Checksum: The 'UDP Checksum' field SHOULD be transmitted as zero by an ITR for either IPv4 [RFC0768] or IPv6 encapsulation [UDP-TUNNELS] [UDP-ZERO]. When a packet with a zero UDP checksum is received by an ETR, the ETR MUST accept the packet for decapsulation. When an ITR transmits a non-zero value for the UDP checksum, it MUST send a correctly computed value in this field. When an ETR receives a packet with a non-zero UDP checksum, it MAY choose to verify the checksum value. If it chooses to perform such verification, and the verification fails, the packet MUST be silently dropped. If the ETR chooses not to perform the verification, or performs the verification successfully, the packet MUST be accepted for decapsulation. The handling of UDP

checksums for all tunneling protocols, including LISP, is under active discussion within the IETF. When that discussion concludes, any necessary changes will be made to align LISP with the outcome of the broader discussion.

UDP Length: The 'UDP Length' field is set for an IPv4-encapsulated packet to be the sum of the inner-header IPv4 Total Length plus the UDP and LISP header lengths. For an IPv6-encapsulated packet, the 'UDP Length' field is the sum of the inner-header IPv6 Payload Length, the size of the IPv6 header (40 octets), and the size of the UDP and LISP headers.

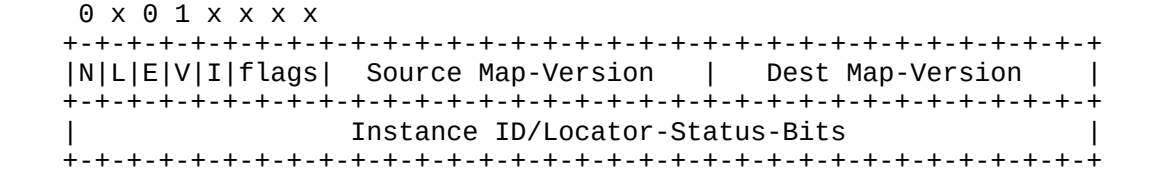
N: The N-bit is the nonce-present bit. When this bit is set to 1, the low-order 24 bits of the first 32 bits of the LISP header contain a Nonce. See Section 6.3.1 for details. Both N- and V-bits MUST NOT be set in the same packet. If they are, a decapsulating ETR MUST treat the 'Nonce/Map-Version' field as having a Nonce value present.

L: The L-bit is the 'Locator-Status-Bits' field enabled bit. When this bit is set to 1, the Locator-Status-Bits in the second 32 bits of the LISP header are in use.

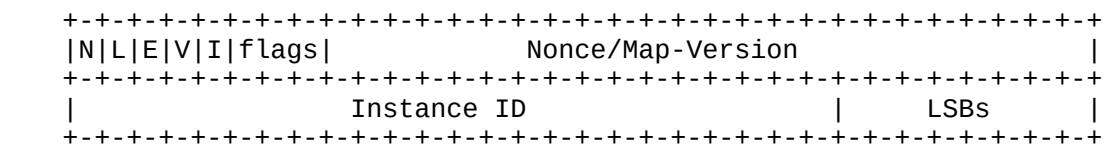


E: The E-bit is the echo-nonce-request bit. This bit MUST be ignored and has no meaning when the N-bit is set to 0. When the N-bit is set to 1 and this bit is set to 1, an ITR is requesting that the nonce value in the 'Nonce' field be echoed back in LISP-encapsulated packets when the ITR is also an ETR. See Section 6.3.1 for details.

V: The V-bit is the Map-Version present bit. When this bit is set to 1, the N-bit MUST be 0. Refer to Section 6.6.3 for more details. This bit indicates that the LISP header is encoded in this case as:



I: The I-bit is the Instance ID bit. See Section 5.5 for more details. When this bit is set to 1, the 'Locator-Status-Bits' field is reduced to 8 bits and the high-order 24 bits are used as an Instance ID. If the L-bit is set to 0, then the low-order 8 bits are transmitted as zero and ignored on receipt. The format of the LISP header would look like this:



flags: The 'flags' field is a 3-bit field reserved for future flag use. It MUST be set to 0 on transmit and MUST be ignored on receipt.

LISP Nonce: The LISP 'Nonce' field is a 24-bit value that is randomly generated by an ITR when the N-bit is set to 1. Nonce generation algorithms are an implementation matter but are required to generate different nonces when sending to different destinations. However, the same nonce can be used for a period of time to the same destination. The nonce is also used when the E-bit is set to request the nonce value to be echoed by the other side when packets are returned. When the E-bit is clear but the N-bit is set, a remote ITR is either echoing a previously requested echo-nonce or providing a random nonce. See Section 6.3.1 for more details.

LISP Locator-Status-Bits (LSBs): When the L-bit is also set, the 'Locator-Status-Bits' field in the LISP header is set by an ITR to indicate to an ETR the up/down status of the Locators in the source site. Each RLOC in a Map-Reply is assigned an ordinal value from 0 to n-1 (when there are n RLOCs in a mapping entry). The Locator-Status-Bits are numbered from 0 to n-1 from the least significant bit of the field. The field is 32 bits when the I-bit is set to 0 and is 8 bits when the I-bit is set to 1. When a Locator-Status-Bit is set to 1, the ITR is indicating to the ETR that the RLOC associated with the bit ordinal has up status. See Section 6.3 for details on how an ITR can determine the status of the ETRs at the same site. When a site has multiple EID-Prefixes that result in multiple mappings (where each could have a different Locator-Set), the Locator-Status-Bits setting in an encapsulated packet MUST reflect the mapping for the EID-Prefix that the inner-header source EID address matches. If the LSB for an anycast Locator is set to 1, then there is at least one RLOC with that address, and the ETR is considered 'up'.

When doing ITR/PITR encapsulation:

- The outer-header 'Time to Live' field (or 'Hop Limit' field, in the case of IPv6) SHOULD be copied from the inner-header 'Time to Live' field.

- The outer-header 'Type of Service' field (or the 'Traffic Class' field, in the case of IPv6) SHOULD be copied from the inner-header 'Type of Service' field (with one exception; see below).

When doing ETR/PETR decapsulation:

- The inner-header 'Time to Live' field (or 'Hop Limit' field, in the case of IPv6) SHOULD be copied from the outer-header 'Time to Live' field, when the Time to Live value of the outer header is less than the Time to Live value of the inner header. Failing to perform this check can cause the Time to Live of the inner header to increment across encapsulation/decapsulation cycles. This check is also performed when doing initial encapsulation, when a packet comes to an ITR or PITR destined for a LISP site.

- The inner-header 'Type of Service' field (or the 'Traffic Class' field, in the case of IPv6) SHOULD be copied from the outer-header 'Type of Service' field (with one exception; see below).

Note that if an ETR/PETR is also an ITR/PITR and chooses to re-encapsulate after decapsulating, the net effect of this is that the new outer header will carry the same Time to Live as the old outer header minus 1.

Copying the Time to Live (TTL) serves two purposes: first, it preserves the distance the host intended the packet to travel; second, and more importantly, it provides for suppression of looping packets in the event there is a loop of concatenated tunnels due to misconfiguration. See Section 9.3 for TTL exception handling for traceroute packets.

The Explicit Congestion Notification ('ECN') field occupies bits 6 and 7 of both the IPv4 'Type of Service' field and the IPv6 'Traffic Class' field [RFC3168]. The 'ECN' field requires special treatment in order to avoid discarding indications of congestion [RFC3168]. ITR encapsulation MUST copy the 2-bit 'ECN' field from the inner header to the outer header. Re-encapsulation MUST copy the 2-bit 'ECN' field from the stripped outer header to the new outer header. If the 'ECN' field contains a congestion indication codepoint (the value is '11', the Congestion Experienced (CE) codepoint), then ETR decapsulation MUST copy the 2-bit 'ECN' field from the stripped outer header to the surviving inner header that is used to forward the

packet beyond the ETR. These requirements preserve CE indications when a packet that uses ECN traverses a LISP tunnel and becomes marked with a CE indication due to congestion between the tunnel endpoints.

5.4. Dealing with Large Encapsulated Packets

This section proposes two mechanisms to deal with packets that exceed the path MTU between the ITR and ETR.

It is left to the implementor to decide if the stateless or stateful mechanism should be implemented. Both or neither can be used, since it is a local decision in the ITR regarding how to deal with MTU issues, and sites can interoperate with differing mechanisms.

Both stateless and stateful mechanisms also apply to Re-encapsulating and Recursive Tunneling, so any actions below referring to an ITR also apply to a TE-ITR.

5.4.1. A Stateless Solution to MTU Handling

An ITR stateless solution to handle MTU issues is described as follows:

1. Define H to be the size, in octets, of the outer header an ITR prepends to a packet. This includes the UDP and LISP header lengths.

2. Define L to be the size, in octets, of the maximum-sized packet an ITR can send to an ETR without the need for the ITR or any intermediate routers to fragment the packet.

3. Define an architectural constant S for the maximum size of a packet, in octets, an ITR must receive so the effective MTU can be met. That is, S = L - H.

When an ITR receives a packet from a site-facing interface and adds H octets worth of encapsulation to yield a packet size greater than L octets, it resolves the MTU issue by first splitting the original packet into 2 equal-sized fragments. A LISP header is then prepended to each fragment. The size of the encapsulated fragments is then (S/2 + H), which is less than the ITR’s estimate of the path MTU between the ITR and its correspondent ETR.

When an ETR receives encapsulated fragments, it treats them as two individually encapsulated packets. It strips the LISP headers and then forwards each fragment to the destination host of the destination site. The two fragments are reassembled at the

destination host into the single IP datagram that was originated by the source host. Note that reassembly can happen at the ETR if the encapsulated packet was fragmented at or after the ITR.

This behavior is performed by the ITR when the source host originates a packet with the 'DF' field of the IP header set to 0. When the 'DF' field of the IP header is set to 1, or the packet is an IPv6 packet originated by the source host, the ITR will drop the packet when the size is greater than L and send an ICMP Too Big message to the source with a value of S, where S is (L - H).

When the outer-header encapsulation uses an IPv4 header, an implementation SHOULD set the DF bit to 1 so ETR fragment reassembly can be avoided. An implementation MAY set the DF bit in such headers to 0 if it has good reason to believe there are unresolvable path MTU issues between the sending ITR and the receiving ETR.

This specification RECOMMENDS that L be defined as 1500.

5.4.2. A Stateful Solution to MTU Handling

An ITR stateful solution to handle MTU issues is described as follows and was first introduced in [OPENLISP]:

1. The ITR will keep state of the effective MTU for each Locator per Map-Cache entry. The effective MTU is what the core network can deliver along the path between the ITR and ETR.

2. When an IPv6-encapsulated packet, or an IPv4-encapsulated packet with the DF bit set to 1, exceeds what the core network can deliver, one of the intermediate routers on the path will send an ICMP Too Big message to the ITR. The ITR will parse the ICMP message to determine which Locator is affected by the effective MTU change and then record the new effective MTU value in the Map-Cache entry.

3. When a packet is received by the ITR from a source inside of the site and the size of the packet is greater than the effective MTU stored with the Map-Cache entry associated with the destination EID the packet is for, the ITR will send an ICMP Too Big message back to the source. The packet size advertised by the ITR in the ICMP Too Big message is the effective MTU minus the LISP encapsulation length.

Even though this mechanism is stateful, it has advantages over the stateless IP fragmentation mechanism, by not involving the destination host with reassembly of ITR fragmented packets.

5.5. Using Virtualization and Segmentation with LISP

When multiple organizations inside of a LISP site are using private addresses [RFC1918] as EID-Prefixes, their address spaces MUST remain segregated due to possible address duplication. An Instance ID in the address encoding can aid in making the entire AFI-based address unique. See IANA Considerations (Section 14.2) for details on possible address encodings.

An Instance ID can be carried in a LISP-encapsulated packet. An ITR that prepends a LISP header will copy a 24-bit value used by the LISP router to uniquely identify the address space. The value is copied to the 'Instance ID' field of the LISP header, and the I-bit is set to 1.

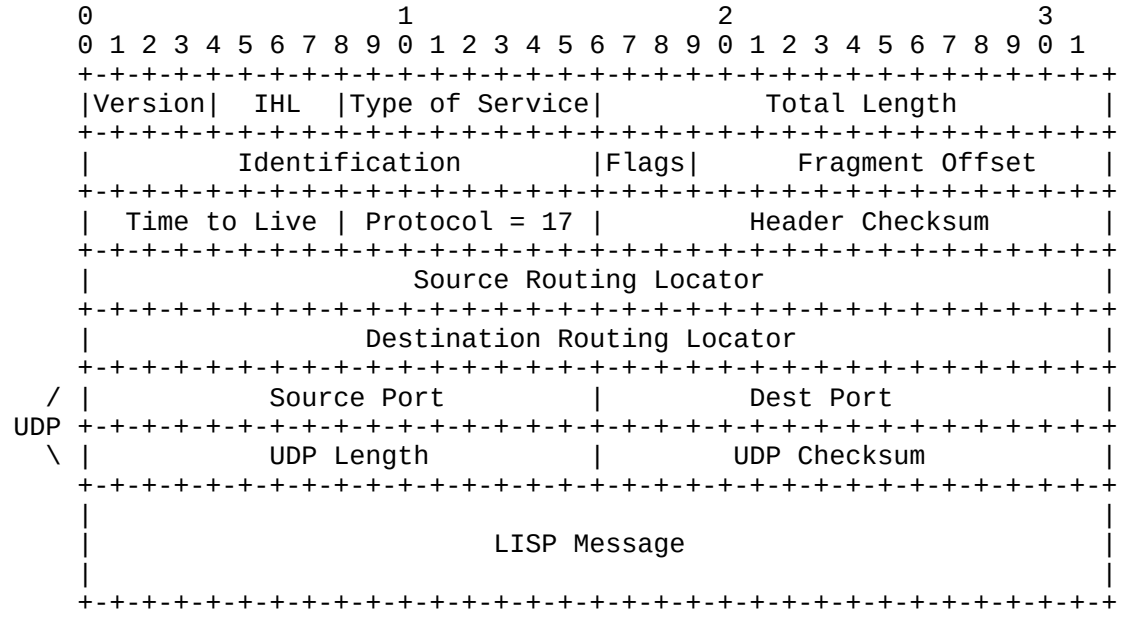
When an ETR decapsulates a packet, the Instance ID from the LISP header is used as a table identifier to locate the forwarding table to use for the inner destination EID lookup.

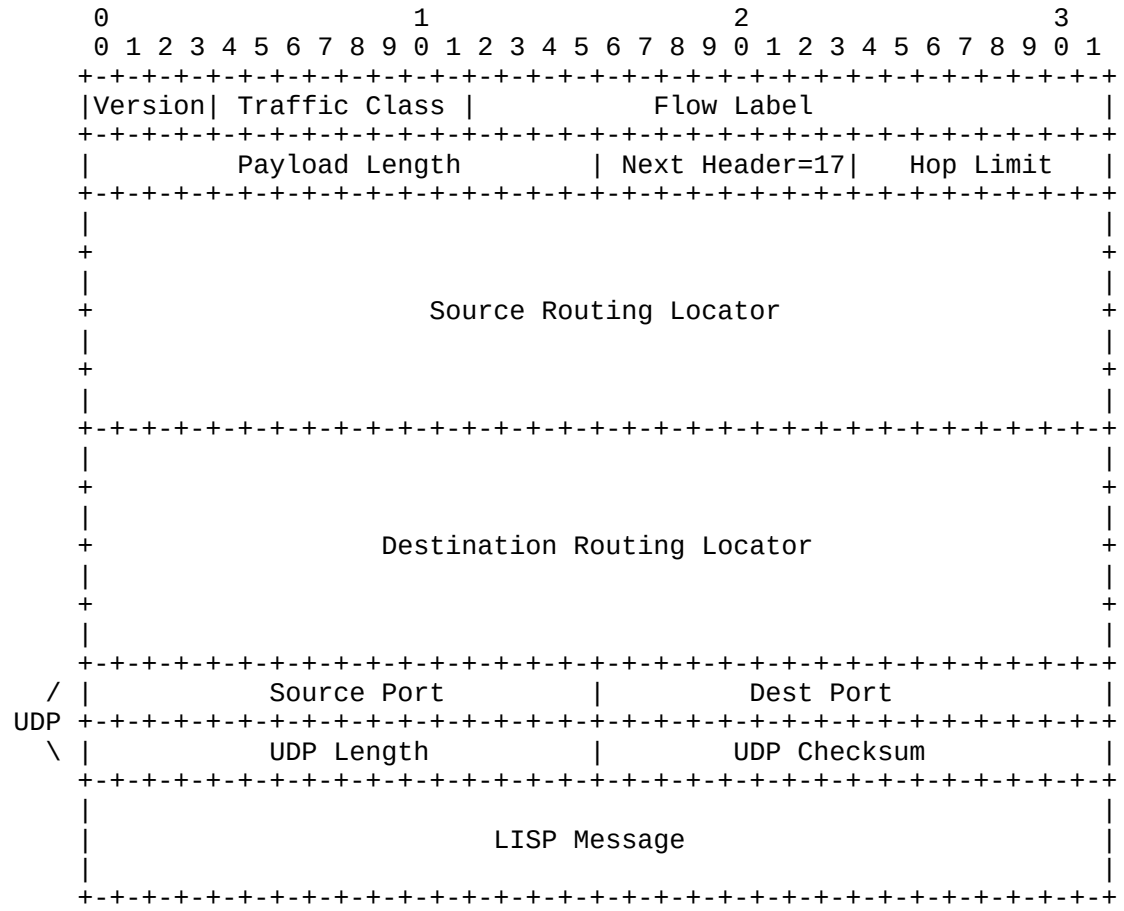
For example, an 802.1Q VLAN tag or VPN identifier could be used as a 24-bit Instance ID.

6. EID-to-RLOC Mapping

6.1. LISP IPv4 and IPv6 Control-Plane Packet Formats

The following UDP packet formats are used by the LISP control plane.





The LISP UDP-based messages are the Map-Request and Map-Reply messages. When a UDP Map-Request is sent, the UDP source port is chosen by the sender and the destination UDP port number is set to 4342. When a UDP Map-Reply is sent, the source UDP port number is set to 4342 and the destination UDP port number is copied from the source port of either the Map-Request or the invoking data packet. Implementations MUST be prepared to accept packets when either the source port or destination UDP port is set to 4342 due to NATs changing port number values.

The 'UDP Length' field will reflect the length of the UDP header and the LISP Message payload.

The UDP checksum is computed and set to non-zero for Map-Request, Map-Reply, Map-Register, and Encapsulated Control Message (ECM) control messages. It MUST be checked on receipt, and if the checksum fails, the packet MUST be dropped.

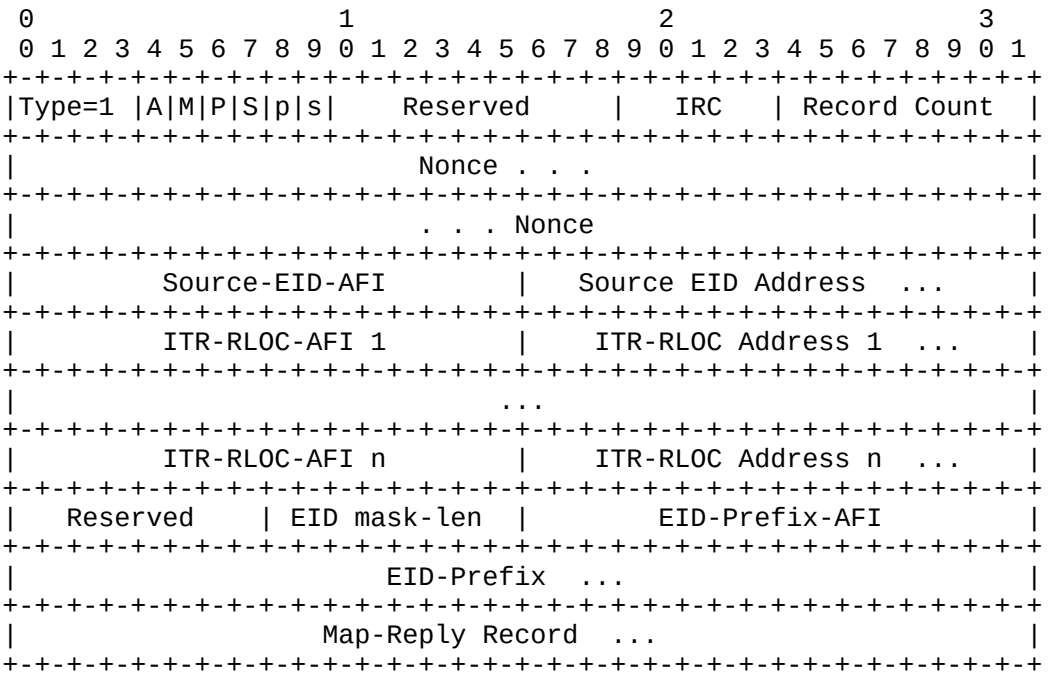
The format of control messages includes the UDP header so the checksum and length fields can be used to protect and delimit message boundaries.

6.1.1. LISP Packet Type Allocations

This section will be the authoritative source for allocating LISP Type values and for defining LISP control message formats. Current allocations are:

Reserved:	0	b’0000’
LISP Map-Request:	1	b’0001’
LISP Map-Reply:	2	b’0010’
LISP Map-Register:	3	b’0011’
LISP Map-Notify:	4	b’0100’
LISP Encapsulated Control Message:	8	b’1000’

6.1.2. Map-Request Message Format



Packet field descriptions:

Type: 1 (Map-Request)

A: This is an authoritative bit, which is set to 0 for UDP-based Map-Requests sent by an ITR. It is set to 1 when an ITR wants the destination site to return the Map-Reply rather than the mapping database system.

M: This is the map-data-present bit. When set, it indicates that a Map-Reply Record segment is included in the Map-Request.

P: This is the probe-bit, which indicates that a Map-Request SHOULD be treated as a Locator reachability probe. The receiver SHOULD respond with a Map-Reply with the probe-bit set, indicating that the Map-Reply is a Locator reachability probe reply, with the nonce copied from the Map-Request. See Section 6.3.2 for more details.

S: This is the Solicit-Map-Request (SMR) bit. See Section 6.6.2 for details.

p: This is the PITR bit. This bit is set to 1 when a PITR sends a Map-Request.

s: This is the SMR-invoked bit. This bit is set to 1 when an xTR is sending a Map-Request in response to a received SMR-based Map-Request.

Reserved: This field MUST be set to 0 on transmit and MUST be ignored on receipt.

IRC: This 5-bit field is the ITR-RLOC Count, which encodes the additional number of ('ITR-RLOC-AFI', 'ITR-RLOC Address') fields present in this message. At least one (ITR-RLOC-AFI, ITR-RLOC-Address) pair MUST be encoded. Multiple 'ITR-RLOC Address' fields are used, so a Map-Replier can select which destination address to use for a Map-Reply. The IRC value ranges from 0 to 31. For a value of 0, there is 1 ITR-RLOC address encoded; for a value of 1, there are 2 ITR-RLOC addresses encoded, and so on up to 31, which encodes a total of 32 ITR-RLOC addresses.

Record Count: This is the number of records in this Map-Request message. A record is comprised of the portion of the packet that is labeled 'Rec' above and occurs the number of times equal to Record Count. For this version of the protocol, a receiver MUST accept and process Map-Requests that contain one or more records,

but a sender MUST only send Map-Requests containing one record. Support for requesting multiple EIDs in a single Map-Request message will be specified in a future version of the protocol.

Nonce: This is an 8-octet random value created by the sender of the Map-Request. This nonce will be returned in the Map-Reply. The security of the LISP mapping protocol critically depends on the strength of the nonce in the Map-Request message. The nonce SHOULD be generated by a properly seeded pseudo-random (or strong random) source. See [RFC4086] for advice on generating security-sensitive random data.

Source-EID-AFI: This is the address family of the 'Source EID Address' field.

Source EID Address: This is the EID of the source host that originated the packet that caused the Map-Request. When Map-Requests are used for refreshing a Map-Cache entry or for RLOC-Probing, an AFI value 0 is used and this field is of zero length.

ITR-RLOC-AFI: This is the address family of the 'ITR-RLOC Address' field that follows this field.

ITR-RLOC Address: This is used to give the ETR the option of selecting the destination address from any address family for the Map-Reply message. This address MUST be a routable RLOC address of the sender of the Map-Request message.

EID mask-len: This is the mask length for the EID-Prefix.

EID-Prefix-AFI: This is the address family of the EID-Prefix according to [AFI].

EID-Prefix: This prefix is 4 octets for an IPv4 address family and 16 octets for 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, and the 'EID mask-len' is set to 32 or 128 for IPv4 or IPv6, respectively. When an xTR wants to query a site about the status of a mapping it already has 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 M-bit is set, this field is the size of a single "Record" in the Map-Reply format. This Map-Reply record contains the EID-to-RLOC mapping entry associated with the Source EID. This allows the ETR that will receive this Map-Request to cache the data if it chooses to do so.

6.1.3. EID-to-RLOC UDP Map-Request Message

A Map-Request is sent from an ITR when it needs a mapping for an EID, wants to test an RLOC for reachability, or wants to refresh a mapping before TTL expiration. For the initial case, the destination IP address used for the Map-Request is the data packet's destination address (i.e., the destination EID) that had a mapping cache lookup failure. For the latter two cases, the destination IP address used for the Map-Request is one of the RLOC addresses from the Locator-Set of the Map-Cache entry. The source address is either an IPv4 or IPv6 RLOC address, depending on whether the Map-Request is using an IPv4 or IPv6 header, respectively. In all cases, the UDP source port number for the Map-Request message is a 16-bit value selected by the ITR/PITR, and the UDP destination port number is set to the well-known destination port number 4342. A successful Map-Reply, which is one that has a nonce that matches an outstanding Map-Request nonce, will update the cached set of RLOCs associated with the EID-Prefix range.

One or more Map-Request ('ITR-RLOC-AFI', 'ITR-RLOC-Address') fields MUST be filled in by the ITR. The number of fields (minus 1) encoded MUST be placed in the 'IRC' field. The ITR MAY include all locally configured Locators in this list or just provide one locator address from each address family it supports. If the ITR erroneously provides no ITR-RLOC addresses, the Map-Replier MUST drop the Map-Request.

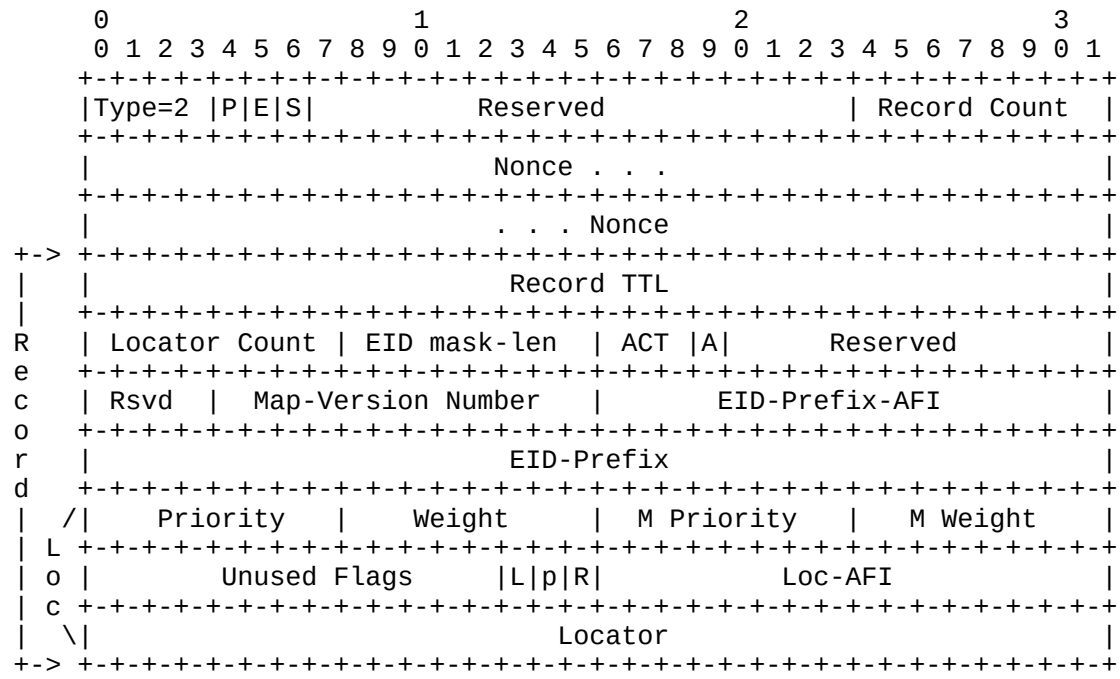
Map-Requests can also be LISP encapsulated using UDP destination port 4342 with a LISP Type value set to "Encapsulated Control Message", when sent from an ITR to a Map-Resolver. Likewise, Map-Requests are LISP encapsulated the same way from a Map-Server to an ETR. Details on Encapsulated Map-Requests and Map-Resolvers can be found in [RFC6833].

Map-Requests MUST be rate-limited. It is RECOMMENDED that a Map-Request for the same EID-Prefix be sent no more than once per second.

An ITR that is configured with mapping database information (i.e., it is also an ETR) MAY optionally include those mappings in a Map-Request. When an ETR configured to accept and verify such "piggybacked" mapping data receives such a Map-Request and it does

not have this mapping in the map-cache, it MAY originate a "verifying Map-Request", addressed to the map-requesting ITR and the ETR MAY add a Map-Cache entry. If the ETR has a Map-Cache entry that matches the "piggybacked" EID and the RLOC is in the Locator-Set for the entry, then it may send the "verifying Map-Request" directly to the originating Map-Request source. If the RLOC is not in the Locator-Set, then the ETR MUST send the "verifying Map-Request" to the "piggybacked" EID. Doing this forces the "verifying Map-Request" to go through the mapping database system to reach the authoritative source of information about that EID, guarding against RLOC-spoofing in the "piggybacked" mapping data.

6.1.4. Map-Reply Message Format



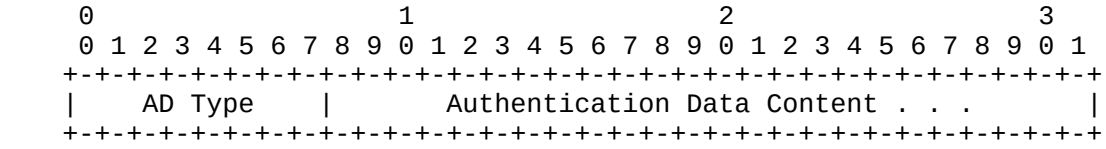
Packet field descriptions:

Type: 2 (Map-Reply)

P: This is the probe-bit, which indicates that the Map-Reply is in response to a Locator reachability probe Map-Request. The 'Nonce' field MUST contain a copy of the nonce value from the original Map-Request. See Section 6.3.2 for more details.

E: This bit indicates that the ETR that sends this Map-Reply message is advertising that the site is enabled for the Echo-Nonce Locator reachability algorithm. See Section 6.3.1 for more details.

S: This is the Security bit. When set to 1, the following authentication information will be appended to the end of the Map-Reply. The detailed format of the Authentication Data Content is for further study.



Reserved: This field MUST be set to 0 on transmit and MUST be ignored on receipt.

Record Count: This is the number of records in this reply message. A record is comprised of that portion of the packet labeled 'Record' above and occurs the number of times equal to Record Count.

Nonce: This is a 24-bit value set in a Data-Probe packet, or a 64-bit value from the Map-Request is echoed in this 'Nonce' field of the Map-Reply. When a 24-bit value is supplied, it resides in the low-order 64 bits of the 'Nonce' field.

Record TTL: This is the time in minutes the recipient of the Map-Reply will store the mapping. If the TTL is 0, the entry SHOULD be removed from the cache immediately. If the value is 0xffffffff, the recipient can decide locally how long to store the mapping.

Locator Count: This is the number of Locator entries. A Locator entry comprises what is labeled above as 'Loc'. The Locator count can be 0, indicating that there are no Locators for the EID-Prefix.

EID mask-len: This is the mask length for the EID-Prefix.

ACT: This 3-bit field describes Negative Map-Reply actions. In any other message type, these bits are set to 0 and ignored on receipt. These bits are used only when the 'Locator Count' field is set to 0. The action bits are encoded only in Map-Reply messages. The actions defined are used by an ITR or PITR when a destination EID matches a negative Map-Cache entry. Unassigned values should cause a Map-Cache entry to be created, and when packets match this negative cache entry, they will be dropped. The current assigned values are:

(0) No-Action: The map-cache is kept alive, and no packet encapsulation occurs.

(1) Natively-Forward: The packet is not encapsulated or dropped but natively forwarded.

(2) Send-Map-Request: The packet invokes sending a Map-Request.

(3) Drop: A packet that matches this map-cache entry is dropped. An ICMP Destination Unreachable message SHOULD be sent.

A: The Authoritative bit, when sent, is always set to 1 by an ETR. When a Map-Server is proxy Map-Replying [RFC6833] for a LISP site, the Authoritative bit is set to 0. This indicates to requesting ITRs that the Map-Reply was not originated by a LISP node managed at the site that owns the EID-Prefix.

Map-Version Number: When this 12-bit value is non-zero, the Map-Reply sender is informing the ITR what the version number is for the EID record contained in the Map-Reply. The ETR can allocate this number internally but MUST coordinate this value with other ETRs for the site. When this value is 0, there is no versioning information conveyed. The Map-Version Number can be included in Map-Request and Map-Register messages. See Section 6.6.3 for more details.

EID-Prefix-AFI: Address family of the EID-Prefix according to [AFI].

EID-Prefix: This prefix is 4 octets for an IPv4 address family and 16 octets for an IPv6 address family.

Priority: Each RLOC is assigned a unicast Priority. Lower values are more preferable. When multiple RLOCs have the same Priority, they MAY be used in a load-split fashion. A value of 255 means the RLOC MUST NOT be used for unicast forwarding.

Weight: When priorities are the same for multiple RLOCs, the Weight indicates how to balance unicast traffic between them. Weight is encoded as a relative weight of total unicast packets that match the mapping entry. For example, if there are 4 Locators in a Locator-Set, where the Weights assigned are 30, 20, 20, and 10, the first Locator will get 37.5% of the traffic, the 2nd and 3rd Locators will get 25% of the traffic, and the 4th Locator will get 12.5% of the traffic. If all Weights for a Locator-Set are equal, the receiver of the Map-Reply will decide how to load-split the traffic. See Section 6.5 for a suggested hash algorithm to distribute the load across Locators with the same Priority and equal Weight values.

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 the RLOC MUST NOT be used for joining a multicast distribution tree. For more details, see [RFC6831].

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 relative weight (similar to the unicast Weights) of the total number of trees built to the source site identified by the EID-Prefix. If all Weights for a Locator-Set are equal, the receiver of the Map-Reply will decide how to distribute multicast state across ITRs. For more details, see [RFC6831].

Unused Flags: These are set to 0 when sending and ignored on receipt.

L: When this bit is set, the Locator is flagged as a local Locator to the ETR that is sending the Map-Reply. When a Map-Server is doing proxy Map-Replying [RFC6833] for a LISP site, the L-bit is set to 0 for all Locators in this Locator-Set.

p: When this bit is set, an ETR informs the RLOC-Probing ITR that the locator address for which this bit is set is the one being RLOC-probed and MAY be different from the source address of the Map-Reply. An ITR that RLOC-probes a particular Locator MUST use this Locator for retrieving the data structure used to store the fact that the Locator is reachable. The p-bit is set for a single Locator in the same Locator-Set. If an implementation sets more than one p-bit erroneously, the receiver of the Map-Reply MUST select the first Locator. The p-bit MUST NOT be set for Locator-Set records sent in Map-Request and Map-Register messages.

R: This is set when the sender of a Map-Reply has a route to the Locator in the Locator data record. This receiver may find this useful to know if the Locator is up but not necessarily reachable from the receiver's point of view. See also Section 6.4 for another way the R-bit may be used.

Locator: This is an IPv4 or IPv6 address (as encoded by the 'Loc-AFI' field) assigned to an ETR. Note that the destination RLOC address MAY be an anycast address. A source RLOC can be an anycast address as well. The source or destination RLOC MUST NOT be the broadcast address (255.255.255.255 or any subnet broadcast address known to the router) and MUST NOT be a link-local multicast address. The source RLOC MUST NOT be a multicast address. The destination RLOC SHOULD be a multicast address if it is being mapped from a multicast destination EID.

6.1.5. EID-to-RLOC UDP Map-Reply Message

A Map-Reply returns an EID-Prefix with a prefix length that is less than or equal to the EID being requested. The EID being requested is either from the destination field of an IP header of a Data-Probe or the EID record of a Map-Request. The RLOCs in the Map-Reply are globally routable IP addresses of all ETRs for the LISP site. Each RLOC conveys status reachability but does not convey path reachability from a requester's perspective. Separate testing of path reachability is required. See Section 6.3 for details.

Note that a Map-Reply may contain different EID-Prefix granularity (prefix + length) than the Map-Request that triggers it. This might occur if a Map-Request were for a prefix that had been returned by an earlier Map-Reply. In such a case, the requester updates its cache with the new prefix information and granularity. For example, a requester with two cached EID-Prefixes that are covered by a Map-Reply containing one less-specific prefix replaces the entry with the less-specific EID-Prefix. Note that the reverse, replacement of one less-specific prefix with multiple more-specific prefixes, can also occur, not by removing the less-specific prefix but rather by adding the more-specific prefixes that, during a lookup, will override the less-specific prefix.

When an ETR is configured with overlapping EID-Prefixes, a Map-Request with an EID that best matches any EID-Prefix MUST be returned in a single Map-Reply message. For instance, if an ETR had database mapping entries for EID-Prefixes:

10.0.0.0/8
10.1.0.0/16
10.1.1.0/24
10.1.2.0/24

A Map-Request for EID 10.1.1.1 would cause a Map-Reply with a record count of 1 to be returned with a mapping record EID-Prefix of 10.1.1.0/24.

A Map-Request for EID 10.1.5.5 would cause a Map-Reply with a record count of 3 to be returned with mapping records for EID-Prefixes 10.1.0.0/16, 10.1.1.0/24, and 10.1.2.0/24.

Note that not all overlapping EID-Prefixes need to be returned but only the more-specific entries (note that in the second example above 10.0.0.0/8 was not returned for requesting EID 10.1.5.5) for the matching EID-Prefix of the requesting EID. When more than one EID-Prefix is returned, all SHOULD use the same Time to Live value so they can all time out at the same time. When a more-specific EID-Prefix is received later, its Time to Live value in the Map-Reply record can be stored even when other less-specific entries exist. When a less-specific EID-Prefix is received later, its map-cache expiration time SHOULD be set to the minimum expiration time of any more-specific EID-Prefix in the map-cache. This is done so the integrity of the EID-Prefix set is wholly maintained and so no more-specific entries are removed from the map-cache while keeping less-specific entries.

Map-Replies SHOULD be sent for an EID-Prefix no more often 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 the EID addresses in the prefix range, thereby reducing the number of Map-Request messages.

Map-Reply records can have an empty Locator-Set. A Negative Map-Reply is a Map-Reply with an empty Locator-Set. Negative Map-Replies convey special actions by the sender to the ITR or PITR that have solicited the Map-Reply. There are two primary applications for Negative Map-Replies. The first is for a Map-Resolver to instruct an ITR or PITR when a destination is for a LISP site versus a non-LISP site, and the other is to source quench Map-Requests that are sent for non-allocated EIDs.

For each Map-Reply record, the list of Locators in a Locator-Set MUST appear in the same order for each ETR that originates a Map-Reply message. The Locator-Set MUST be sorted in order of ascending IP address where an IPv4 locator address is considered numerically 'less than' an IPv6 locator address.

When sending a Map-Reply message, the destination address is copied from one of the 'ITR-RLOC' fields from the Map-Request. The ETR can choose a locator address from one of the address families it supports. For Data-Probes, the destination address of the Map-Reply is copied from the source address of the Data-Probe message that is invoking the reply. The source address of the Map-Reply is one of the local IP addresses chosen to allow Unicast Reverse Path Forwarding (uRPF) checks to succeed in the upstream service provider. The destination port of a Map-Reply message is copied from the source port of the Map-Request or Data-Probe, and the source port of the Map-Reply message is set to the well-known UDP port 4342.

6.1.5.1. Traffic Redirection with Coarse EID-Prefixes

When an ETR is misconfigured or compromised, it could return coarse EID-Prefixes in Map-Reply messages it sends. The EID-Prefix could cover EID-Prefixes that are allocated to other sites, redirecting their traffic to the Locators of the compromised site.

To solve this problem, there are two basic solutions that could be used. The first is to have Map-Servers proxy Map-Reply on behalf of ETRs so their registered EID-Prefixes are the ones returned in Map-Replies. Since the interaction between an ETR and Map-Server is secured with shared keys, it is easier for an ETR to detect misbehavior. The second solution is to have ITRs and PITRs cache EID-Prefixes with mask lengths that are greater than or equal to a configured prefix length. This limits the damage to a specific width of any EID-Prefix advertised but needs to be coordinated with the allocation of site prefixes. These solutions can be used independently or at the same time.

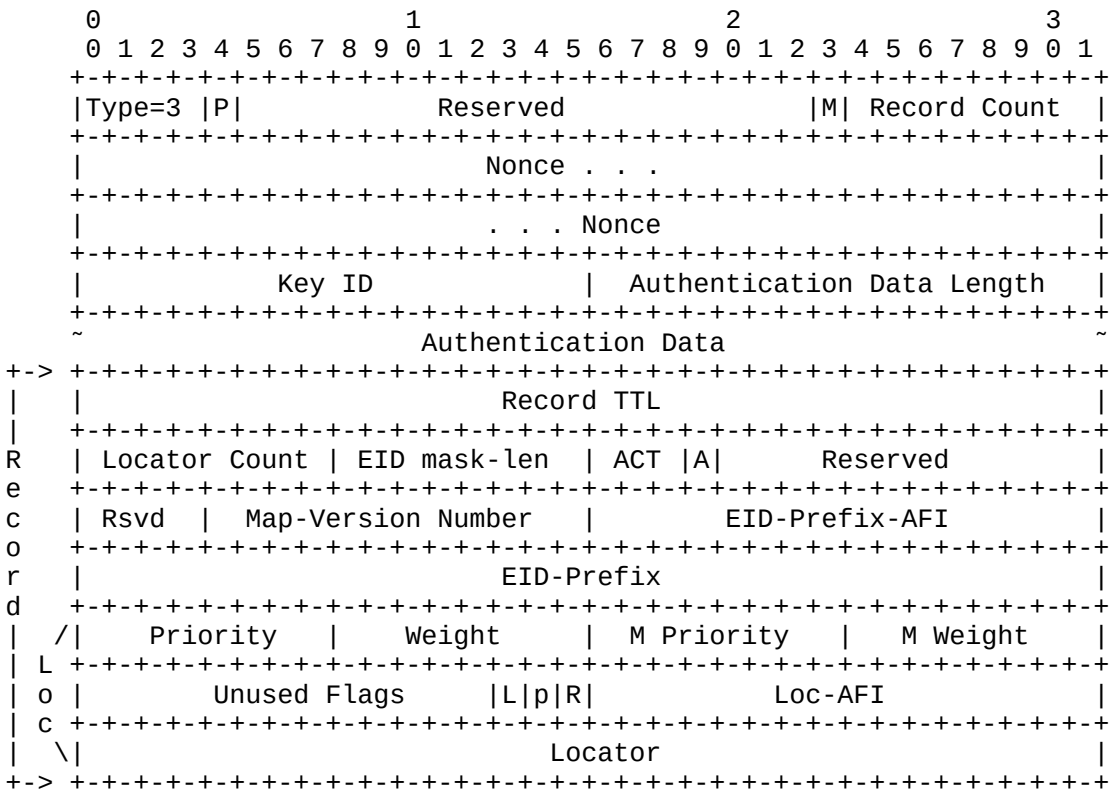
At the time of this writing, other approaches are being considered and researched.

6.1.6. Map-Register Message Format

The usage details of the Map-Register message can be found in specification [RFC6833]. This section solely defines the message format.

The message is sent in UDP with a destination UDP port of 4342 and a randomly selected UDP source port number.

The Map-Register message format is:



Packet field descriptions:

Type: 3 (Map-Register)

P: This is the proxy Map-Reply bit. When set to 1, an ETR sends a Map-Register message requesting the Map-Server to proxy a Map-Reply. The Map-Server will send non-authoritative Map-Replies on behalf of the ETR. Details on this usage can be found in [RFC6833].

Reserved: This field MUST be set to 0 on transmit and MUST be ignored on receipt.

M: This is the want-map-notify bit. When set to 1, an ETR is requesting a Map-Notify message to be returned in response to sending a Map-Register message. The Map-Notify message sent by a Map-Server is used to acknowledge receipt of a Map-Register message.

Record Count: This is the number of records in this Map-Register message. A record is comprised of that portion of the packet labeled 'Record' above and occurs the number of times equal to Record Count.

Nonce: This 8-octet 'Nonce' field is set to 0 in Map-Register messages. Since the Map-Register message is authenticated, the 'Nonce' field is not currently used for any security function but may be in the future as part of an anti-replay solution.

Key ID: This is a configured ID to find the configured Message Authentication Code (MAC) algorithm and key value used for the authentication function. See Section 14.4 for codepoint assignments.

Authentication Data Length: This is the length in octets of the 'Authentication Data' field that follows this field. The length of the 'Authentication Data' field is dependent on the MAC algorithm used. The length field allows a device that doesn't know the MAC algorithm to correctly parse the packet.

Authentication Data: This is the message digest used from the output of the MAC algorithm. The entire Map-Register payload is authenticated with this field preset to 0. After the MAC is computed, it is placed in this field. Implementations of this specification MUST include support for HMAC-SHA-1-96 [RFC2404], and support for HMAC-SHA-256-128 [RFC4868] is RECOMMENDED.

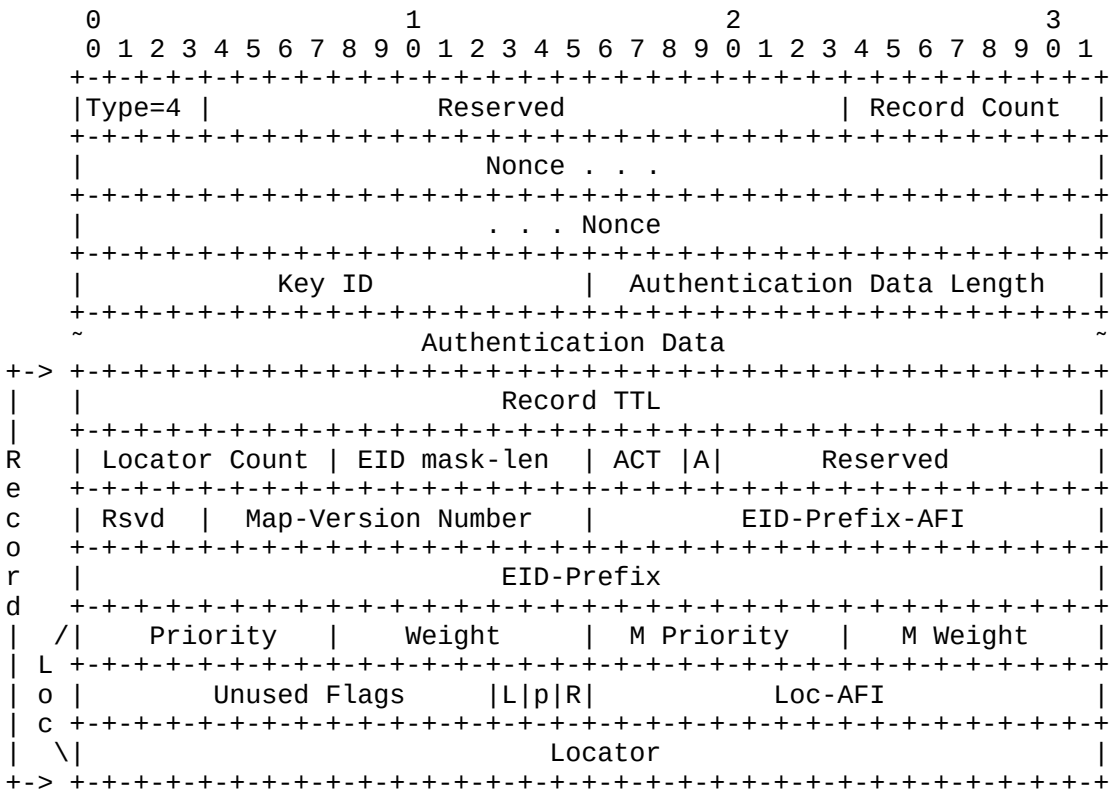
The definition of the rest of the Map-Register can be found in Section 6.1.4.

6.1.7. Map-Notify Message Format

The usage details of the Map-Notify message can be found in specification [RFC6833]. This section solely defines the message format.

The message is sent inside a UDP packet with source and destination UDP ports equal to 4342.

The Map-Notify message format is:



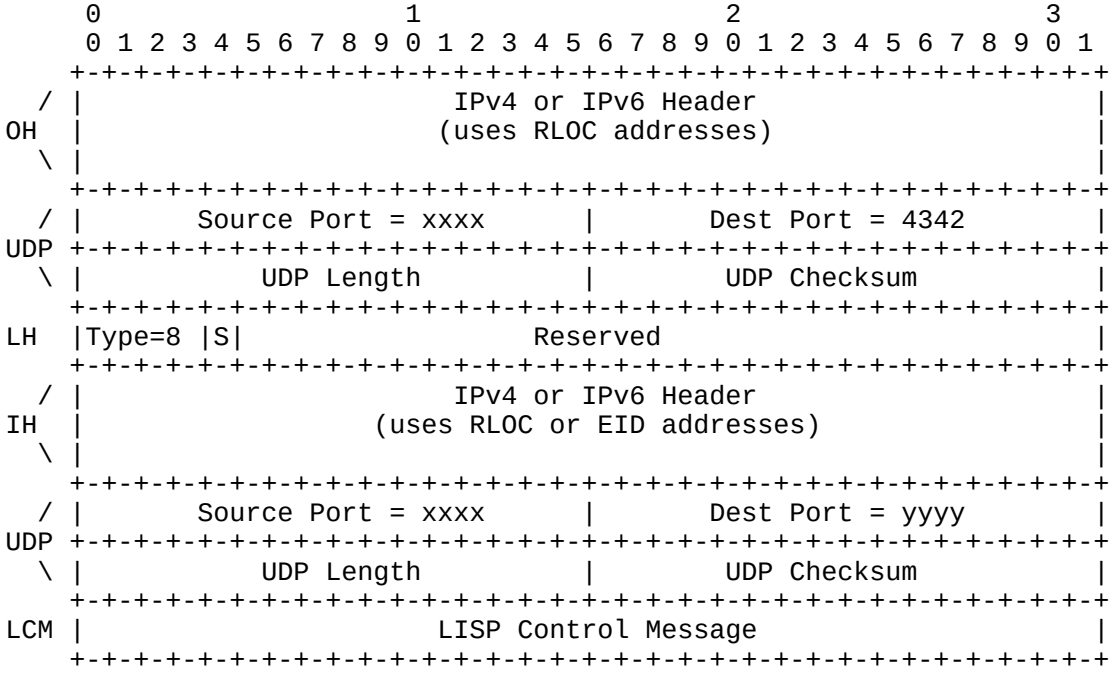
Packet field descriptions:

Type: 4 (Map-Notify)

The Map-Notify message has the same contents as a Map-Register message. See the Map-Register section for field descriptions.

6.1.8. Encapsulated Control Message Format

An Encapsulated Control Message (ECM) is used to encapsulate control packets sent between xTRs and the mapping database system described in [RFC6833].



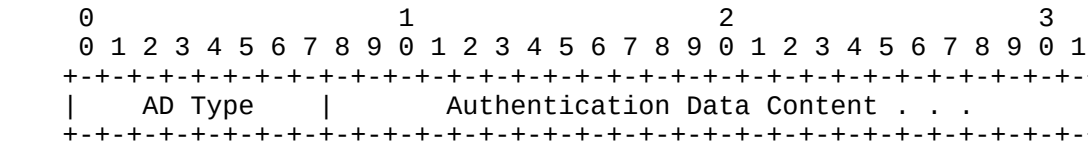
Packet header descriptions:

OH: The outer IPv4 or IPv6 header, which uses RLOC addresses in the source and destination header address fields.

UDP: The outer UDP header with destination port 4342. The source port is randomly allocated. The checksum field MUST be non-zero.

LH: Type 8 is defined to be a "LISP Encapsulated Control Message", and what follows is either an IPv4 or IPv6 header as encoded by the first 4 bits after the 'Reserved' field.

S: This is the Security bit. When set to 1, the field following the 'Reserved' field will have the following format. The detailed format of the Authentication Data Content is for further study.



IH: The inner IPv4 or IPv6 header, which can use either RLOC or EID addresses in the header address fields. When a Map-Request is encapsulated in this packet format, the destination address in this header is an EID.

UDP: The inner UDP header, where the port assignments depend on the control packet being encapsulated. When the control packet is a Map-Request or Map-Register, the source port is selected by the ITR/PITR and the destination port is 4342. When the control packet is a Map-Reply, the source port is 4342 and the destination port is assigned from the source port of the invoking Map-Request. Port number 4341 MUST NOT be assigned to either port. The checksum field MUST be non-zero.

LCM: The format is one of the control message formats described in this section. At this time, only Map-Request messages are allowed to be encapsulated. In the future, PIM Join/Prune messages [RFC6831] might be allowed. Encapsulating other types of LISP control messages is for further study. When Map-Requests are sent for RLOC-Probing purposes (i.e., the probe-bit is set), they MUST NOT be sent inside Encapsulated Control Messages.

6.2. Routing Locator Selection

Both the client-side and server-side may need control over the selection of RLOCs for conversations between them. This control 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 are different scenarios for choosing RLOCs and the controls that are available:

- The server-side returns one RLOC. The client-side can only use one RLOC. The server-side has complete control of the selection.

- The server-side returns a list of RLOCs where a subset of the list has the same best Priority. The client can only use the subset list according to the weighting assigned by the server-side. In this case, the server-side controls both the subset list and

load-splitting across its members. The client-side can use RLOCs outside of the subset list if it determines that the subset list is unreachable (unless RLOCs are set to a Priority of 255). Some sharing of control exists: the server-side determines the destination RLOC list and load distribution while the client-side has the option of using alternatives to this list if RLOCs in the list are unreachable.

- The server-side sets a Weight of 0 for the RLOC subset list. In this case, 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. Again, the client can use alternative RLOCs if the server-provided list of RLOCs is unreachable.

- Either side (more likely the server-side ETR) decides not to send a Map-Request. For example, if the server-side ETR does not send Map-Requests, it gleans RLOCs from the client-side ITR, giving the client-side ITR responsibility for bidirectional RLOC reachability and preferability. Server-side ETR gleaning of the client-side ITR RLOC is done by caching the inner-header source EID and the outer-header source RLOC of received packets. The client-side ITR controls how traffic is returned and can alternate using an outer-header source RLOC, which then can be added to the list the server-side ETR uses to return traffic. Since no Priority or Weights are provided using this method, the server-side ETR MUST assume that each client-side ITR RLOC uses the same best Priority with a Weight of zero. In addition, since EID-Prefix encoding cannot be conveyed in data packets, the EID-to-RLOC Cache on Tunnel Routers can grow to be very large.

- A "gleaned" Map-Cache entry, one learned from the source RLOC of a received encapsulated packet, is only stored and used for a few seconds, pending verification. Verification is performed by sending a Map-Request to the source EID (the inner-header IP source address) of the received encapsulated packet. A reply to this "verifying Map-Request" is used to fully populate the Map-Cache entry for the "gleaned" EID and is stored and used for the time indicated from the 'TTL' field of a received Map-Reply. When a verified Map-Cache entry is stored, data gleaning no longer occurs for subsequent packets that have a source EID that matches the EID-Prefix of the verified entry.

RLOCs that appear in EID-to-RLOC Map-Reply messages are assumed to be reachable when the R-bit for the Locator record is set to 1. When the R-bit is set to 0, an ITR or PITR MUST NOT encapsulate to the RLOC. Neither the information contained in a Map-Reply nor that stored in the mapping database system provides reachability

information for RLOCs. Note that reachability is not part of the mapping system and is determined using one or more of the Routing Locator reachability algorithms described in the next section.

6.3. Routing Locator Reachability

Several mechanisms for determining RLOC reachability are currently defined:

1. An ETR may examine the Locator-Status-Bits in the LISP header of an encapsulated data packet received from an ITR. If the ETR is also acting as an ITR and has traffic to return to the original ITR site, it can use this status information to help select an RLOC.

2. An ITR may receive an ICMP Network Unreachable or Host Unreachable message for an RLOC it is using. This indicates that the RLOC is likely down. Note that trusting ICMP messages may not be desirable, but neither is ignoring them completely. Implementations are encouraged to follow current best practices in treating these conditions.

3. An ITR that participates in the global routing system can determine that an RLOC is down if no BGP Routing Information Base (RIB) route exists that matches the RLOC IP address.

4. An ITR may receive an ICMP Port Unreachable message from a destination host. This occurs if an ITR attempts to use interworking [RFC6832] and LISP-encapsulated data is sent to a non-LISP-capable site.

5. An ITR may receive a Map-Reply from an ETR in response to a previously sent Map-Request. The RLOC source of the Map-Reply is likely up, since the ETR was able to send the Map-Reply to the ITR.

6. When an ETR receives an encapsulated packet from an ITR, the source RLOC from the outer header of the packet is likely up.

7. An ITR/ETR pair can use the Locator reachability algorithms described in this section, namely Echo-Noncing or RLOC-Probing.

When determining Locator up/down reachability by examining the Locator-Status-Bits from the LISP-encapsulated data packet, an ETR will receive up-to-date status from an encapsulating ITR about reachability for all ETRs at the site. CE-based ITRs at the source site can determine reachability relative to each other using the site IGP as follows:

- Under normal circumstances, each ITR will advertise a default route into the site IGP.
- If an ITR fails or if the upstream link to its PE fails, its default route will either time out or be withdrawn.

Each ITR can thus observe the presence or lack of a default route originated by the others to determine the Locator-Status-Bits it sets for them.

RLOCs listed in a Map-Reply are numbered with ordinals 0 to n-1. The Locator-Status-Bits in a LISP-encapsulated packet are numbered from 0 to n-1 starting with the least significant bit. For example, if an RLOC listed in the 3rd position of the Map-Reply goes down (ordinal value 2), then all ITRs at the site will clear the 3rd least significant bit (xxxx x0xx) of the 'Locator-Status-Bits' field for the packets they encapsulate.

When an ETR decapsulates a packet, it will check for any change in the 'Locator-Status-Bits' field. When a bit goes from 1 to 0, the ETR, if acting also as an ITR, will refrain from encapsulating packets to an RLOC that is indicated as down. It will only resume using that RLOC if the corresponding Locator-Status-Bit returns to a value of 1. Locator-Status-Bits are associated with a Locator-Set per EID-Prefix. Therefore, when a Locator becomes unreachable, the Locator-Status-Bit that corresponds to that Locator's position in the list returned by the last Map-Reply will be set to zero for that particular EID-Prefix.

When ITRs at the site are not deployed in CE routers, the IGP can still be used to determine the reachability of Locators, provided they are injected into the IGP. This is typically done when a /32 address is configured on a loopback interface.

When ITRs receive ICMP Network Unreachable or Host Unreachable messages as a method to determine unreachability, they will refrain from using Locators that are described in Locator lists of Map-Replies. However, using this approach is unreliable because many network operators turn off generation of ICMP Destination Unreachable messages.

If an ITR does receive an ICMP Network Unreachable or Host Unreachable message, it MAY originate its own ICMP Destination Unreachable message destined for the host that originated the data packet the ITR encapsulated.

Also, BGP-enabled ITRs can unilaterally examine the RIB to see if a locator address from a Locator-Set in a mapping entry matches a prefix. If it does not find one and BGP is running in the Default-Free Zone (DFZ), it can decide to not use the Locator even though the Locator-Status-Bits indicate that the Locator is up. In this case, the path from the ITR to the ETR that is assigned the Locator is not available. More details are in [LOC-ID-ARCH].

Optionally, an ITR can send a Map-Request to a Locator, and if a Map-Reply is returned, reachability of the Locator has been determined. Obviously, sending such probes increases the number of control messages originated by Tunnel Routers for active flows, so Locators are assumed to be reachable when they are advertised.

This assumption does create a dependency: Locator unreachability is detected by the receipt of ICMP Host Unreachable messages. When a Locator has been determined to be unreachable, it is not used for active traffic; this is the same as if it were listed in a Map-Reply with Priority 255.

The ITR can test the reachability of the unreachable Locator by sending periodic Requests. Both Requests and Replies MUST be rate-limited. Locator reachability testing is never done with data packets, since that increases the risk of packet loss for end-to-end sessions.

When an ETR decapsulates a packet, it knows that it is reachable from the encapsulating ITR because that is how the packet arrived. In most cases, the ETR can also reach the ITR but cannot assume this to be true, due to the possibility of path asymmetry. In the presence of unidirectional traffic flow from an ITR to an ETR, the ITR SHOULD NOT use the lack of return traffic as an indication that the ETR is unreachable. Instead, it MUST use an alternate mechanism to determine reachability.

6.3.1. Echo Nonce Algorithm

When data flows bidirectionally between Locators from different sites, a data-plane mechanism called "nonce echoing" can be used to determine reachability between an ITR and ETR. When an ITR wants to solicit a nonce echo, it sets the N- and E-bits and places a 24-bit nonce [RFC4086] in the LISP header of the next encapsulated data packet.

When this packet is received by the ETR, the encapsulated packet is forwarded as normal. When the ETR next sends a data packet to the ITR, it includes the nonce received earlier with the N-bit set and E-bit cleared. The ITR sees this "echoed nonce" and knows that the path to and from the ETR is up.

The ITR will set the E-bit and N-bit for every packet it sends while in the echo-nonce-request state. The time the ITR waits to process the echoed nonce before it determines the path is unreachable is variable and is a choice left for the implementation.

If the ITR is receiving packets from the ETR but does not see the nonce echoed while being in the echo-nonce-request state, then the path to the ETR is unreachable. This decision may be overridden by other Locator reachability algorithms. Once the ITR determines that the path to the ETR is down, it can switch to another Locator for that EID-Prefix.

Note that "ITR" and "ETR" are relative terms here. Both devices MUST be implementing both ITR and ETR functionality for the echo nonce mechanism to operate.

The ITR and ETR may both go into the echo-nonce-request state at the same time. The number of packets sent or the time during which echo nonce requests are sent is an implementation-specific setting. However, when an ITR is in the echo-nonce-request state, it can echo the ETR’s nonce in the next set of packets that it encapsulates and subsequently continue sending echo-nonce-request packets.

This mechanism does not completely solve the forward path reachability problem, as traffic may be unidirectional. That is, the ETR receiving traffic at a site may not be the same device as an ITR that transmits traffic from that site, or the site-to-site traffic is unidirectional so there is no ITR returning traffic.

The echo-nonce algorithm is bilateral. That is, if one side sets the E-bit and the other side is not enabled for echo-noncing, then the echoing of the nonce does not occur and the requesting side may erroneously consider the Locator unreachable. An ITR SHOULD only set the E-bit in an encapsulated data packet when it knows the ETR is enabled for echo-noncing. This is conveyed by the E-bit in the Map-Reply message.

Note that other Locator reachability mechanisms are being researched and can be used to compliment or even override the echo nonce algorithm. See the next section for an example of control-plane probing.

6.3.2. RLOC-Probing Algorithm

RLOC-Probing is a method that an ITR or PITR can use to determine the reachability status of one or more Locators that it has cached in a Map-Cache entry. The probe-bit of the Map-Request and Map-Reply messages is used for RLOC-Probing.

RLOC-Probing is done in the control plane on a timer basis, where an ITR or PITR will originate a Map-Request destined to a locator address from one of its own locator addresses. A Map-Request used as an RLOC-probe is NOT encapsulated and NOT sent to a Map-Server or to the mapping database system as one would when soliciting mapping data. The EID record encoded in the Map-Request is the EID-Prefix of the Map-Cache entry cached by the ITR or PITR. The ITR may include a mapping data record for its own database mapping information that contains the local EID-Prefixes and RLOCs for its site. RLOC-probes are sent periodically using a jittered timer interval.

When an ETR receives a Map-Request message with the probe-bit set, it returns a Map-Reply with the probe-bit set. The source address of the Map-Reply is set according to the procedure described in Section 6.1.5. The Map-Reply SHOULD contain mapping data for the EID-Prefix contained in the Map-Request. This provides the opportunity for the ITR or PITR that sent the RLOC-probe to get mapping updates if there were changes to the ETR’s database mapping entries.

There are advantages and disadvantages of RLOC-Probing. The greatest benefit of RLOC-Probing is that it can handle many failure scenarios allowing the ITR to determine when the path to a specific Locator is reachable or has become unreachable, thus providing a robust mechanism for switching to using another Locator from the cached Locator. RLOC-Probing can also provide rough Round-Trip Time (RTT) estimates between a pair of Locators, which can be useful for network management purposes as well as for selecting low delay paths. The major disadvantage of RLOC-Probing is in the number of control messages required and the amount of bandwidth used to obtain those benefits, especially if the requirement for failure detection times is very small.

Continued research and testing will attempt to characterize the tradeoffs of failure detection times versus message overhead.

6.4. EID Reachability within a LISP Site

A site may be multihomed using two or more ETRs. The hosts and infrastructure within a site will be addressed using one or more EID-Prefixes that are mapped to the RLOCs of the relevant ETRs in the mapping system. One possible failure mode is for an ETR to lose reachability to one or more of the EID-Prefixes within its own site. When this occurs when the ETR sends Map-Replies, it can clear the R-bit associated with its own Locator. And when the ETR is also an ITR, it can clear its Locator-Status-Bit in the encapsulation data header.

It is recognized that there are no simple solutions to the site partitioning problem because it is hard to know which part of the EID-Prefix range is partitioned and which Locators can reach any sub-ranges of the EID-Prefixes. This problem is under investigation with the expectation that experiments will tell us more. Note that this is not a new problem introduced by the LISP architecture. The problem exists today when a multihomed site uses BGP to advertise its reachability upstream.

6.5. 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. When more than one best Priority Locator exists, the ITR can decide how to load-share traffic against the corresponding Locators.

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:

1. Either a source and destination address hash or the traditional 5-tuple hash can be used. The traditional 5-tuple hash includes the source and destination addresses; source and destination TCP, UDP, or Stream Control Transmission Protocol (SCTP) port numbers; and the IP protocol number field or IPv6 next-protocol fields of a packet that a host originates from within a LISP site. When a packet is not a TCP, UDP, or SCTP packet, the source and destination addresses only from the header are used to compute the hash.

2. Take the hash value and divide it by the number of Locators stored in the Locator-Set for the EID-to-RLOC mapping.

3. The remainder will yield a value of 0 to "number of Locators minus 1". Use the remainder to select the Locator in the Locator-Set.

Note that when a packet is LISP encapsulated, the source port number in the outer UDP header needs to be set. Selecting a hashed value allows core routers that are attached to Link Aggregation Groups (LAGs) to load-split the encapsulated packets across member links of such LAGs. Otherwise, core routers would see a single flow, since packets have a source address of the ITR, for packets that are originated by different EIDs at the source site. A suggested setting for the source port number computed by an ITR is a 5-tuple hash function on the inner header, as described above.

Many core router implementations use a 5-tuple hash to decide how to balance packet load across members of a LAG. The 5-tuple hash includes the source and destination addresses of the packet and the source and destination ports when the protocol number in the packet is TCP or UDP. For this reason, UDP encoding is used for LISP encapsulation.

6.6. Changing the Contents of EID-to-RLOC Mappings

Since the LISP architecture uses a caching scheme to retrieve and store EID-to-RLOC mappings, the only way an ITR can get a more up-to-date mapping is to re-request the mapping. However, the ITRs do not know when the mappings change, and the ETRs do not keep track of which ITRs requested its mappings. For scalability reasons, we want to maintain this approach but need to provide a way for ETRs to change their mappings and inform the sites that are currently communicating with the ETR site using such mappings.

When adding a new Locator record in lexicographic order to the end of a Locator-Set, it is easy to update mappings. We assume that new mappings will maintain the same Locator ordering as the old mapping but will just have new Locators appended to the end of the list. So, some ITRs can have a new mapping while other ITRs have only an old mapping that is used until they time out. When an ITR has only an old mapping but detects bits set in the Locator-Status-Bits that correspond to Locators beyond the list it has cached, it simply ignores them. However, this can only happen for locator addresses that are lexicographically greater than the locator addresses in the existing Locator-Set.

When a Locator record is inserted in the middle of a Locator-Set, to maintain lexicographic order, the SMR procedure in Section 6.6.2 is used to inform ITRs and PITRs of the new Locator-Status-Bit mappings.

When a Locator record is removed from a Locator-Set, ITRs that have the mapping cached will not use the removed Locator because the xTRs will set the Locator-Status-Bit to 0. So, even if the Locator is in the list, it will not be used. For new mapping requests, the xTRs

can set the Locator AFI to 0 (indicating an unspecified address), as well as setting the corresponding Locator-Status-Bit to 0. This forces ITRs with old or new mappings to avoid using the removed Locator.

If many changes occur to a mapping over a long period of time, one will find empty record slots in the middle of the Locator-Set and new records appended to the Locator-Set. At some point, it would be useful to compact the Locator-Set so the Locator-Status-Bit settings can be efficiently packed.

We propose here three approaches for Locator-Set compaction: one operational mechanism and two protocol mechanisms. The operational approach uses a clock sweep method. The protocol approaches use the concept of Solicit-Map-Requests and Map-Versioning.

6.6.1. Clock Sweep

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

1. 24 hours before a mapping change is to take effect, a network administrator configures the ETRs at a site to start the clock sweep window.

2. During the clock sweep window, ETRs continue to send Map-Reply messages with the current (unchanged) mapping records. The TTL for these mappings is set to 1 hour.

3. 24 hours later, all previous cache entries will have timed out, and any active cache entries will time out within 1 hour. During this 1-hour window, the ETRs continue to send Map-Reply messages with the current (unchanged) mapping records with the TTL set to 1 minute.

4. At the end of the 1-hour window, the ETRs will send Map-Reply messages with the new (changed) mapping records. So, any active caches can get the new mapping contents right away if not cached, or in 1 minute if they had the mapping cached. The new mappings are cached with a TTL equal to the TTL in the Map-Reply.

6.6.2. Solicit-Map-Request (SMR)

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

Since the ETRs don't keep track of remote ITRs that have cached their mappings, they do not know which ITRs need to have their mappings updated. As a result, an ETR will solicit Map-Requests (called an SMR message) from those sites to which it has been sending encapsulated data for the last minute. In particular, an ETR will send an SMR to an ITR to which it has recently sent encapsulated data.

An SMR message is simply a bit set in a Map-Request message. An ITR or PITR will send a Map-Request when they receive an SMR message. Both the SMR sender and the Map-Request responder MUST rate-limit these messages. Rate-limiting can be implemented as a global rate-limiter or one rate-limiter per SMR destination.

The following procedure shows how an SMR exchange occurs when a site is doing Locator-Set compaction for an EID-to-RLOC mapping:

1. When the database mappings in an ETR change, the ETRs at the site begin to send Map-Requests with the SMR bit set for each Locator in each Map-Cache entry the ETR caches.

2. A remote ITR that receives the SMR message will schedule sending a Map-Request message to the source locator address of the SMR message or to the mapping database system. A newly allocated random nonce is selected, and the EID-Prefix used is the one copied from the SMR message. If the source Locator is the only Locator in the cached Locator-Set, the remote ITR SHOULD send a Map-Request to the database mapping system just in case the single Locator has changed and may no longer be reachable to accept the Map-Request.

3. The remote ITR MUST rate-limit the Map-Request until it gets a Map-Reply while continuing to use the cached mapping. When Map-Versioning as described in Section 6.6.3 is used, an SMR sender can detect if an ITR is using the most up-to-date database mapping.

4. The ETRs at the site with the changed mapping will reply to the Map-Request with a Map-Reply message that has a nonce from the SMR-invoked Map-Request. The Map-Reply messages SHOULD be rate-limited. This is important to avoid Map-Reply implosion.

5. The ETRs at the site with the changed mapping record the fact that the site that sent the Map-Request has received the new mapping data in the Map-Cache entry for the remote site so the Locator-Status-Bits are reflective of the new mapping for packets going to the remote site. The ETR then stops sending SMR messages.

Experimentation is in progress to determine the appropriate rate-limit parameters.

For security reasons, an ITR MUST NOT process unsolicited Map-Replies. To avoid Map-Cache entry corruption by a third party, a sender of an SMR-based Map-Request MUST be verified. If an ITR receives an SMR-based Map-Request and the source is not in the Locator-Set for the stored Map-Cache entry, then the responding Map-Request MUST be sent with an EID destination to the mapping database system. Since the mapping database system is a more secure way to reach an authoritative ETR, it will deliver the Map-Request to the authoritative source of the mapping data.

When an ITR receives an SMR-based Map-Request for which it does not have a cached mapping for the EID in the SMR message, it MAY not send an SMR-invoked Map-Request. This scenario can occur when an ETR sends SMR messages to all Locators in the Locator-Set it has stored in its map-cache but the remote ITRs that receive the SMR may not be sending packets to the site. There is no point in updating the ITRs until they need to send, in which case they will send Map-Requests to obtain a Map-Cache entry.

6.6.3. Database Map-Versioning

When there is unidirectional packet flow between an ITR and ETR, and the EID-to-RLOC mappings change on the ETR, it needs to inform the ITR so encapsulation to a removed Locator can stop and can instead be started to a new Locator in the Locator-Set.

An ETR, when it sends Map-Reply messages, conveys its own Map-Version Number. This is known as the Destination Map-Version Number. ITRs include the Destination Map-Version Number in packets they encapsulate to the site. When an ETR decapsulates a packet and detects that the Destination Map-Version Number is less than the current version for its mapping, the SMR procedure described in Section 6.6.2 occurs.

An ITR, when it encapsulates packets to ETRs, can convey its own Map-Version Number. This is known as the Source Map-Version Number. When an ETR decapsulates a packet and detects that the Source Map-Version Number is greater than the last Map-Version Number sent in a Map-Reply from the ITR's site, the ETR will send a Map-Request to one of the ETRs for the source site.

A Map-Version Number is used as a sequence number per EID-Prefix, so values that are greater are considered to be more recent. A value of 0 for the Source Map-Version Number or the Destination Map-Version Number conveys no versioning information, and an ITR does no comparison with previously received Map-Version Numbers.

A Map-Version Number can be included in Map-Register messages as well. This is a good way for the Map-Server to assure that all ETRs for a site registering to it will be synchronized according to Map-Version Number.

See [RFC6834] for a more detailed analysis and description of Database Map-Versioning.

7. Router Performance Considerations

LISP is designed to be very "hardware-based forwarding friendly". A few implementation techniques can be used to incrementally implement LISP:

- When a tunnel-encapsulated packet is received by an ETR, the outer destination address may not be the address of the router. This makes it challenging for the control plane to get packets from the hardware. This may be mitigated by creating special Forwarding Information Base (FIB) entries for the EID-Prefixes of EIDs served by the ETR (those for which the router provides an RLOC translation). These FIB entries are marked with a flag indicating that control-plane processing should be performed. The forwarding logic of testing for particular IP protocol number values is not necessary. There are a few proven cases where no changes to existing deployed hardware were needed to support the LISP data-plane.

- On an ITR, prepending a new IP header consists of adding more octets to a MAC rewrite string and prepending the string as part of the outgoing encapsulation procedure. Routers that support Generic Routing Encapsulation (GRE) tunneling [RFC2784] or 6to4 tunneling [RFC3056] may already support this action.

- A packet’s source address or interface the packet was received on can be used to select VRF (Virtual Routing/Forwarding). The VRF’s routing table can be used to find EID-to-RLOC mappings.

For performance issues related to map-cache management, see Section 12.

8. Deployment Scenarios

This section will explore how and where ITRs and ETRs can be deployed and will discuss the pros and cons of each deployment scenario. For a more detailed deployment recommendation, refer to [LISP-DEPLOY].

There are two basic deployment tradeoffs to consider: centralized versus distributed caches; and flat, Recursive, or Re-encapsulating Tunneling. When deciding on centralized versus distributed caching, the following issues should be considered:

- Are the Tunnel Routers spread out so that the caches are spread across all the memories of each router? A centralized cache is when an ITR keeps a cache for all the EIDs it is encapsulating to. The packet takes a direct path to the destination Locator. A distributed cache is when an ITR needs help from other re-encapsulating routers because it does not store all the cache entries for the EIDs it is encapsulating to. So, the packet takes a path through re-encapsulating routers that have a different set of cache entries.

- Should management "touch points" be minimized by only choosing a few Tunnel Routers, just enough for redundancy?

- In general, using more ITRs doesn’t increase management load, since caches are built and stored dynamically. On the other hand, using more ETRs does require more management, since EID-Prefix-to-RLOC mappings need to be explicitly configured.

When deciding on flat, Recursive, or Re-encapsulating Tunneling, the following issues should be considered:

- Flat tunneling implements a single tunnel between the source site and destination site. This generally offers better paths between sources and destinations with a single tunnel path.

- Recursive Tunneling is when tunneled traffic is again further encapsulated in another tunnel, either to implement VPNs or to perform Traffic Engineering. When doing VPN-based tunneling, the site has some control, since the site is prepending a new tunnel header. In the case of TE-based tunneling, the site may have

control if it is prepending a new tunnel header, but if the site's ISP is doing the TE, then the site has no control. Recursive Tunneling generally will result in suboptimal paths but with the benefit of steering traffic to parts of the network that have more resources available.

- The technique of re-encapsulation ensures that packets only require one tunnel header. So, if a packet needs to be re-routed, it is first decapsulated by the ETR and then re-encapsulated with a new tunnel header using a new RLOC.

The next sub-sections will examine where Tunnel Routers can reside in the network.

8.1. First-Hop/Last-Hop Tunnel Routers

By locating Tunnel Routers close to hosts, the EID-Prefix set is at the granularity of an IP subnet. So, at the expense of more EID-Prefix-to-RLOC sets for the site, the caches in each Tunnel Router can remain relatively small. But caches always depend on the number of non-aggregated EID destination flows active through these Tunnel Routers.

With more Tunnel Routers doing encapsulation, the increase in control traffic grows as well: since the EID granularity is greater, more Map-Requests and Map-Replies are traveling between more routers.

The advantage of placing the caches and databases at these stub routers is that the products deployed in this part of the network have better price-memory ratios than their core router counterparts. Memory is typically less expensive in these devices, and fewer routes are stored (only IGP routes). These devices tend to have excess capacity, both for forwarding and routing states.

LISP functionality can also be deployed in edge switches. These devices generally have layer-2 ports facing hosts and layer-3 ports facing the Internet. Spare capacity is also often available in these devices.

8.2. Border/Edge Tunnel Routers

Using Customer Edge (CE) routers for tunnel endpoints allows the EID space associated with a site to be reachable via a small set of RLOCs assigned to the CE routers for that site. This is the default behavior envisioned in the rest of this specification.

This offers the opposite benefit of the first-hop/last-hop Tunnel Router scenario: the number of mapping entries and network management touch points is reduced, allowing better scaling.

One disadvantage is that fewer network resources are used to reach host endpoints, thereby centralizing the point-of-failure domain and creating network choke points at the CE router.

Note that more than one CE router at a site can be configured with the same IP address. In this case, an RLOC is an anycast address. This allows resilience between the CE routers. That is, if a CE router fails, traffic is automatically routed to the other routers using the same anycast address. However, this comes with the disadvantage where the site cannot control the entrance point when the anycast route is advertised out from all border routers. Another disadvantage of using anycast Locators is the limited advertisement scope of /32 (or /128 for IPv6) routes.

8.3. ISP Provider Edge (PE) Tunnel Routers

The use of ISP PE routers as tunnel endpoint routers is not the typical deployment scenario envisioned in this specification. This section attempts to capture some of the reasoning behind this preference for implementing LISP on CE routers.

The use of ISP PE routers as tunnel endpoint routers gives an ISP, rather than a site, control over the location of the egress tunnel endpoints. That is, the ISP can decide whether the tunnel endpoints are in the destination site (in either CE routers or last-hop routers within a site) or at other PE edges. The advantage of this case is that two tunnel headers can be avoided. By having the PE be the first router on the path to encapsulate, it can choose a TE path first, and the ETR can decapsulate and re-encapsulate for a tunnel to the destination end site.

An obvious disadvantage is that the end site has no control over where its packets flow or over the RLOCs used. Other disadvantages include difficulty in synchronizing path liveness updates between CE and PE routers.

As mentioned in earlier sections, a combination of these scenarios is possible at the expense of extra packet header overhead; if both site and provider want control, then Recursive or Re-encapsulating Tunnels are used.

8.4. LISP Functionality with Conventional NATs

LISP routers can be deployed behind Network Address Translator (NAT) devices to provide the same set of packet services hosts have today when they are addressed out of private address space.

It is important to note that a locator address in any LISP control message MUST be a globally routable address and therefore SHOULD NOT contain [RFC1918] addresses. If a LISP router is configured with private addresses, they MUST be used only in the outer IP header so the NAT device can translate properly. Otherwise, EID addresses MUST be translated before encapsulation is performed. Both NAT translation and LISP encapsulation functions could be co-located in the same device.

More details on LISP address translation can be found in [RFC6832].

8.5. Packets Egressing a LISP Site

When a LISP site is using two ITRs for redundancy, the failure of one ITR will likely shift outbound traffic to the second. This second ITR's cache may not be populated with the same EID-to-RLOC mapping entries as the first. If this second ITR does not have these mappings, traffic will be dropped while the mappings are retrieved from the mapping system. The retrieval of these messages may increase the load of requests being sent into the mapping system. Deployment and experimentation will determine whether this issue requires more attention.

9. Traceroute Considerations

When a source host in a LISP site initiates a traceroute to a destination host in another LISP site, it is highly desirable for it to see the entire path. Since packets are encapsulated from the ITR to the ETR, the hop across the tunnel could be viewed as a single hop. However, LISP traceroute will provide the entire path so the user can see 3 distinct segments of the path from a source LISP host to a destination LISP host:

Segment 1 (in source LISP site based on EIDs):

source host ---> first hop ... next hop ---> ITR

Segment 2 (in the core network based on RLOCs):

ITR ---> next hop ... next hop ---> ETR

Segment 3 (in the destination LISP site based on EIDs):

ETR ---> next hop ... last hop ---> destination host

For segment 1 of the path, ICMP Time Exceeded messages are returned in the normal manner as they are today. The ITR performs a TTL decrement and tests for 0 before encapsulating. Therefore, the ITR's hop is seen by the traceroute source as having an EID address (the address of the site-facing interface).

For segment 2 of the path, ICMP Time Exceeded messages are returned to the ITR because the TTL decrement to 0 is done on the outer header, so the destinations of the ICMP messages are the ITR RLOC address and the source RLOC address of the encapsulated traceroute packet. The ITR looks inside of the ICMP payload to inspect the traceroute source so it can return the ICMP message to the address of the traceroute client and also retain the core router IP address in the ICMP message. This is so the traceroute client can display the core router address (the RLOC address) in the traceroute output. The ETR returns its RLOC address and responds to the TTL decrement to 0, as the previous core routers did.

For segment 3, the next-hop router downstream from the ETR will be decrementing the TTL for the packet that was encapsulated, sent into the core, decapsulated by the ETR, and forwarded because it isn't the final destination. If the TTL is decremented to 0, any router on the path to the destination of the traceroute, including the next-hop router or destination, will send an ICMP Time Exceeded message to the source EID of the traceroute client. The ICMP message will be encapsulated by the local ITR and sent back to the ETR in the originated traceroute source site, where the packet will be delivered to the host.

9.1. IPv6 Traceroute

IPv6 traceroute follows the procedure described above, since the entire traceroute data packet is included in the ICMP Time Exceeded message payload. Therefore, only the ITR needs to pay special attention to forwarding ICMP messages back to the traceroute source.

9.2. IPv4 Traceroute

For IPv4 traceroute, we cannot follow the above procedure, since IPv4 ICMP Time Exceeded messages only include the invoking IP header and 8 octets that follow the IP header. Therefore, when a core router sends an IPv4 Time Exceeded message to an ITR, all the ITR has in the ICMP payload is the encapsulated header it prepended, followed by a UDP header. The original invoking IP header, and therefore the identity of the traceroute source, is lost.

The solution we propose to solve this problem is to cache traceroute IPv4 headers in the ITR and to match them up with corresponding IPv4 Time Exceeded messages received from core routers and the ETR. The ITR will use a circular buffer for caching the IPv4 and UDP headers of traceroute packets. It will select a 16-bit number as a key to find them later when the IPv4 Time Exceeded messages are received. When an ITR encapsulates an IPv4 traceroute packet, it will use the 16-bit number as the UDP source port in the encapsulating header. When the ICMP Time Exceeded message is returned to the ITR, the UDP header of the encapsulating header is present in the ICMP payload, thereby allowing the ITR to find the cached headers for the traceroute source. The ITR puts the cached headers in the payload and sends the ICMP Time Exceeded message to the traceroute source retaining the source address of the original ICMP Time Exceeded message (a core router or the ETR of the site of the traceroute destination).

The signature of a traceroute packet comes in two forms. The first form is encoded as a UDP message where the destination port is inspected for a range of values. The second form is encoded as an ICMP message where the IP identification field is inspected for a well-known value.

9.3. Traceroute Using Mixed Locators

When either an IPv4 traceroute or IPv6 traceroute is originated and the ITR encapsulates it in the other address family header, one cannot get all 3 segments of the traceroute. Segment 2 of the traceroute cannot be conveyed to the traceroute source, since it is expecting addresses from intermediate hops in the same address format for the type of traceroute it originated. Therefore, in this case, segment 2 will make the tunnel look like one hop. All the ITR has to do to make this work is to not copy the inner TTL to the outer, encapsulating header's TTL when a traceroute packet is encapsulated using an RLOC from a different address family. This will cause no TTL decrement to 0 to occur in core routers between the ITR and ETR.

10. Mobility Considerations

There are several kinds of mobility, of which only some might be of concern to LISP. Essentially, they are as follows.

10.1. Site Mobility

A site wishes to change its attachment points to the Internet, and its LISP Tunnel Routers will have new RLOCs when it changes upstream providers. Changes in EID-to-RLOC mappings for sites are expected to be handled by configuration, outside of LISP.

10.2. Slow Endpoint Mobility

An individual endpoint wishes to move but is not concerned about maintaining session continuity. Renumbering is involved. LISP can help with the issues surrounding renumbering [RFC4192] [LISA96] by decoupling the address space used by a site from the address spaces used by its ISPs [RFC4984].

10.3. Fast Endpoint Mobility

Fast endpoint mobility occurs when an endpoint moves relatively rapidly, changing its IP-layer network attachment point. Maintenance of session continuity is a goal. This is where the Mobile IPv4 [RFC5944] and Mobile IPv6 [RFC6275] [RFC4866] mechanisms are used and primarily where interactions with LISP need to be explored.

The problem is that as an endpoint moves, it may require changes to the mapping between its EID and a set of RLOCs for its new network location. When this is added to the overhead of Mobile IP binding updates, some packets might be delayed or dropped.

In IPv4 mobility, when an endpoint is away from home, packets to it are encapsulated and forwarded via a home agent that resides in the home area the endpoint’s address belongs to. The home agent will encapsulate and forward packets either directly to the endpoint or to a foreign agent that resides where the endpoint has moved to. Packets from the endpoint may be sent directly to the correspondent node, may be sent via the foreign agent, or may be reverse-tunneled back to the home agent for delivery to the mobile node. As the mobile node’s EID or available RLOC changes, LISP EID-to-RLOC

mappings are required for communication between the mobile node and the home agent, whether via the foreign agent or not. As a mobile endpoint changes networks, up to three LISP mapping changes may be required:

- The mobile node moves from an old location to a new visited network location and notifies its home agent that it has done so. The Mobile IPv4 control packets the mobile node sends pass through one of the new visited network's ITRs, which needs an EID-to-RLOC mapping for the home agent.

- The home agent might not have the EID-to-RLOC mappings for the mobile node's "care-of" address or its foreign agent in the new visited network, in which case it will need to acquire them.

- When packets are sent directly to the correspondent node, it may be that no traffic has been sent from the new visited network to the correspondent node's network, and the new visited network's ITR will need to obtain an EID-to-RLOC mapping for the correspondent node's site.

In addition, if the IPv4 endpoint is sending packets from the new visited network using its original EID, then LISP will need to perform a route-returnability check on the new EID-to-RLOC mapping for that EID.

In IPv6 mobility, packets can flow directly between the mobile node and the correspondent node in either direction. The mobile node uses its "care-of" address (EID). In this case, the route-returnability check would not be needed but one more LISP mapping lookup may be required instead:

- As above, three mapping changes may be needed for the mobile node to communicate with its home agent and to send packets to the correspondent node.

- In addition, another mapping will be needed in the correspondent node's ITR, in order for the correspondent node to send packets to the mobile node's "care-of" address (EID) at the new network location.

When both endpoints are mobile, the number of potential mapping lookups increases accordingly.

As a mobile node moves, there are not only mobility state changes in the mobile node, correspondent node, and home agent, but also state changes in the ITRs and ETRs for at least some EID-Prefixes.

The goal is to support rapid adaptation, with little delay or packet loss for the entire system. Also, IP mobility can be modified to require fewer mapping changes. In order to increase overall system performance, there may be a need to reduce the optimization of one area in order to place fewer demands on another.

In LISP, one possibility is to "glean" information. When a packet arrives, the ETR could examine the EID-to-RLOC mapping and use that mapping for all outgoing traffic to that EID. It can do this after performing a route-returnability check, to ensure that the new network location does have an internal route to that endpoint. However, this does not cover the case where an ITR (the node assigned the RLOC) at the mobile-node location has been compromised.

Mobile IP packet exchange is designed for an environment in which all routing information is disseminated before packets can be forwarded. In order to allow the Internet to grow to support expected future use, we are moving to an environment where some information may have to be obtained after packets are in flight. Modifications to IP mobility should be considered in order to optimize the behavior of the overall system. Anything that decreases the number of new EID-to-RLOC mappings needed when a node moves, or maintains the validity of an EID-to-RLOC mapping for a longer time, is useful.

10.4. Fast Network Mobility

In addition to endpoints, a network can be mobile, possibly changing xTRs. A "network" can be as small as a single router and as large as a whole site. This is different from site mobility in that it is fast and possibly short-lived, but different from endpoint mobility in that a whole prefix is changing RLOCs. However, the mechanisms are the same, and there is no new overhead in LISP. A map request for any endpoint will return a binding for the entire mobile prefix.

If mobile networks become a more common occurrence, it may be useful to revisit the design of the mapping service and allow for dynamic updates of the database.

The issue of interactions between mobility and LISP needs to be explored further. Specific improvements to the entire system will depend on the details of mapping mechanisms. Mapping mechanisms should be evaluated on how well they support session continuity for mobile nodes.

10.5. LISP Mobile Node Mobility

A mobile device can use the LISP infrastructure to achieve mobility by implementing the LISP encapsulation and decapsulation functions and acting as a simple ITR/ETR. By doing this, such a "LISP mobile node" can use topologically independent EID IP addresses that are not advertised into and do not impose a cost on the global routing system. These EIDs are maintained at the edges of the mapping system (in LISP Map-Servers and Map-Resolvers) and are provided on demand to only the correspondents of the LISP mobile node.

Refer to [LISP-MN] for more details.

11. Multicast Considerations

A multicast group address, as defined in the original Internet architecture, is an identifier of a grouping of topologically independent receiver host locations. The address encoding itself does not determine the location of the receiver(s). The multicast routing protocol, and the network-based state the protocol creates, determine where the receivers are located.

In the context of LISP, a multicast group address is both an EID and a Routing Locator. Therefore, no specific semantic or action needs to be taken for a destination address, as it would appear in an IP header. Therefore, a group address that appears in an inner IP header built by a source host will be used as the destination EID. The outer IP header (the destination Routing Locator address), prepended by a LISP router, will use the same group address as the destination Routing Locator.

Having said that, only the source EID and source Routing Locator need to be dealt with. Therefore, an ITR merely needs to put its own IP address in the source 'Routing Locator' field when prepending the outer IP header. This source Routing Locator address, like any other Routing Locator address, MUST be globally routable.

Therefore, an EID-to-RLOC mapping does not need to be performed by an ITR when a received data packet is a multicast data packet or when processing a source-specific Join (either by IGMPv3 or PIM). But the source Routing Locator is decided by the multicast routing protocol in a receiver site. That is, an EID-to-RLOC translation is done at control time.

Another approach is to have the ITR not encapsulate a multicast packet and allow the packet built by the host to flow into the core even if the source address is allocated out of the EID namespace. If the RPF-Vector TLV [RFC5496] is used by PIM in the core, then core

routers can RPF to the ITR (the locator address, which is injected into core routing) rather than the host source address (the EID address, which is not injected into core routing).

To avoid any EID-based multicast state in the network core, the first approach is chosen for LISP-Multicast. Details for LISP-Multicast and interworking with non-LISP sites are described in [RFC6831] and [RFC6832].

12. Security Considerations

It is believed that most of the security mechanisms will be part of the mapping database service when using control-plane procedures for obtaining EID-to-RLOC mappings. For data-plane-triggered mappings, as described in this specification, protection is provided against ETR spoofing by using route-returnability (see Section 3) mechanisms evidenced by the use of a 24-bit 'Nonce' field in the LISP encapsulation header and a 64-bit 'Nonce' field in the LISP control message.

The nonce, coupled with the ITR accepting only solicited Map-Replies, provides a basic level of security, in many ways similar to the security experienced in the current Internet routing system. It is hard for off-path attackers to launch attacks against these LISP mechanisms, as they do not have the nonce values. Sending a large number of packets to accidentally find the right nonce value is possible but would already by itself be a denial-of-service (DoS) attack. On-path attackers can perform far more serious attacks, but on-path attackers can launch serious attacks in the current Internet as well, including eavesdropping, blocking, or redirecting traffic. See more discussion on this topic in Section 6.1.5.1.

LISP does not rely on a PKI or a more heavyweight authentication system. These systems challenge one of the primary design goals of LISP -- scalability.

DoS attack prevention will depend on implementations rate-limiting Map-Requests and Map-Replies to the control plane as well as rate-limiting the number of data-triggered Map-Replies.

An incorrectly implemented or malicious ITR might choose to ignore the Priority and Weights provided by the ETR in its Map-Reply. This traffic-steering would be limited to the traffic that is sent by this ITR's site and no more severe than if the site initiated a bandwidth DoS attack on (one of) the ETR's ingress links. The ITR's site would typically gain no benefit from not respecting the Weights and would likely receive better service by abiding by them.

To deal with map-cache exhaustion attempts in an ITR/PITR, the implementation should consider putting a maximum cap on the number of entries stored with a reserve list for special or frequently accessed sites. This should be a configuration policy control set by the network administrator who manages ITRs and PITRs. When overlapping EID-Prefixes occur across multiple Map-Cache entries, the integrity of the set must be wholly maintained. So, if a more-specific entry cannot be added due to reaching the maximum cap, then none of the less-specific entries should be stored in the map-cache.

Given that the ITR/PITR maintains a cache of EID-to-RLOC mappings, cache sizing and maintenance are issues to be kept in mind during implementation. It is a good idea to have instrumentation in place to detect thrashing of the cache. Implementation experimentation will be used to determine which cache management strategies work best. In general, it is difficult to defend against cache-thrashing attacks. It should be noted that an undersized cache in an ITR/PITR not only causes adverse effects on the site or region it supports but may also cause increased Map-Request loads on the mapping system.

"Piggybacked" mapping data as discussed in Section 6.1.3 specifies how to handle such mappings and includes the possibility for an ETR to temporarily accept such a mapping before verification when running in "trusted" environments. In such cases, there is a potential threat that a fake mapping could be inserted (even if only for a short period) into a map-cache. As noted in Section 6.1.3, an ETR MUST be specifically configured to run in such a mode and might usefully only consider some specific ITRs as also running in that same trusted environment.

There is a security risk implicit in the fact that ETRs generate the EID-Prefix to which they are responding. An ETR can claim a shorter prefix than it is actually responsible for. Various mechanisms to ameliorate or resolve this issue will be examined in the future [LISP-SEC].

Spoofing of inner-header addresses of LISP-encapsulated packets is possible, as with any tunneling mechanism. ITRs MUST verify the source address of a packet to be an EID that belongs to the site’s EID-Prefix range prior to encapsulation. An ETR must only decapsulate and forward datagrams with an inner-header destination that matches one of its EID-Prefix ranges. If, upon receipt and decapsulation, the destination EID of a datagram does not match one of the ETR’s configured EID-Prefixes, the ETR MUST drop the datagram. If a LISP-encapsulated packet arrives at an ETR, it SHOULD compare the inner-header source EID address and the outer-header source RLOC address with the mapping that exists in the mapping database. Then,

when spoofing attacks occur, the outer-header source RLOC address can be used to trace back the attack to the source site, using existing operational tools.

This experimental specification does not address automated key management (AKM). BCP 107 [RFC4107] provides guidance in this area. In addition, at the time of this writing, substantial work is being undertaken to improve security of the routing system [RFC6518] [RFC6480] [BGP-SEC] [LISP-SEC]. Future work on LISP should address the issues discussed in BCP 107 as well as other open security considerations, which may require changes to this specification.

13. Network Management Considerations

Considerations for network management tools exist so the LISP protocol suite can be operationally managed. These mechanisms can be found in [LISP-MIB] and [RFC6835].

14. IANA Considerations

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the LISP specification, in accordance with BCP 26 [RFC5226].

There are four namespaces (listed in the sub-sections below) in LISP that have been registered.

- LISP IANA registry allocations should not be made for purposes unrelated to LISP routing or transport protocols.

- The following policies are used here with the meanings defined in BCP 26: "Specification Required", "IETF Review", "Experimental Use", and "First Come First Served".

14.1. LISP ACT and Flag Fields

New ACT values (Section 6.1.4) can be allocated through IETF review or IESG approval. Four values have already been allocated by this specification (Section 6.1.4).

In addition, LISP has a number of flag fields and reserved fields, such as the LISP header flags field (Section 5.3). New bits for flags in these fields can be implemented after IETF review or IESG approval, but these need not be managed by IANA.

14.2. LISP Address Type Codes

LISP Address [LCAF] type codes have a range from 0 to 255. New type codes MUST be allocated consecutively, starting at 0. Type Codes 0-127 are to be assigned by IETF review or IESG approval.

Type Codes 128-255 are available according to the [RFC5226] First Come First Served policy.

This registry, initially empty, is constructed for future use in experimental work related to LISP Canonical Address Format (LCAF) values. See [LCAF] for details of other possible unapproved address encodings. The unapproved LCAF encodings are an area for further study and experimentation.

14.3. LISP UDP Port Numbers

The IANA registry has allocated UDP port numbers 4341 and 4342 for lisp-data and lisp-control operation, respectively. IANA has updated the description for UDP ports 4341 and 4342 as follows:

lisp-data	4341 udp	LISP Data Packets
lisp-control	4342 udp	LISP Control Packets

14.4. LISP Key ID Numbers

The following Key ID values are defined by this specification as used in any packet type that references a 'Key ID' field:

Name	Number	Defined in
None	0	n/a
HMAC-SHA-1-96	1	[RFC2404]
HMAC-SHA-256-128	2	[RFC4868]

Number values are in the range of 0 to 65535. The allocation of values is on a first come first served basis.

15. Known Open Issues and Areas of Future Work

As an experimental specification, this work is, by definition, incomplete. Specific areas where additional experience and work are needed include the following:

- At present, only [RFC6836] is defined for implementing a database of EID-to-RLOC mapping information. Additional research on other mapping database systems is strongly encouraged.

- Failure and recovery of LISP site partitioning (see Section 6.4) in the presence of redundant configuration (see Section 8.5) needs further research and experimentation.

- The characteristics of map-cache management under exceptional conditions, such as denial-of-service attacks, are not fully understood. Further experience is needed to determine whether current caching methods are practical or in need of further development. In particular, the performance, scaling, and security characteristics of the map-cache will be discovered as part of this experiment. Performance metrics to be observed are packet reordering associated with the LISP Data-Probe and loss of the first packet in a flow associated with map-caching. The impact of these upon TCP will be observed. See Section 12 for additional thoughts and considerations.

- Preliminary work has been done to ensure that sites employing LISP can interconnect with the rest of the Internet. This work is documented in [RFC6832], but further experimentation and experience are needed.

- At present, no mechanism for automated key management for message authentication is defined. Addressing automated key management is necessary before this specification can be developed into a Standards Track RFC. See Section 12 for further details regarding security considerations.

- In order to maintain security and stability, Internet protocols typically isolate the control and data planes. Therefore, user activity cannot cause control-plane state to be created or destroyed. LISP does not maintain this separation. The degree to which the loss of separation impacts security and stability is a topic for experimental observation.

- LISP allows for the use of different mapping database systems. While only one [RFC6836] is currently well defined, each mapping database will likely have some impact on the security of the EID-to-RLOC mappings. How each mapping database system’s security properties impact LISP overall is for further study.

- An examination of the implications of LISP on Internet traffic, applications, routers, and security is needed. This will help implementors understand the consequences for network stability, routing protocol function, routing scalability, migration and backward compatibility, and implementation scalability (as influenced by additional protocol components; additional state; and additional processing for encapsulation, decapsulation, and liveness).

- Experiments need to verify that LISP produces no significant change in the behavior of protocols run between end-systems over a LISP infrastructure versus being run directly between those same end-systems.

- Experiments need to verify that the issues raised in the Critique section of [RFC6115] are either insignificant or have been addressed by updates to LISP.

Other LISP documents may also include open issues and areas for future work.

16. References

16.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2404] Madson, C. and R. Glenn, "The Use of HMAC-SHA-1-96 within ESP and AH", RFC 2404, November 1998.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[RFC3232] Reynolds, J., "Assigned Numbers: RFC 1700 is Replaced by an On-line Database", RFC 3232, January 2002.

[RFC4086] Eastlake, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, June 2005.

[RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, August 2006.

[RFC4868] Kelly, S. and S. Frankel, "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec", RFC 4868, May 2007.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5496] Wijnands, IJ., Boers, A., and E. Rosen, "The Reverse Path Forwarding (RPF) Vector TLV", RFC 5496, March 2009.

[RFC5944] Perkins, C., "IP Mobility Support for IPv4, Revised", RFC 5944, November 2010.

[RFC6115] Li, T., "Recommendation for a Routing Architecture", RFC 6115, February 2011.

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

[RFC6833] Farinacci, D. and V. Fuller, "Locator/ID Separation Protocol (LISP) Map-Server Interface", RFC 6833, January 2013.

[RFC6834] Iannone, L., Saucez, D., and O. Bonaventure, "Locator/ID Separation Protocol (LISP) Map-Versioning", RFC 6834, January 2013.

[RFC6836] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol Alternative Logical Topology (LISP+ALT)", RFC 6836, January 2013.

16.2. Informative References

[AFI] IANA, "Address Family Numbers", <http://www.iana.org/assignments/address-family-numbers>.

[BGP-SEC] Lepinski, M. and S. Turner, "An Overview of BGPSEC", Work in Progress, May 2012.

[CHIAPPA] Chiappa, J., "Endpoints and Endpoint names: A Proposed Enhancement to the Internet Architecture", 1999, <http://mercury.lcs.mit.edu/~jnc/tech/endpoints.txt>.

[CONS] Brim, S., Chiappa, N., Farinacci, D., Fuller, V., Lewis, D., and D. Meyer, "LISP-CONS: A Content distribution Overlay Network Service for LISP", Work in Progress, April 2008.

[EMACS] Brim, S., Farinacci, D., Meyer, D., and J. Curran, "EID Mappings Multicast Across Cooperating Systems for LISP", Work in Progress, November 2007.

[LCAF] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", Work in Progress, January 2013.

[LISA96] Lear, E., Tharp, D., Katinsky, J., and J. Coffin, "Renumbering: Threat or Menace?", Usenix Tenth System Administration Conference (LISA 96), October 1996.

[LISP-DEPLOY] Jakab, L., Cabellos-Aparicio, A., Coras, F., Domingo-Pascual, J., and D. Lewis, "LISP Network Element Deployment Considerations", Work in Progress, October 2012.

[LISP-MIB] Schudel, G., Jain, A., and V. Moreno, "LISP MIB", Work in Progress, January 2013.

[LISP-MN] Farinacci, D., Lewis, D., Meyer, D., and C. White, "LISP Mobile Node", Work in Progress, October 2012.

[LISP-SEC] Maino, F., Ermagan, V., Cabellos, A., Saucez, D., and O. Bonaventure, "LISP-Security (LISP-SEC)", Work in Progress, October 2012.

[LOC-ID-ARCH] Meyer, D. and D. Lewis, "Architectural Implications of Locator/ID Separation", Work in Progress, January 2009.

[OPENLISP] Iannone, L., Saucez, D., and O. Bonaventure, "OpenLISP Implementation Report", Work in Progress, July 2008.

[RADIR] Narten, T., "On the Scalability of Internet Routing", Work in Progress, February 2010.

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC4107] Bellovin, S. and R. Housley, "Guidelines for Cryptographic Key Management", BCP 107, RFC 4107, June 2005.

[RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for Renumbering an IPv6 Network without a Flag Day", RFC 4192, September 2005.

[RFC4866] Arkko, J., Vogt, C., and W. Haddad, "Enhanced Route Optimization for Mobile IPv6", RFC 4866, May 2007.

[RFC4984] Meyer, D., Zhang, L., and K. Fall, "Report from the IAB Workshop on Routing and Addressing", RFC 4984, September 2007.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, February 2012.

[RFC6518] Lebovitz, G. and M. Bhatia, "Keying and Authentication for Routing Protocols (KARP) Design Guidelines", RFC 6518, February 2012.

[RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, January 2013.

[RFC6832] Lewis, D., Meyer, D., Farinacci, D., and V. Fuller, "Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites", RFC 6832, January 2013.

[RFC6835] Farinacci, D. and D. Meyer, "The Locator/ID Separation Protocol Internet Groper (LIG)", RFC 6835, January 2013.

[RFC6837] Lear, E., "NERD: A Not-so-novel Endpoint ID (EID) to Routing Locator (RLOC) Database", RFC 6837, January 2013.

[UDP-TUNNELS] Eubanks, M., Chimento, P., and M. Westerlund, "IPv6 and UDP Checksums for Tunneled Packets", Work in Progress, January 2013.

[UDP-ZERO] Fairhurst, G. and M. Westerlund, "Applicability Statement for the use of IPv6 UDP Datagrams with Zero Checksums", Work in Progress, December 2012.

Appendix A. Acknowledgments

An initial thank you goes to Dave Oran for planting the seeds for the initial ideas for LISP. His consultation continues to provide value to the LISP authors.

A special and appreciative thank you goes to Noel Chiappa for providing architectural impetus over the past decades on separation of location and identity, as well as detailed reviews of the LISP architecture and documents, coupled with enthusiasm for making LISP a practical and incremental transition for the Internet.

The authors would like to gratefully acknowledge many people who have contributed discussions and ideas to the making of this proposal. They include Scott Brim, Andrew Partan, John Zwiebel, Jason Schiller, Lixia Zhang, Dorian Kim, Peter Schoenmaker, Vijay Gill, Geoff Huston, David Conrad, Mark Handley, Ron Bonica, Ted Seely, Mark Townsley, Chris Morrow, Brian Weis, Dave McGrew, Peter Lothberg, Dave Thaler, Eliot Lear, Shane Amante, Ved Kafle, Olivier Bonaventure, Luigi Iannone, Robin Whittle, Brian Carpenter, Joel Halpern, Terry Manderson, Roger Jorgensen, Ran Atkinson, Stig Venaas, Iljitsch van Beijnum, Roland Bless, Dana Blair, Bill Lynch, Marc Woolward, Damien Saucez, Damian Lezama, Attilla De Groot, Parantap Lahiri, David Black, Roque Gagliano, Isidor Kouvelas, Jesper Skriver, Fred Templin, Margaret Wasserman, Sam Hartman, Michael Hofling, Pedro Marques, Jari Arkko, Gregg Schudel, Srinivas Subramanian, Amit Jain, Xu Xiaohu, Dhirendra Trivedi, Yakov Rekhter, John Scudder, John Drake, Dimitri Papadimitriou, Ross Callon, Selina Heimlich, Job Snijders, Vina Ermagan, Albert Cabellos, Fabio Maino, Victor Moreno, Chris White, Clarence Filsfils, and Alia Atlas.

This work originated in the Routing Research Group (RRG) of the IRTF. An individual submission was converted into the IETF LISP working group document that became this RFC.

The LISP working group would like to give a special thanks to Jari Arkko, the Internet Area AD at the time that the set of LISP documents were being prepared for IESG last call, and for his meticulous reviews and detailed commentaries on the 7 working group last call documents progressing toward experimental RFCs.

Authors' Addresses

Dino Farinacci
Cisco Systems
Tasman Drive
San Jose, CA 95134
USA

EMail: farinacci@gmail.com

Vince Fuller

EMail: vaf@vaf.net

Dave Meyer
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

EMail: dmm@1-4-5.net

Darrel Lewis
Cisco Systems
170 Tasman Drive
San Jose, CA
USA

EMail: darlewis@cisco.com